



RADIANCEKIT

Podręcznik Użytkownika

Fotorealistyczna rekonstrukcja 3D
z użyciem Gaussian Splatting

Wersja 1.5.0 · macOS 26.0+ · Maj 2026

BJOERN KINDLER · KINDLER-DEV.DE

Przegląd

Wprowadzenie — Co powinieneś wiedzieć	3
Czym jest RadianceKit?	3
Czym jest Gaussian Splatting?	3
Rozdział 1 — Pasek menu	5
Menu File	6
Menu Mode	9
Menu Training	11
Menu Viewport	14
Menu Export	20
Menu Help	25
Wskazówka: Cmd-Z w menu Edit	29
Skróty klawiszowe w przeglądzie	30
Rozdział 2 — Inspector (Expert View)	31
Sekcja Look (L1–L5)	34
Sekcja Presets (I1–I11)	37
Sekcja Konfiguracja treningu (I12–I22)	43
Sekcja Enhancements (I26–I29, I42–I44)	49
Sekcja Metryki (I30–I38)	56
Sekcja wykresu straty (I39–I41)	62
Kiedy sięgnąć po Inspector?	65
Rozdział 3 — Ustawienia	67
Karta General	68
Karta AI Helpers	73
Ustawienia lustra Inspector'a	76
Kiedy co?	77
Rozdział 4 — Okna pomocnicze	78
User Guide (W1–W4)	79
Keyboard Shortcuts (W5–W6)	82
Manage Storage (W7–W12)	84
Pareto Dashboard (W13–W22)	88
Holdout Analysis (W23–W29)	95
BayesOpt Console (W30–W39)	100
Okno główne: przebieg straty i liczba gaussianów (I39–I41, odsyłacz)	108
Pudełko reguł kciuka	109
Rozdział 6 — Konfiguracja treningu	110
Iteration (T1–T2)	112
Learning Rates (T3–T10)	114

Densification — Classic (T11–T16)	121
Loss (T17–T20)	126
SH-Degree-Progression (T21)	130
Performance (T22–T25)	131
Diagnose und Punktwolken-Vorbereitung (T26–T30)	133
Regularisierung (T31–T37)	136
Refinement (T38–T44)	139
Sky-Dome (T45–T48)	144
Adam + LR-Schedule (T49–T55)	146
Post-Processing + Apple AI (T56–T60)	150
MCMC-Densification (T61–T73)	152
Mip-Splatting (Q1.5) (T74–T76)	159
Adaptive Densification (Q5) (T77–T79)	161
Curriculum (Q6) (T80–T81)	163
Statische Presets (TP1–TP9)	163
Methode:	166
Welches Feld wofür? (Cheat-Sheet)	167
Gefährliche Felder	168
Rozdział 7 — Wbudowane ustawienia wstępne jakości	169
Kiedy które ustawienie wstępne?	179
Szybkie porównanie	180
Własne ustawienia wstępne	182
Rozdział 8 — Formaty eksportu	183
Który format kiedy?	197
Szybkie porównanie	198
Rozdział 9 — Backendy SfM	199
Który backend kiedy?	204
Szybkie porównanie	205
Rozdział 10 — Tryb początkujący	206
Z1 — Import (Bilder & Preset wählen)	206
Z2 — Verarbeitung (SfM + Training)	215
Z3 — Vorschau (3D-Modell drehen)	222
Z4 — Export (Format wählen & speichern)	226
Wechsel zu Expert-Modus	231
Häufige Fragen	232

Jak czytać ten podręcznik

Każdy wpis w podręczniku ma ten sam układ. Po lewej stronie znajdziesz ścieżki obsługi i szczegóły techniczne, a po prawej, w ciepłym panelu bocznym, zawsze proste wyjaśnienie. Małe ikony na początku wiersza zdradzają jednym spojrzeniem, jakiego rodzaju informacja teraz nastąpi.

CZTERY IKONY



Gdzie to znaleźć? Konkretna ścieżka kliknięć w aplikacji — pasek menu, sekcja Inspectora lub krok Trybu początkującego. Tutaj również znajdują się powiązane skróty klawiszowe. Ikona to pinezka mapy i pokazuje: tu w interfejsie użytkownika znajduje się dana funkcja.



Szczegóły. Wartości domyślne, zakresy wartości i ścieżki kodu. Spotkasz to przede wszystkim przy ustawieniach treningu, które nie są pozycją menu, lecz parametrami liczbowymi. Ikona pokazuje małą kartę specyfikacji.



Technicznie. Co funkcja robi wewnętrznie, jakie parametry mają wpływ, na co reaguje i jakie ma efekty uboczne. Dla czytelników, którzy chcą zrozumieć, co dzieje się w tle. Ikona to blok suwaków i symbolicznie reprezentuje pokrętła pod maską.



Prostymi słowami. Główne przesłanie wyrażone jasnymi słowami — bez żargonu, bez kodu. Przeczytaj tę sekcję najpierw, jeśli chcesz tylko szybko wiedzieć, do czego służy dana funkcja i kiedy jej potrzebujesz. Ikona to dymek dialogowy i oznacza „w skrócie”. Ta kolumna ma zawsze ciepły piaskowy odcień, aby oko od razu ją znalazło.

KOLORY ROZDZIAŁÓW

Każdy rozdział ma własny kolor akcentu, który rozpoznasz po identyfikatorze (na przykład **M1**) na lewo od każdego tytułu wpisu oraz po małych ikonach przed nim. Przeglądając, od razu widzisz, w którym rozdziale aktualnie się znajdujesz.

- 1** Menu
- 2** Inspector
- 3** Ustawienia
- 4** Okna pomocnicze
- 6** Trening
- 7** Ustawienia wstępne
- 8** Eksporty
- 9** SfM
- 10** Tryb początkujący

WSKAZÓWKI NAWIGACJI

Szybki start. Jeśli interesuje cię tylko obsługa, przejdź bezpośrednio do **Rozdziału 10 – Tryb początkujący**. To prowadzona wersja w czterech krokach i nie wymaga żadnej wcześniejszej wiedzy.

Głębsze wejście. **Rozdział 2 – Inspector** oraz **Rozdział 7 – Ustawienia wstępne** wyjaśniają elementy sterujące i predefiniowane profile jakości dostępne w Trybie eksperta.

Wyszukiwanie. Spis treści i wyszukiwanie pełnotekstowe PDF pomagają znaleźć konkretną funkcję. Nie musisz czytać podręcznika od początku do końca.

Wprowadzenie — Co powinieneś wiedzieć

Czym jest RadianceKit?

RadianceKit to natywna aplikacja macOS, która z serii zwykłych zdjęć lub filmu tworzy interaktywną, „chodzalną” rekonstrukcję 3D. Wejściami są na przykład od 50 do 500 ujęć wykonanych wokół obiektu, przez pomieszczenie lub nad krajobrazem. Wyjściem jest tzw. scena Gaussian Splatting — model 3D, który możesz oglądać na Macu w czasie rzeczywistym z dowolnej perspektywy, który można eksportować i osadzać na stronach internetowych, a który w głównych aspektach wygląda fotorealistycznie.

Aplikacja działa całkowicie lokalnie na twoim Macu — żadne obrazy nie są przesyłane do chmury, nie jest wymagane logowanie, nie ma abonamentu. Intensywnie wykorzystuje GPU twojego Maca z układem Apple Silicon (seria M): pełny trening może trwać od dwóch minut do kilku godzin w zależności od sceny i ustawienia wstępnego. Podczas obliczeń możesz normalnie dalej pracować na Macu; RadianceKit działa w tle i zgłosi się, gdy wynik będzie gotowy.

Istnieją dwa tryby obsługi: *Tryb początkujący* (Simple Mode) prowadzi cię w czterech krokach przez przepływ Import → Wybór ustawienia wstępnego → Trening → Eksport. *Tryb eksperta* (Expert Mode) otwiera duży Inspector ze wszystkimi pokrętkami, okno podglądu na żywo i wykresy diagnostyczne. Możesz w każdej chwili przełączać się między trybami; dane sceny pozostają zachowane.

Czym jest Gaussian Splatting?

Gaussian Splatting (często w skrócie 3DGS lub po prostu *Splatting*) to stosunkowo nowa metoda fotorealistycznej prezentacji 3D, zaprezentowana w 2023 roku w artykule z Graz i INRIA. Idea: zamiast modelować scenę jako klasyczną siatkę wielokątną (trójkąty) lub jako siatkę wokseli, składa się ją z milionów małych, miękkich „chmur” 3D — każda pojedyncza chmurka to rozkład Gaussa 3D (stąd nazwa) z własną pozycją, rozmiarem, kształtem, kolorem i przezroczystością. Te chmurki są tak trenowane, aby z wszystkich kątów widzenia twoich zdjęć wejściowych razem dawały odpowiedni obraz.

W praktyce oznacza to: Gaussian Splatting potrafi przedstawiać odbicia, blaski, miękkie liście, włosy lub zasłony tak, jak klasyczne modelowanie 3D nie może lub może tylko ogromnym kosztem. W zamian wynik nie jest edytowalnym modelem 3D w klasycznym sensie — nie możesz po prostu przesunąć pojedynczej ściany ani przestawić wazonu. To raczej *zamrożone ujęcie* przestrzeni, przez które możesz się swobodnie poruszać. Dla wielu zastosowań — wizualizacja architektoniczna, prezentacja produktu, wirtualne zwiedzanie, kryminalistyka, dziedzictwo kulturowe — to właśnie odpowiednia siła.







Aby z obrazów wejściowych powstała scena 3D, potrzebne są dwa kroki. Najpierw aplikacja oblicza za pomocą procedury zwanej *Structure-from-Motion (SfM)*, gdzie znajdowała się twoja kamera przy każdym zdjęciu. Przy okazji powstaje zgrubna chmura punktów sceny. Następnie startuje właściwy trening Gaussian Splatting: wychodząc od tej zgrubnej chmury, miliony chmurek 3D są stopniowo rozprawdane, powiększane, dopracowywane i korygowane w pozycji i kolorze, aż z wszystkich kątów widzenia wejściowych dadzą pasujący obraz.

Nie musisz wiedzieć o żadnym z tych elementów, aby korzystać z RadianceKit. Tryb początkujący całkowicie ukrywa te kroki. Ale jeśli chcesz zrozumieć, co oznaczają liczby diagnostyczne w Trybie eksperta (iteracja, loss, gaussianów, SSIM ...) lub dlaczego niektóre sceny stają się ładniejsze od innych, to w późniejszych rozdziałach podręcznika znajdziesz odpowiedzi.

ROZDZIAŁ

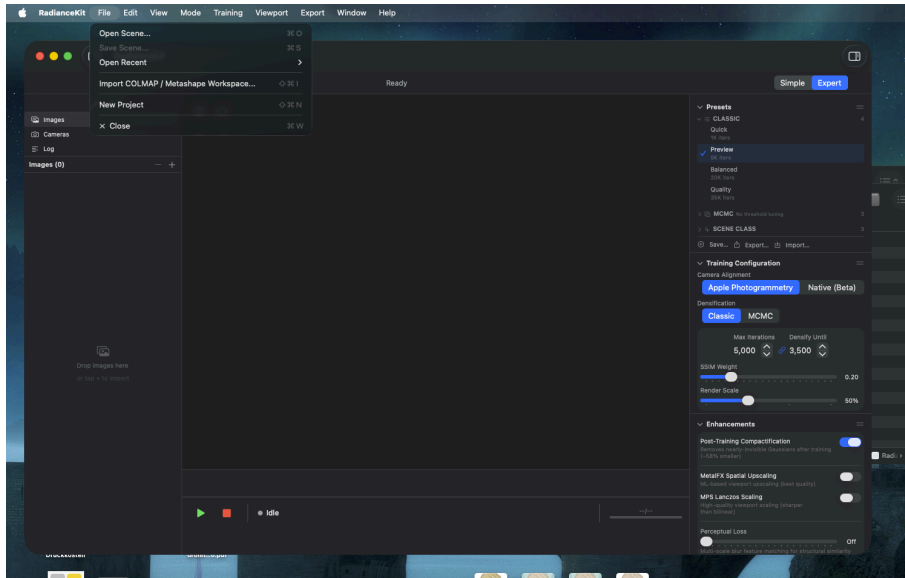
Rozdział 1 — Pasek menu

Pasek menu RadianceKit grupuje wszystkie funkcje, które nie znajdują się bezpośrednio w oknie głównym lub Inspectorze. To przede wszystkim akcje działające na całą scenę (otwieranie, zapisywanie, nowy projekt), sterowanie treningiem (start, pauza, kontynuacja), obsługa widoku (auto-rotacja, zrzut ekranu, kolor tła) oraz uruchamianie eksportów do różnych formatów 3D i medialnych. Dochodzą do tego przejścia do wszystkich okien pomocniczych (User Guide, Pareto Dashboard, Holdout Analysis, BayesOpt Console).

Skróty klawiszowe znajdują się po prawej stronie każdej pozycji menu. Konwencje:  oznacza klawisz Command (Apple),  to Shift,  to Option (Alt) a  to Control. Przykład:  oznacza Shift+Command+T. Wszystkie udokumentowane tu skróty są dodatkowo wyszczególnione w osobnym oknie przeglądu przez Help → Keyboard Shortcuts (/).

Następujące 42 pozycje są udokumentowane w kolejności inwentarza (M1–M42), pogrupowane według odpowiedniego menu na najwyższym poziomie. Wszystkie pozycje zostały zweryfikowane wobec bieżącego stanu kodu (linie 175–477). Żadne pozycje nie zostały usunięte ani przestarzałe w stosunku do inwentarza; nowa pozycja menu Edit (Cmd-Z dla „Remove Image„) jest dostarczana przez systemowy framework NSUndoManager i dlatego nie pojawia się w kodzie RadianceKitApp (zobacz wskazówkę na końcu rozdziału).

Menu File



Rysunek 1: Menu File rozwinięte — pozycje M1 do M6

Menu File zastępuje standardową pozycję Apple „New Window,, akcjami specyficznymi dla projektu. Obejmuje ładowanie/zapisywanie scen, dynamiczną listę Recent, import workspace oraz twarde reset do stanu pustego.

M1 File > Open Scene...

GDZIE

Pasek menu → File → Open Scene... (⌘O).

TECHNICZNIE

Otwiera okno dialogowe pliku dla formatów pakietu `RadianceScene`, `.ply`, `.splat` i `.spz`. Pojedynczy wybór, może pokazywać zarówno pliki, jak i katalogi (dla formatu pakietu). Po udanym wyborze ścieżka jest wpisywana na listę Recent, a scena ładowana asynchronicznie — poprzednia jest zastępowana, a potok treningu inicjalizowany załadowanym stanem. Pliki PLY/SPZ/Splat są odczytywane przez odpowiednie loadery formatu; pakiet `.radiancecene` to katalog z manifestem, snapshotem chmury i wynikami SfM.

PROSTYMI SŁOWAMI

Tak ładujesz już wytrenowaną scenę ponownie do aplikacji. Działa z własnym formatem RadianceKit oraz ze standardowymi formatami PLY, SPLAT i SPZ, które tworzą inne programy do splattingu. Użyj tego, gdy np. wytrenowałeś scenę przez noc i chcesz następnego dnia kontynuować lub eksportować. Przy otwieraniu dotychczasowy stan w oknie głównym jest zastępowany — zapisz więc wcześniej, jeśli zależy ci jeszcze na bieżącej scenie. Ścieżka automatycznie łąduje w „Open Recent,, (M3), abyś następnym razem szybciej do niej dotarł.

M2 File > Save Scene...

Pasek menu → File → Save Scene... (#S).

 **TECHNICZNIE**

Otwiera okno dialogowe zapisu pliku z Content-Type pakietu `RadianceScene` i wstępnie wypełnioną nazwą pliku `scene.radiancecene`. Zapisuje pakiet katalogu z `manifest.json`, serializowaną chmurą gaussianów (snapshot PLY) i zrzutem wyniku SfM, tak aby po ponownym otwarciu działało również Continue-Training. Pozycja jest nieaktywna, dopóki nie istnieją gaussiany. Nie zapisuje do ścieżki logów treningu, lecz tam, gdzie wskazuje okno dialogowe zapisu — typowo pod `~/Documents/`.

 **PROSTYMI SŁOWAMI**

Zapisuje twoją bieżącą scenę jako plik (dokładniej: jako folder pakietu, który wygląda jak plik). Dopiero potem możesz tę scenę później ponownie otworzyć przez „Open Scene...” (M1). W pakiecie łądają zarówno chmura gaussianów, jak i wynik SfM, dzięki czemu możesz także później dodać Continue-Training (M12–M14). Dopóki nie zakończyłeś jeszcze treningu, pozycja jest wyszarzona. Domyślna nazwa to `scene.radiancecene` — możesz jednak w oknie dialogowym Save podać własną nazwę.

M3 File > Open Recent > [nazwa sceny]

Pasek menu → File → Open Recent → (lista).

 **TECHNICZNIE**

Dynamiczne podmenu generowane z listy ostatnio otwieranych ścieżek (zapisanej w ustawieniach). Każdy wpis listy jest oznaczony nazwą pliku i ładowany po kliknięciu. Gdy lista jest pusta, zamiast tego pojawia się dezaktywowana etykieta „No Recent Scenes”. W stylu Apple lista trzyma N ostatnio otwieranych scen — ograniczenie ma miejsce przy zapisie do ustawień, a nie w samym builderze menu.

 **PROSTYMI SŁOWAMI**

Tutaj widzisz ostatnio otwarte sceny i możesz jednym kliknięciem wskoczyć do nich z powrotem, bez konieczności przechożenia przez okno dialogowe pliku. Jeśli właśnie zacząłeś, lista jest pusta i wyświetla się szaro w menu. Każda scena, którą otwierasz przez „Open Scene...” (M1), automatycznie łąduje na tej liście. Jeśli kiedyś lista zrobi się zbyt pełna albo chcesz ją wyczyścić z powodów prywatności, użyj „Clear Recent” (M4).

M4 File > Open Recent > Clear Recent GDZIE

Pasek menu → File → Open Recent → Clear Recent.

 TECHNICZNIE

Czyści listę Recent w ustawieniach. Działa natychmiast, bez dialogu potwierdzenia. Pozycja pojawia się w podmenu tylko, gdy w liście Recent w ogóle są jakieś wpisy (znajduje się pod dzielnikiem po ścieżkach).

 PROSTYMI SŁOWAMI

Usuwa listę ostatnio otwartych scen. Praktyczne, gdy bawiłeś się testowym zestawem danych i nie chcesz już widzieć ścieżek. Same pliki scen nie są przy tym usuwane — tylko skróty w menu. Akcja działa natychmiast, bez pytania; potem w podmenu pojawia się „No Recent Scenes„. Pozycja pojawia się tylko wtedy, gdy w liście są jakieś sceny — przy pustej liście jest niewidoczna.

M5 File > Import COLMAP / Metashape Workspace... GDZIE

Pasek menu → File → Import COLMAP / Metashape Workspace... (⇧⌘I).

 TECHNICZNIE

Otwiera selektor folderu. Oczekuje folderu z układem workspace COLMAP (np. `sparse/0/cameras.{bin,txt}` plus `images/`). Po wyborze przeprowadzana jest wstępna weryfikacja workspace — rozpoznaje trzy układy (`sparse/0/`, `sparse/`, `root`) oraz czy rekonstrukcja jest binarna (`cameras.bin`) czy tekstowa ETH3D (`cameras.txt`). Przy powodzeniu workspace jest importowany; w przeciwnym razie pojawia się tylko ostrzeżenie w logu aplikacji. Zobacz także rozdział 9 „Backendy SfM„, Q6 dla pełnej logiki potoku.

 PROSTYMI SŁOWAMI

Jeśli używasz Metashape, COLMAP, RealityCapture lub podobnego oprogramowania do rekonstrukcji kamer i masz eksport, wczytujesz tutaj folder. RadiancKit pomija wtedy etap SfM i startuje od razu z trenin-
giem — to przy dużych scenach oszczędza godziny. Przeciągnięcie i upuszczenie na okno główne działa tak samo. Oczekiwany jest folder z układem COLMAP (czyli `sparse/0/` z `cameras.*` plus folder `images/`). Więcej o obsługiwanych układach i przepływach pracy w rozdziale 9 „Backendy SfM„.

M6 File > New Project



GDZIE

Pasek menu → File → New Project (⇧⌘N).

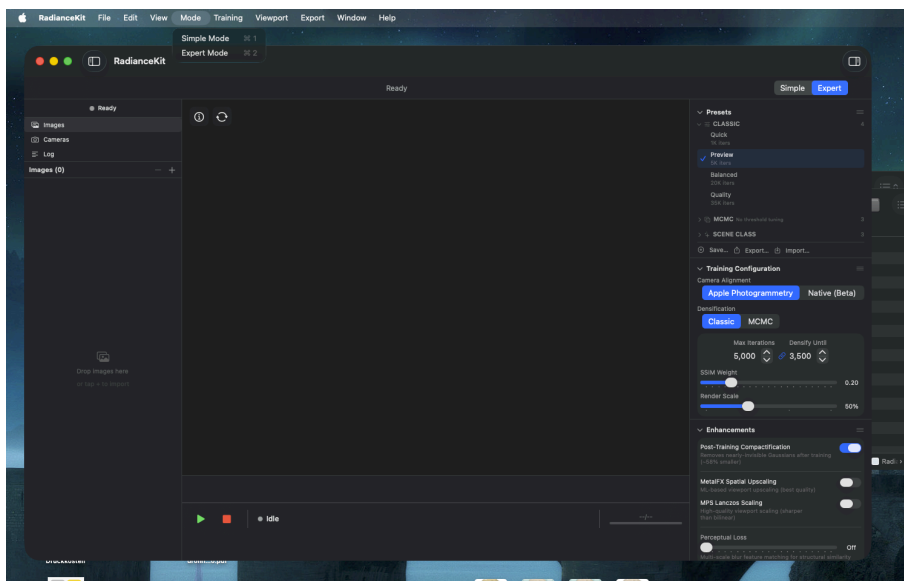
TECHNICZNIE

Sprawdza, czy jest niezapisana praca. Jeśli tak, pojawia się dialog potwierdzenia, zanim cokolwiek przepadnie. Jeśli nie ma nic do zapisania, reset działa bezpośrednio — czyści zaimportowane obrazy, wynik SfM, chmurę gaussianów, stan treningu i wszystkie zależne wskaźniki UI. Uwaga: utworzona przez użytkownika biblioteka ustawień wstępnych pozostaje zachowana, ponieważ znajduje się w ustawieniach aplikacji, a nie w stanie projektu.

PROSTYMI SŁOWAMI

Resetuje wszystko do pustego startu — tak, jakbyś właśnie świeżo otworzył aplikację. Jeśli masz jeszcze niezapisaną pracę, aplikacja wcześniej pyta. Użyj tego, gdy chcesz zacząć z zupełnie inną sceną. Zaimportowane obrazy, wynik SfM, chmura gaussianów i stan treningu są całkowicie wymazywane. Twoje własne ustawienia wstępne pozostają jednak zachowane, ponieważ znajdują się w ustawieniach aplikacji i nie należą do sceny.

Menu Mode



Rysunek 2: Menu Mode z przełącznikami Simple- i Expert-Mode

Dwa proste przełączniki między prowadzonym Simple Mode (typu kreator, 4 kroki) a pełnym Expert Mode (klasyczny układ Inspector'a ze wszystkimi pokrętkami).

M7 Mode > Simple Mode

GDZIE

Pasek menu → Mode → Simple Mode (⌘1).

TECHNICZNIE

Przełącza stan aplikacji w tryb Simple. Główny obszar aplikacji pokazuje wtedy prowadzony przepływ pracy zamiast układu Expert. Stan trybu jest zapisywany w ustawieniach (zobacz S1 „Default Mode„ w rozdziale 3 Settings).

PROSTYMI SŁOWAMI

Przełącza na wariant krok-po-kroku, w którym aplikacja prowadzi cię przez import, przetwarzanie, podgląd i eksport. Polecane, gdy dopiero zaczynasz lub szybko potrzebujesz wyniku. Większość pokręteł szczegółowych jest ukryta — pracujesz z sensownymi domyślnymi wartościami. Jeśli potem chcesz zagłębić się bardziej, po prostu przełącz w Expert Mode (M8). Który tryb jest aktywny przy starcie aplikacji, możesz ustawić w ustawieniach (rozdział 3, S1).

M8 Mode > Expert Mode

GDZIE

Pasek menu → Mode → Expert Mode (⌘2).

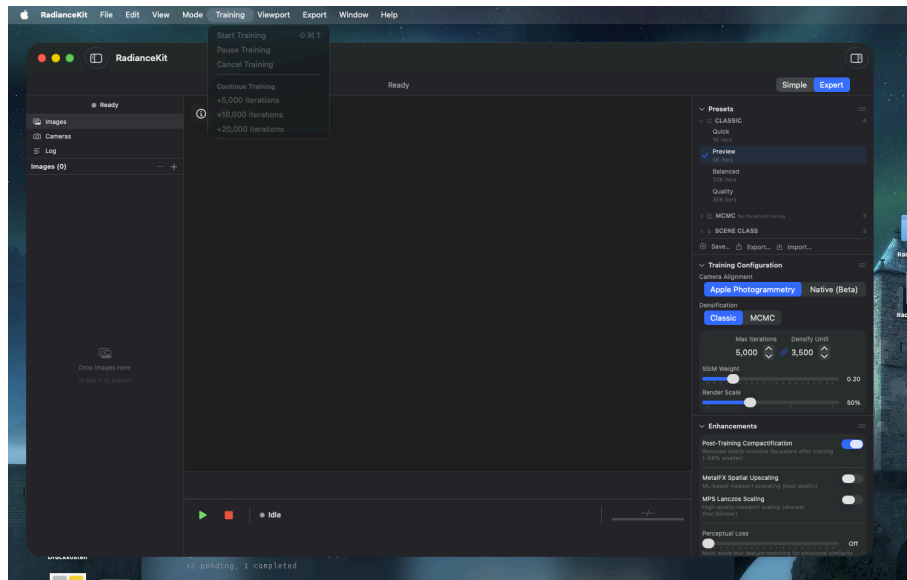
TECHNICZNIE

Przełącza stan aplikacji w Expert Mode. Pojawia się wtedy pełny układ Inspectora ze wszystkimi sekcjami (Presets, TrainingConfig, Enhancements, Metrics, LossChart, ProjectNavigator). W Expert Mode dostępne są wszystkie parametry treningu, selektor COLMAP, przełączniki Mid-Compact i diagnostyka. Również Live-Preview działa tylko w tym trybie.

PROSTYMI SŁOWAMI

Przełącza w pełny widok ze wszystkimi pokrętłami. Widzisz tu wykresy loss w czasie rzeczywistym, możesz precyzyjnie dobrać wszystkie parametry i zarządzać kilkoma porównawczymi konfiguracjami przez ustawienia wstępne. Polecane, gdy chcesz zrozumieć, co trening robi wewnętrznie, lub gdy chcesz celowo eksperymentować. Również Live-Preview, selektor COLMAP i diagnostyka są dostępne tylko tutaj. Jeśli czujesz się przytłoczony, wróć przez M7 do Simple Mode — twoja scena pozostaje przy tym zachowana.

Menu Training



Rysunek 3: Menu Training z podmenu Continue — pozycje M9 do M14

Cztery akcje wokół przebiegu treningu: start, pauza, anulowanie i przedłużenie o określoną liczbę iteracji. Wszystkie trzy pozycje Continue są zabramkowane IAP (w wersji Free-Trial nieklikalne).

M9 Training > Start Training

GDZIE

Pasek menu → Training → Start Training (⇧⌘T).

TECHNICZNIE

Startuje potok treningu asynchronicznie. Wymaganie: istnieje wynik SfM i obecnie nie działa żaden inny potok. Oba warunki blokują pozycję, jeśli niespełnione. Przy starcie odczytywane są bieżące wartości konfiguracji, tworzony nowy log JSONL pod `~/Documents/RadianceKit/Logs/training_YYYY-MM-DD_HHmss.jsonl`, i w zależności od wyboru strategii jechana jest ścieżka klasyczna lub MCMC. Stan treningu zmienia się z „idle„ na „training“.

PROSTYMI SŁOWAMI

Naciska duży zielony przycisk — gdy tylko zaimportujesz zdjęcia i rekonstrukcja kamer będzie gotowa, rozpoczyna się właściwy trening Gaussian Splatting. Pozwól aplikacji działać; w zależności od ustawienia wstępnego od 1 minuty (Quick) do kilku godzin (MCMC Quality). Pozycja pozostaje szara, dopóki nie ma jeszcze wyniku SfM lub trwa inny potok. Każdy przebieg zapisuje przy okazji log do `~/Documents/RadianceKit/Logs/`, który możesz później ocenić przez Pareto Dashboard (M40).

M10 Training > Pause Training **GDZIE**

Pasek menu → Training → Pause Training.

 **TECHNICZNIE**

Pauzuje działający trening. Odblokowywany tylko, gdy stan treningu to „training„. Pauzowanie zatrzymuje pętlę iteracji w następnym bezpiecznym punkcie synchronizacji, zachowuje pełny stan GPU (bufory gaussianów, momenty optimizera, pozycja schedulera) i przełącza na „paused“. Wznowienie następuje przez ponowne naciśnięcie (tytuł pozycji jest statyczny — aplikacja przełącza jednak między Pause/Resume we właściwej logice). Pauzowane treningi nie przeżywają wyjścia z aplikacji; w takim przypadku zapisz scenę i rozszerz później przez pozycję Continue-Training (M12–M14).

 **PROSTYMI SŁOWAMI**

Zatrzymuje trening na chwilę bez utraty postępu. Praktyczne, gdy komputer jest ci potrzebny chwilowo do czegoś ważniejszego. Ponowne kliknięcie wznawia. Nie działa przez restarty aplikacji — jeśli naprawdę chcesz później kontynuować, zakończ trening przez Cancel (M11), zapisz scenę przez Save Scene (M2) i potem użyj Continue Training (M12–M14). Podczas pauzy GPU całkowicie odpoczywa; pamięć pozostaje jednak zajęta.

M11 Training > Cancel Training **GDZIE**

Pasek menu → Training → Cancel Training.

 **TECHNICZNIE**

Anuluje działający trening. Aktywne, gdy stan treningu nie jest „idle„. Ustawia flagę anulowania w silniku treningu, co czysto kończy pętlę iteracji w następnym punkcie synchronizacji, zapisuje finalny wpis Summary do logu JSONL i resetuje stan na „idle“. Dotychczas wytrenowana chmura pozostaje zachowana (można ją zapisać lub eksportować), ale jest oznaczona jako „cancelled„.

 **PROSTYMI SŁOWAMI**

Definitywnie anuluje działający trening. Dotychczasowy stan pozostaje — jeśli więc po kilku tysiącach iteracji masz już prezentowalny wynik, możesz go potem mimo wszystko wyeksportować. Jeśli chcesz tylko krótko przerwać, użyj zamiast tego Pause (M10). W logu treningu przebieg jest oznaczony jako „cancelled„, finalna wartość loss jest mimo to zapisywana. Anulowaną scenę możesz również później kontynuować przez Continue Training (M12–M14), o ile aplikacja w międzyczasie nie została zamknięta.

M12 Training > Continue Training > +5 000 iteracji **GDZIE**

Pasek menu → Training → Continue Training → +5,000 iterations.

 **TECHNICZNIE**

Kontynuuje trening o 5 000 iteracji. Aktywne, gdy zakończony trening jest kontynuowalny i wersja pełna jest odblokowana. Kontynuowalność obowiązuje, gdy istnieje zakończony trening, a pełny stan optimizera nadal jest w pamięci. Przy Continue prowadzone są dalej momenty Adam i scheduler LR, tak że kontynuacja zachowuje się jak ciągły przebieg 25K/45K/60K zamiast nowego startu. Log JSONL otrzymuje nowy wpis konfiguracji z setupem przyrostowym. Dostępne tylko w wersji pełnej.

 **PROSTYMI SŁOWAMI**

Dodaje 5 000 dalszych kroków treningu. Użyj tego, gdy wynik po pierwszym przebiegu jest blisko, ale jeszcze nie do końca ostry. Działa tylko w płatnej wersji pełnej. W przeciwieństwie do całkowicie nowego przebiegu stan optimizera pozostaje zachowany, dzięki czemu kontynuacja sprawia wrażenie ciągłego przebiegu. Jeśli potrzebujesz więcej niż 5 000 kroków, weź od razu M13 (+10 000) lub M14 (+20 000).

M13 Training > Continue Training > +10 000 iteracji **GDZIE**

Pasek menu → Training → Continue Training → +10,000 iterations.

 **TECHNICZNIE**

Identyczne jak M12, ale z 10 000 dodatkowymi iteracjami. Te same warunki wstępne, ta sama ścieżka schedulera LR. Polecane, gdy początkowy trening został przeprowadzony z ustawieniem średniego poziomu i chcesz zobaczyć znaczącą poprawę jakości bez całkowitego restartowania przebiegu.

 **PROSTYMI SŁOWAMI**

Wydłuża trening o 10 000 kroków — środkowa z trzech dostępnych wartości Continue. Dobry wybór, gdy pierwszy przebieg był okej, ale chcesz jeszcze wyraźniej lepiej. Tak jak w M12 i M14 przebieg współczynnika uczenia jest płynnie kontynuowany, zamiast startować od nowa. Dostępne tylko w wersji pełnej.

M14 Training > Continue Training > +20 000 iteracji

GDZIE

Pasek menu → Training → Continue Training → +20,000 iterations.

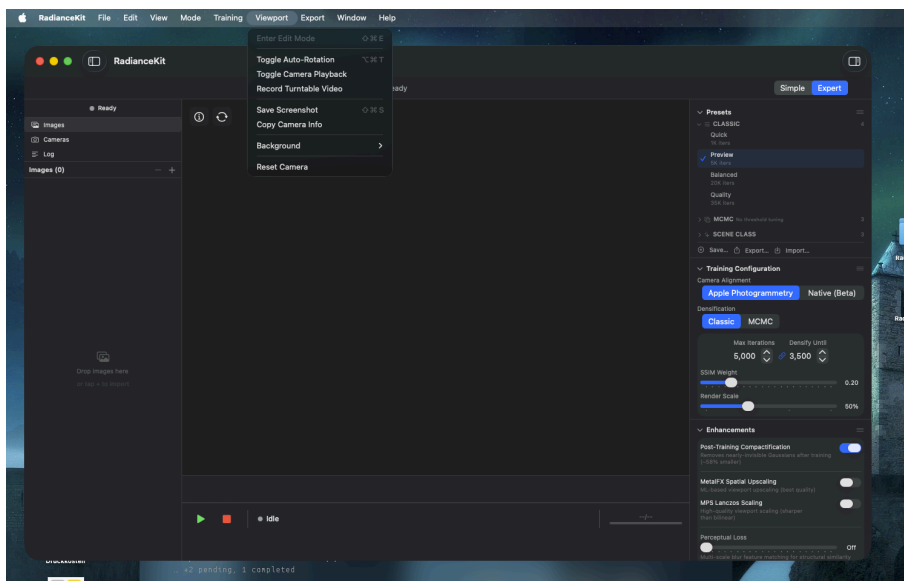
TECHNICZNIE

Identyczne jak M12 / M13, ale z 20 000 dodatkowymi iteracjami. Największy predefiniowany skok Continue. Przy treningach MCMC to często jest to, co decyduje o różnicy między „pasuje”, a „przydatne do benchmarku”; przy Classic od 35–40K z doświadczenia przybywa niewiele.

PROSTYMI SŁOWAMI

Dodaje 20 000 dodatkowych kroków treningu, maksymalną wartość Continue. Użyj tego, gdy naprawdę chcesz wycisnąć ostatnią kroplę jakości. Przy klasycznym treningu po 40 000 kroków często niewiele to już daje — przy MCMC natomiast często się opłaca, ponieważ tam konwergencja przebiega wolniej. Licz się w zależności od sceny ze znaczącym dodatkowym czasem. Jak M12 i M13 ta pozycja również jest dostępna tylko w wersji pełnej.

Menu Viewport



Rysunek 4: Menu Viewport z Edit-Mode, sterowaniem kamerą i podmenu Background

Steruje widokiem 3D: Edit-Mode do selekcji gaussianów i cleanupu, sterowanie kamerą (auto-rotacja, odtwarzanie, nagrywanie), zrzut ekranu, kolor tła i reset.

M15 Viewport > Enter/Exit Edit Mode **GDZIE**

Pasek menu → Viewport → Enter Edit Mode (lub „Exit Edit Mode”, w zależności od stanu). $\text{⇧}\text{⌘}\text{E}$.

 **TECHNICZNIE**

Tytuł pozycji jest dynamiczny i pokazuje w zależności od stanu „Exit Edit Mode” lub „Enter Edit Mode”. Przy naciśnięciu Edit-Mode w rendererze widoku jest przełączany. Przy wyjściu z trybu Edit dodatkowo resetowana jest bieżąca selekcja. Tryb Edit aktywuje selekcję kliknięciem na gaussianach, selekcję prostokątem i usuwanie zaznaczonych gaussianów (zobacz obszar edytora UI). Dezaktywowane, dopóki nie jest podłączony renderer widoku.

 **PROSTYMI SŁOWAMI**

Przełącza między normalnym widokiem 3D a trybem edycji, w którym możesz zaznaczać pojedyncze gaussiany i usuwać (np. floatery lub odstające punkty w tle). Przy wyjściu selekcja jest automatycznie resetowana. Pozycja pozostaje szara, dopóki w widoku nie jest widoczna jeszcze scena. Etykieta przełącza się w zależności od stanu między „Enter Edit Mode”, a „Exit Edit Mode” — zawsze więc widzisz, w jakim trybie obecnie jesteś.

M16 Viewport > Toggle Auto-Rotation **GDZIE**

Pasek menu → Viewport → Toggle Auto-Rotation ($\text{⌘}\text{⇧}\text{T}$).

 **TECHNICZNIE**

Włącza lub wyłącza ciągłą rotację kamery widoku wokół pionowej osi przez środek sceny. Oś i prędkość pochodzą z konfiguracji sterowania kamerą. Auto-rotacja jest czystym efektem widoku i nie wpływa ani na trening, ani na nagrywanie — jeśli równolegle używasz rejestratora wideo Turntable (M18), auto-rotacja dostarcza dokładnie tę ścieżkę, którą rejestrator przechwytuje.

 **PROSTYMI SŁOWAMI**

Obraca kamerę cały czas powoli wokół twojej sceny, abyś mógł ją widzieć ze wszystkich stron, bez przeciągania myszą. Ponowne kliknięcie zatrzymuje rotację. Praktyczne przy oglądaniu gotowych wytrenowanych scen lub jako animacja tła do demo na żywo. Jeśli równolegle nagrywasz wideo (M18), auto-rotacja dostarcza dokładnie ten ruch, który rejestrator przechwytuje.

M17 Viewport > Toggle Camera Playback **GDZIE**

Pasek menu → Viewport → Toggle Camera Playback.

 **TECHNICZNIE**

Przełącza odtwarzanie ścieżki kamery. Gdy istnieje zarejestrowana ścieżka kamery (np. z poprzedniego nagrania lub ponieważ została wczytana `transforms.json`), ścieżka jest odtwarzana — kamera widoku nie porusza się więc już po wejściach mysz/trackpad, lecz odtwarza trajektorię klatka po klatce. Ponowne naciśnięcie pauzuje odtwarzanie.

 **PROSTYMI SŁOWAMI**

Odtwarza wcześniej zarejestrowaną lub zaimportowaną ścieżkę kamery. Tak możesz prześledzić oryginalną ścieżkę, którą zrobiona była scena, lub sprawdzić planowany ruch orbitalny przed eksportem wideo. Podczas trwania odtwarzania wejścia mysz i trackpad są dezaktywowane — kamera ściśle podąża za ścieżką. Ponowne kliknięcie pauzuje odtwarzanie. Jeśli nie wczytałeś ani nie zarejestrowałeś żadnej ścieżki kamery, nic się nie dzieje.

M18 Viewport > Record Turntable Video **GDZIE**

Pasek menu → Viewport → Record Turntable Video.

 **TECHNICZNIE**

Przełącza nagrywanie widoku. Przy pierwszym naciśnięciu startuje rejestracja klatek do tymczasowej ścieżki; przy drugim naciśnięciu nagrywanie jest kończone, kodowane i zapisywane do ścieżki MP4 (ścieżka jest pytana przez okno dialogowe zapisu). W przeciwieństwie do Export → Media → Orbit Video (M31), który tworzy stałą ścieżkę 360° z konfigurowalnym czasem trwania, rejestrator Turntable nagrywa *na żywo* to, co widzisz w widoku — możesz więc również nagrać ręczną ścieżkę kamery.

 **PROSTYMI SŁOWAMI**

Nagrywa bezpośrednio w widoku wideo. Niezależnie od tego, czy kamera obraca się automatycznie, czy poruszasz nią sam myszą — wszystko, co widzisz, jest zapisywane do pliku MP4. W przeciwieństwie do eksportu „Orbit Video„ (M31) sam określasz ścieżkę kamery. Pierwsze kliknięcie startuje nagrywanie, drugie kończy je i pyta o miejsce zapisu. Praktyczne, gdy np. chcesz pokazać konkretne ujęcie szczegółu, które nie byłoby możliwe ze sztywnym ruchem orbitalnym.

M19 Viewport > Save Screenshot GDZIE

Pasek menu → Viewport → Save Screenshot (⇧#S).

 TECHNICZNIE

Przechwytuje pojedynczą klatkę widoku w pełnej rozdzielczości renderingu (czyli nie układ pikseli okna, lecz pełna zawartość celu renderingu) jako plik PNG. Ścieżka jest pytana przez okno dialogowe zapisu. Kolor tła (M21–M23) jest wpalany. Ustawienia upscalingu MetalFX/MPS z Enhancements (zobacz I27/I28) działają, jeśli aktywne — zrzut ekranu pokazuje więc upscalowany output.

 PROSTYMI SŁOWAMI

Zapisuje migawkę twojego bieżącego widoku 3D jako obraz PNG. Praktyczne do materiałów marketingowych lub szybkiego porównania. Uwaga: tło jest częścią obrazu — jeśli potrzebujesz przezroczystości, lepiej wyeksportuj plik sceny. Rozdzielczość odpowiada wewnętrznemu celowi renderingu, a nie twojej wielkości okna — obraz jest więc często ostrzejszy, niż wygląda w oknie. Ewentualne ustawienia upscalingu (Inspector → Enhancements) również wchodzą w grę.

M20 Viewport > Copy Camera Info GDZIE

Pasek menu → Viewport → Copy Camera Info.

 TECHNICZNIE

Odczytuje bieżącą pozę kamery widoku (pozycja, punkt look-at, wektor up) oraz wartości FOV ze sterowania kamerą i zapisuje je jako wieloliniowy tekst do schowka. Format jest czytelny dla człowieka (label = value na linię), nie JSON. Praktyczne, by odtworzyć specyficzny widok do celów debugowania lub udostępnić wsparciu.

 PROSTYMI SŁOWAMI

Kopiuje bieżącą pozycję kamery i kierunek patrzenia jako tekst do schowka. Jeśli chcesz np. pokazać współprogramiście, skąd jakieś miejsce w scenie wygląda dziwnie, wkleisz tekst po prostu do maila lub okna czatu. Format jest czytelny dla człowieka (jedna linia na wartość), nie JSON. Głównie przeznaczony do zgłaszania błędów lub zapytań do wsparcia.

M21 Viewport > Background > Dark Gray GDZIE

Pasek menu → Viewport → Background → Dark Gray.

 TECHNICZNIE

Ustawia kolor tła widoku na ciemny szary (RGB 0.1/0.1/0.1). Renderer używa tego koloru jako tła, na którym gaussiany są kompozytowane. Domyślny kolor przy starcie aplikacji steruje opcja Settings S3 „Default Viewport Background„.

 PROSTYMI SŁOWAMI

Koloruje tło widoku 3D na ciemnoszaro. Standardowy wybór dla większości scen — oferuje dobry kontrast zarówno dla jasnych, jak i ciemnych gaussianów, bez tego by oko czepiało się czystej powierzchni czarnej lub białej. Kolor jest również przejmowany w rzutach ekranu (M19) i orbitalnych wideo (M31). Jeśli Dark Gray jest dla Ciebie zbyt mało spektakularny, dla porównania wypróbuj również Black (M22) lub White (M23). Który kolor jest aktywny przy starcie aplikacji, możesz ustawić w ustawieniach (S3).

M22 Viewport > Background > Black GDZIE

Pasek menu → Viewport → Background → Black.

 TECHNICZNIE

Ustawia kolor tła widoku na czystą czerń (RGB 0/0/0). Pomaga, jeśli scena ma wiele jasnych floaterów i chcesz je zidentyfikować, lub do materiałów marketingowych z ciemnym look-and-feel.

 PROSTYMI SŁOWAMI

Czarne tło. Dobre dla bardzo jasnych scen lub gdy chcesz zajrzeć w tryb Edit i szukasz małych jasnych gaussianów (floaterów), które giną w szarości. Również idealne do materiałów marketingowych z ciemnym, dramatycznym wyglądem. Kolor jest wpalany w rzuty ekranu i wideo orbitalne — jeśli potrzebujesz przezroczystości do późniejszego compositu, czerń jest najgorszym wyborem. Dla ciemnych floaterów przełącz w drugą stronę na White (M23).

M23 Viewport > Background > White **GDZIE**

Pasek menu → Viewport → Background → White.

 **TECHNICZNIE**

Ustawia kolor tła widoku na czystą biel (RGB 1/1/1). Użyteczne, gdy scena ma przeważnie ciemną wartość i chcesz zobaczyć ciemne floatery (typowy szum tła plenerowego).

 **PROSTYMI SŁOWAMI**

Białe tło. Praktyczne, gdy motyw lepiej wygląda jasne-na-ciemnym, lub by znaleźć ciemne odstające punkty, które możesz potem usunąć w trybie Edit (M15). Przy scenach plenerowych biel jest często bardziej użyteczna niż czerń, ponieważ typowe floatery plenerowe są raczej ciemne. Jak przy innych opcjach tła kolor jest przejmowany w zrzutach ekranu i wideo.

M24 Viewport > Reset Camera **GDZIE**

Pasek menu → Viewport → Reset Camera.

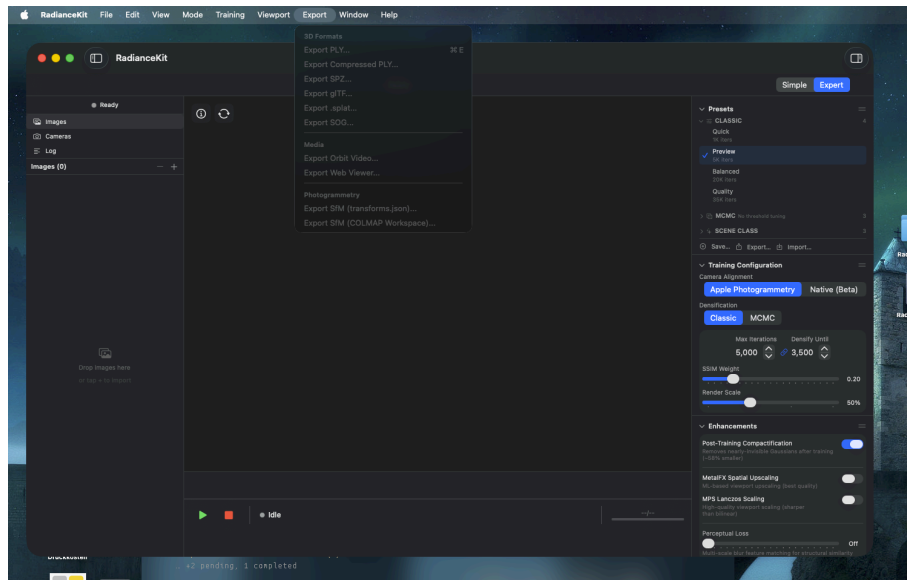
 **TECHNICZNIE**

Resetuje kamerę widoku, opuszcza widok kamery treningu i zatrzymuje auto-rotację. W ten sposób kamera wraca na początkową pozycję (typowo: przed sceną, lekko z góry patrząc), auto-rotacja jest wyłączona, a jeśli renderer właśnie pokazywał kamerę treningu (jedną z póź SfM), wraca do Free-Camera.

 **PROSTYMI SŁOWAMI**

Przywraca kamerę widoku w pozycję startową. Jeśli zgubiłeś się przy obracaniu lub wypchnąłeś scenę poza kadr — jedno kliknięcie tutaj i znów widzisz, co powinieneś widzieć. Jednocześnie wyłącza auto-rotację, jeśli właśnie działa, i wraca z zamrożonej kamery treningu w wolny widok. Tak otrzymujesz w każdym przypadku czysty restart widoku.

Menu Export



Rysunek 5: Menu Export z trzema grupami podmenu — 3D Formats, Media i Photogrammetry

Osiem celów eksportu plus dwa eksporty fotogrametrii, zgrupowane w trzech sekcjach (3D Formats, Media, Photogrammetry). Pierwsze sześć budowanych jest przez wspólną rutynę pomocniczą, która otwiera okno dialogowe zapisu i rejestruje eksport w katalogu formatów. Pozycje fotogrametrii mają indywidualną logikę. Wszystkie eksporty fotogrametrii i niektóre eksporty 3D są dostępne tylko w wersji pełnej.

M25 Export > 3D Formats > Export PLY...

GDZIE

Pasek menu → Export → 3D Formats → PLY (⌘E).

TECHNICZNIE

Otwiera okno dialogowe zapisu z domyślną nazwą `gaussians.ply`. Przy OK bieżąca chmura gaussianów jest zapisywana do standardowego formatu PLY ASCII/binarnego — zgodnego z SuperSplat, PolyCam, PlayCanvas i wszystkimi typowymi viewrami 3DGS. Pełne współczynniki SH, pełna precyzja (Float32 na pole). Rozmiar pliku często kilkaset MB przy $\geq 500K$ gaussianów.

PROSTYMI SŁÓWAMI

Zapisuje twoją scenę 3D jako standardowy plik PLY. To najbardziej uniwersalny format — prawie każde oprogramowanie może to wczytać, od SuperSplat przez PolyCam po PlayCanvas. Pliki są jednak duże, często kilkaset megabajtów. Użyj PLY, gdy chcesz kontynuować w pełnej jakości lub archiwizować. Jeśli chcesz scenę udostępnić przez sieć, zobacz raczej SPZ (M27) lub Compressed PLY (M26) — są znacznie mniejsze.

M26 Export > 3D Formats > Export Compressed PLY... GDZIE

Pasek menu → Export → 3D Formats → Compressed PLY.

 TECHNICZNIE

Zapisuje chmurę gaussianów w formacie Compressed-PLY z własną kwantyzacją pól Position, Scale, Rotation i SH. 5–10x mniejsze pliki niż nieskompresowane PLY (M25) przy minimalnych stratach wizualnych. Zgodne z SuperSplat (który czyta standard Compressed-PLY) i PlayCanvas. Domyślna nazwa pliku `gaussians_compressed.ply`.

 PROSTYMI SŁOWAMI

Jak normalne PLY, ale 5–10 razy mniejsze. Jakość pozostaje prawie taka sama. Użyj tego, gdy chcesz udostępnić plik online lub wysłać mailem. Działa bezpośrednio z SuperSplat i PlayCanvas. Jeśli jednak twój system docelowy potrzebuje jeszcze mniejszych plików (mobile, demo przeglądarkowe), weź zamiast tego SPZ (M27) — to jeszcze agresywniej skompresowane. Dla pełnej jakości edycji weź nieskompresowane PLY (M25).

M27 Export > 3D Formats > Export SPZ... GDZIE

Pasek menu → Export → 3D Formats → SPZ.

 TECHNICZNIE

Zapisuje chmurę gaussianów w formacie SPZ — formacie splat opublikowanym przez Niantic z agresywną kwantyzacją (~90% mniejszy niż nieskompresowane PLY). Optymalizowany przede wszystkim pod viewery webowe i aplikacje mobilne. Zgodny z Niantic Splatt3R, `gsplat.js` i viewerem przeglądarkowym Niantic.

 PROSTYMI SŁOWAMI

Jeden z najmniejszych formatów. Około 10x mniejszy niż normalne PLY. Użyj tego przede wszystkim, gdy chcesz pokazać scenę w przeglądarce lub oglądać przez aplikację mobilną. Dla maksymalnej jakości PLY jest lepszym wyborem. SPZ jest opracowany przez Niantic i działa bezpośrednio z `gsplat.js`, Splatt3R i viewerem webowym Niantic. Z powodu silnej kompresji nie możesz plików SPZ łatwo dalej trenować — do edycji weź PLY.

M28 Export > 3D Formats > Export glTF... GDZIE

Pasek menu → Export → 3D Formats → glTF.

 TECHNICZNIE

Zapisuje plik `.glb` (binarny glTF) z rozszerzeniem `KHR_gaussian_splatting`. Zgodny ze standardem, odpowiedni do potoków używających silników glTF jak Babylon.js lub Three.js, implementujących rozszerzenie `KHR_gaussian_splatting`.

 PROSTYMI SŁOWAMI

Zapisuje scenę w formacie glTF, który rozumie wiele programów 3D i silników webowych — pod warunkiem, że obsługują rozszerzenie Gaussian Splatting. Jeśli masz specyficzny potok 3D (np. Three.js lub Babylon.js), który to rozumie, to jest twój format. Plik wychodzi jako binarny `.glb` — pojedynczy pakiet zawierający wszystko. Dla klasycznych przepływów pracy splattingu zwykle PLY lub SPZ to lepszy wybór, ponieważ więcej narzędzi je bezpośrednio rozumie.

M29 Export > 3D Formats > Export .splat... GDZIE

Pasek menu → Export → 3D Formats → .splat.

 TECHNICZNIE

Zapisuje format `.splat` Antimatter15 — stały rozmiar 32 bajty na gaussiana (pozycja jako $3 \times \text{Float32}$, skala jako $3 \times \text{Float32}$, rotacja jako $4 \times \text{Uint8}$ znormalizowany kwaternion, RGB+Opacity jako $4 \times \text{Uint8}$). Brak współczynników SH wyższych niż DC. Najmniejszy plik z bezpośrednią zgodnością z przeglądarką. Dla `gsplat.js` i viewera online demo `antimatter15`.

 PROSTYMI SŁOWAMI

Najprostszy format viewera webowego. Mały i natychmiast wyświetlalny w każdej przeglądarce. Traci jednak detal oświetleniowy (wyższe współczynniki SH przepadają — splat wygląda z każdego kąta tak samo, zamiast reagować na światło). Dobry do maksymalnej wydajności webowej, do fotorealizmu raczej SPZ lub PLY. Działa z vieweirem online `antimatter15` i `gsplat.js`. Każdy gaussian zajmuje stałe 32 bajty, co czyni format prostym i kompatybilnym — ale właśnie za cenę głębi szczegółu.

M30 Export > 3D Formats > Export SOG... GDZIE

Pasek menu → Export → 3D Formats → SOG.

 TECHNICZNIE

Zapisuje chmurę gaussianów w formacie SOG. SOG („Self-Organizing Gaussian”) to format PlayCanvas z układem atlasu tekstur i kompresją WebP danych kwantyzowanych. Skaluje się z 15–20× lepszym współczynnikiem rozmiaru niż PLY. Eksport wywołuje wewnętrznie `cwebp` jako zewnętrzne narzędzie — dlatego w wersji sandbox (App Store) potencjalnie ograniczony.

 PROSTYMI SŁOWAMI

Bardzo mały format dla przepływów PlayCanvas. Około 15–20 razy mniejszy niż PLY, ponieważ dane są pakowane do układu atlasu tekstur i kompresowane WebP. Jeśli nie masz przepływu PlayCanvas, SPZ lub Compressed PLY zwykle to lepszy wybór. Eksport wywołuje wewnętrznie `cwebp` jako zewnętrzne narzędzie — w wersji App Store (sandbox) ten krok może być ograniczony.

M31 Export > Media > Export Orbit Video... GDZIE

Pasek menu → Export → Media → Orbit Video.

 TECHNICZNIE

Renderuje orbitę 360° wokół środka sceny i koduje ją jako MP4 (H.264) lub MOV (HEVC, w zależności od domyślnych systemowych). W przeciwieństwie do M18 (nagrywanie na żywo) ścieżka jest tutaj sztywno zadana — czas trwania wybierany w ustawieniach lub w kroku eksportu Simple Mode.

 PROSTYMI SŁOWAMI

Tworzy automatycznie wideo z obrotem wokół twojej sceny. Brak ręcznego poruszania. Dobre do social media lub szybkiego demo. Jeśli chcesz sam sterować kamerą, użyj zamiast tego Record Turntable Video (M18). Ścieżka jest stała: pełna orbita 360° wokół środka sceny, czas trwania wybierasz w ustawieniach lub w kroku eksportu Simple Mode. Wideo jest w zależności od systemu wyjściem jako H.264-MP4 lub HEVC-MOV.

M32 Export > Media > Export Web Viewer... GDZIE

Pasek menu → Export → Media → Web Viewer.

 TECHNICZNIE

Pakuje samodzielny viewer HTML (oparty na gsplat.js) plus dane gaussianów zakodowane w Base64 do pojedynczego pliku `.html`. Plik ten działa offline w każdej nowoczesnej przeglądarce — brak zależności od serwera, brak zewnętrznych URL-i. Rozmiar pliku to około współczynnik 1.3 większy niż wariant SPZ (przez narzut Base64).

 PROSTYMI SŁOWAMI

Zapisuje twoją scenę jako samo-uruchamiającą się stronę WWW. Podwójne kliknięcie na plik HTML → przeglądarka otwiera się → gotowa interaktywna scena 3D. Działa bez internetu, można ją wysłać mailem, to najprostszyszy sposób udostępnienia wyniku przyjacielom lub klientom. Plik zawiera kompletny viewer gsplat.js oraz dane gaussianów w pojedynczym dokumencie — nic nie jest doładowywane z sieci. Rozmiar pliku to około jedna trzecia większa niż eksport SPZ, ale za to u odbiorcy nie potrzebujesz dodatkowego oprogramowania.

M33 Export > Photogrammetry > Export SfM (transforms.json)... GDZIE

Pasek menu → Export → Photogrammetry → Export SfM (transforms.json).

 TECHNICZNIE

Własna ścieżka eksportu (nie przez wspólną rutynę pomocniczą), ponieważ eksportowana jest nie chmura gaussianów, lecz wynik SfM. Otwiera okno dialogowe zapisu z `transforms.json` jako domyślną nazwą i Content-Type `json`. Przy OK zapisywany jest zgodny z nerfstudio plik `transforms.json` z wewnętrznymi parametrami kamer, pozami (jako macierz 4x4 w konwencji NeRF) i ścieżkami klatek. Tekst pomocy w UI wskazuje, że obrazy treningowe muszą zostać skopiowane jako siostrzany folder `images/`. Aktywne tylko, gdy istnieje wynik SfM i wersja pełna jest odblokowana.

 PROSTYMI SŁOWAMI

Jeśli chcesz dalej używać wyniku SfM w innym oprogramowaniu jak nerfstudio, Brush, gsplat lub OpenSplat, eksportujesz tutaj pozycje kamer. Umieść dodatkowo swoje obrazy treningowe w folderze `images/` obok pliku `transforms.json` — inaczej program docelowy nie będzie mógł przypisać obrazów. Pozycja jest wyszarzona, dopóki nie ma jeszcze wyniku SfM, i w wersji Free-Trial zablokowana. Dla przepływu COLMAP-Workspace weź zamiast tego M34.

M34 Export > Photogrammetry > Export SfM (COLMAP Workspace)...

GDZIE

Pasek menu → Export → Photogrammetry → Export SfM (COLMAP Workspace).

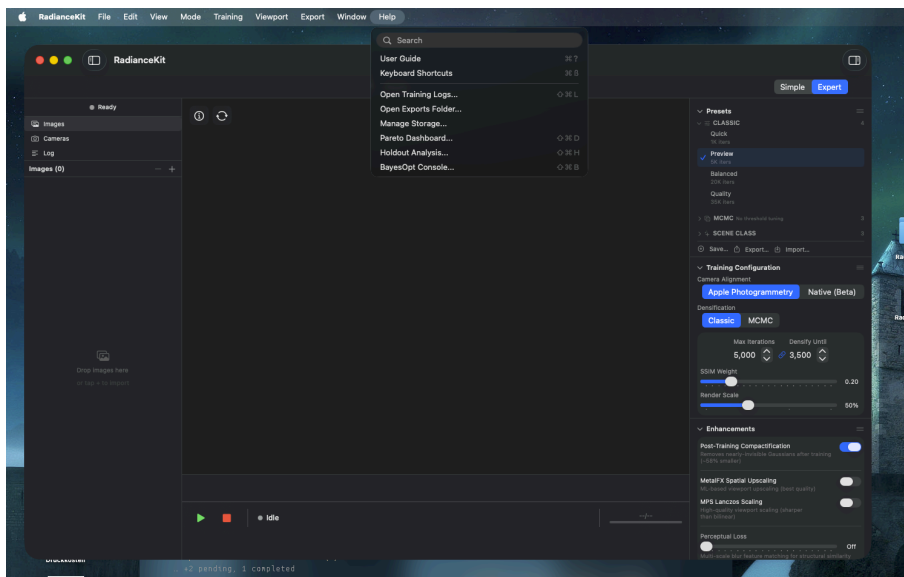
TECHNICZNIE

Otwiera okno dialogowe zapisu z domyślną nazwą `colmap-workspace` (bez rozszerzenia, ponieważ to folder). Zapisuje standardowy workspace COLMAP z `sparse/0/cameras.bin`, `images.bin`, `points3D.bin`. Pozwala otworzyć obliczoną w RadianceKit lub zaimportowaną rekonstrukcję SfM w innych narzędziach jak Postshot, Nerfstudio lub Meshroom, lub przy ponownym przebiegu A/B wczytać do samego RadianceKit jako już-obliczone wejście (przez M5) — oszczędza czas obliczeń. Aktywne tylko, gdy istnieje wynik SfM i wersja pełna jest odblokowana.

PROSTYMI SŁOWAMI

Jak M33, ale w formacie COLMAP zamiast nerfstudio. Jeśli używasz Postshot, Meshroom, Nerfstudio lub innego narzędzia z przepływem COLMAP, to jest twój eksport. Praktyczny efekt uboczny: możesz ten folder później przez M5 ponownie wczytać do RadianceKit i oszczędzić sobie czas obliczeń SfM przy następnym przebiegu — szczególnie przy dużych scenach zysk czasu rzędu godzin. Jak M33 dostępne tylko, gdy istnieje wynik SfM, i w wersji Free-Trial zablokowane.

Menu Help



Rysunek 6: Menu Help z pozycjami dokumentacji, folderów i analizy

Siedem pozycji: dwa okna dokumentacji (User Guide, Keyboard Shortcuts), trzy skróty do folderów (Training Logs, Exports, Storage) oraz trzy okna analizy (Pareto Dashboard, Holdout Analysis, BayesOpt Console). Zgodnie ze stylem Apple menu Help pojawia się zupełnie z prawej strony. Standardowe menu Help jest całkowicie zastąpione własnym wariantem RadianceKit.

M35 Help > User Guide **GDZIE**

Pasek menu → Help → User Guide (⌘?).

 **TECHNICZNIE**

Otwiera okno User Guide. Pokazuje nawigację z paskiem bocznym tematów i obszarem szczegółów przewijania w domyślnym rozmiarze 860×640. Treści są statycznie umieszczone (nie parsowane z Markdown).

 **PROSTYMI SŁOWAMI**

Otwiera wewnętrzną instrukcję aplikacji. Jeśli nie chcesz czytać wszystkiego w tym podręczniku, znajdziesz tam najważniejsze kroki bezpośrednio w programie. Instrukcja jest zbudowana jako osobne okno z paskiem bocznym tematów — możesz więc celowo skakać do poszczególnych tematów. Treści są krótsze niż ten podręcznik i koncentrują się na najczęstszych przepływach pracy.

M36 Help > Keyboard Shortcuts **GDZIE**

Pasek menu → Help → Keyboard Shortcuts (⌘/).

 **TECHNICZNIE**

Otwiera okno Keyboard Shortcuts — prosty układ przewijania ze wszystkimi skrótami aplikacji, zgrupowanymi według menu na najwyższym poziomie. Domyślny rozmiar 440×560. Treści są również statycznie umieszczone.

 **PROSTYMI SŁOWAMI**

Otwiera okno z kompletną listą wszystkich skrótów klawiszowych. Jeśli np. nie pamiętasz, którym klawiszem uruchamiasz trening, zaglądasz tam. Podsumowanie znajduje się także na końcu tego rozdziału. Lista jest zgrupowana według menu na najwyższym poziomie, tak że szybko skoczysz do właściwego obszaru. Pomocne, gdy przestawiasz się ze stylu myszy na klawiaturę.

M37 Help > Open Training Logs... **GDZIE**

Pasek menu → Help → Open Training Logs... (⇧⌘L).

 **TECHNICZNIE**

Oblicza folder logów jako ~/Documents/RadianceKit/Logs, tworzy go w razie potrzeby i otwiera w Finderze. Każdy przebieg treningu zapisuje tam własny plik JSONL `training_YYYY-MM-DD_HHmms.jsonl`.

 **PROSTYMI SŁOWAMI**

Otwiera w Finderze folder z wszystkimi dotychczasowymi protokołami treningu. Jeśli coś poszło źle lub chcesz sprawdzić, kiedy dokładnie trening do jakiej wartości zbiegł, znajdziesz to tutaj w plikach JSONL. Na każdy przebieg treningu zakładany jest dokładnie jeden plik ze znacznikiem czasu — możesz go również wczytać do innych narzędzi lub wysłać mailem do wsparcia. Jeśli chcesz oceny graficznej, Pareto Dashboard (M40) to lepsze wejście.

M38 Help > Open Exports Folder... **GDZIE**

Pasek menu → Help → Open Exports Folder...

 **TECHNICZNIE**

Analogicznie do M37, ale z ~/Documents/RadianceKit/Exports. Tworzony przy pierwszym przebiegu auto-testu lub przy pierwszym kliknięciu; potem lądują tam standardowe ścieżki wszystkich eksportów auto-testu (np. `autotest_<timestamp>.ply`). Wybrane ręcznie przez okno dialogowe zapisy eksporty NIE muszą iść tutaj, lecz tam, gdzie użytkownik je zapisuje — dlatego ten folder jest interesujący przede wszystkim dla auto-testów.

 **PROSTYMI SŁOWAMI**

Otwiera folder, w którym aplikacja składowe swoje własne eksporty (przede wszystkim przebiegi auto-testu). Jeśli zapisałeś jakiś eksport ręcznie przez okno dialogowe zapisu gdzie indziej, jest on tam, a nie w tym folderze. Praktyczne do sprzątnięcia lub by sprawdzić, ile miejsca zajmują wcześniejsze eksporty testowe. Jeśli potrzebujesz kompletnego przeglądu wraz z logami i pakietami scen, weź zamiast tego Manage Storage (M39).

M39 Help > Manage Storage...

Pasek menu → Help → Manage Storage...



Otwiera Storage-Browser (zobacz rozdział 4 Okna pomocnicze, ID W7–W12). Listuje wszystkie persistentowane sceny, logi treningu, eksporty i cache w folderze ~/Documents/RadianceKit/ z rozmiarem, umożliwia Revealing i Move-to-Trash na wpis.

 PROSTYMI SŁOWAMI

Otwiera przeglądarkę okna pokazującą, ile miejsca RadianceKit zajmuje na twoim dysku — na scenę, log i eksport. Możesz bezpośrednio usuwać pojedyncze rzeczy bez konieczności chodzenia do Findera. Praktyczne po dłuższym użytkowaniu, gdy dysk się zapełnia — wcześniejsze logi i eksporty auto-testów mogą sumarycznie zajmować kilka gigabajtów. Przez Revealing w każdej chwili dostajesz się również do klasycznego widoku.

M40 Help > Pareto Dashboard...

Pasek menu → Help → Pareto Dashboard... (⇧⌘D).



Otwiera Pareto Dashboard (zobacz rozdział 4, ID W13–W22). Dashboard ładuje wszystkie logi JSONL treningów z ~/Documents/RadianceKit/Logs/, porządkuje je według sceny i ustawienia wstępnego oraz rysuje wykres scatter Pareto (standard: Loss vs Gaussians, opcjonalnie Loss vs Wallclock lub PSNR vs iteracje).

 PROSTYMI SŁOWAMI

Otwiera przegląd wszystkich dotychczasowych przebiegów treningu jako wykres. Od razu widzisz, który przebieg dostarczył najlepszej równowagi jakości i rozmiaru. Praktyczne, gdy chcesz porównać różne ustawienia wstępne ze sobą. Domyślnie wykres pokazuje Loss przeciwko liczbie gaussianów — możesz jednak również przełączyć na czas wallclock lub PSNR. Dane pochodzą z logów JSONL treningu (M37); im więcej masz przebiegów, tym bardziej wymowna staje się ocena.

M41 Help > Holdout Analysis...

Pasek menu → Help → Holdout Analysis... (⇧⌘H).



Otwiera okno Holdout Analysis (zobacz rozdział 4, ID W23–W29). Ładuje `transforms.json`, rysuje kamery jako globus 3D i pozwala na rozdziały train/test fold (kątowo lub liniowo, 2–8 foldów). Wyjściem jest `fold-assignment.json`, którego trening może użyć w odpowiednich konfiguracjach treningu jako zestaw testowy.

PROSTYMI SŁOWAMI

Pomaga ci rozdzielić twoje ujęcia kamer na zestawy treningowy i testowy — abyś mógł obiektywnie zmierzyć, jak dobra jest twoja scena (na obrazach, których trening nie widział). Raczej narzędzie badawcze i benchmarkowe. Kamery są przedstawiane jako globus 3D; możesz wybierać między 2 a 8 foldami, równomiernie w kącie lub liniowo po kolejności. Wynikiem jest mały plik JSON, którego trening potem używa jako zestawu testowego.

M42 Help > BayesOpt Console...

Pasek menu → Help → BayesOpt Console... (⇧⌘B).



Otwiera konsolę BayesOpt (zobacz rozdział 4, ID W30–W39). Ładuje predefiniowane przestrzenie poszukiwań (np. „MCMC scale-reg + opacity-reg + ssim„), wykonuje próby Bayesian Optimization asynchronicznie i pokazuje krzywą zbieżności oraz log prób na żywo.

PROSTYMI SŁOWAMI

Wbudowana konsola auto-tunera. Zamiast ręcznie próbować różnych parametrów, aplikacja może to sama uruchamiać przez noc i pod koniec proponować ci najlepsze wartości dla twojej sceny. Bardzo zaawansowane narzędzie — dla większości przepływów pracy dobre ustawienie wstępne (zobacz rozdział 7) wystarcza. Wybierasz predefiniowaną przestrzeń poszukiwań (np. „MCMC scale-reg + opacity-reg + ssim„) i widzisz na żywo krzywą zbieżności oraz log prób. Zaplanuj w zależności od konfiguracji kilka godzin do dni.

Wskazówka: Cmd-Z w menu Edit

Od maja 2026 Project Navigator w Expert Mode wspiera usuwanie zaimportowanych obrazów przez przycisk minus lub klawisz Backspace, a cofanie przez `Cmd-Z`. Ta akcja `Cmd-Z` pojawia się w menu Edit macOS (dostarczonym przez SwiftUI) jako „Undo Remove Image„, dopóki usunięty obraz jest jeszcze przywracalny. Jest rejestrowana przez standardowy system; dlatego nie ma własnego wpisu z ID M w inwentarzu.

Skróty klawiszowe w przeglądarce

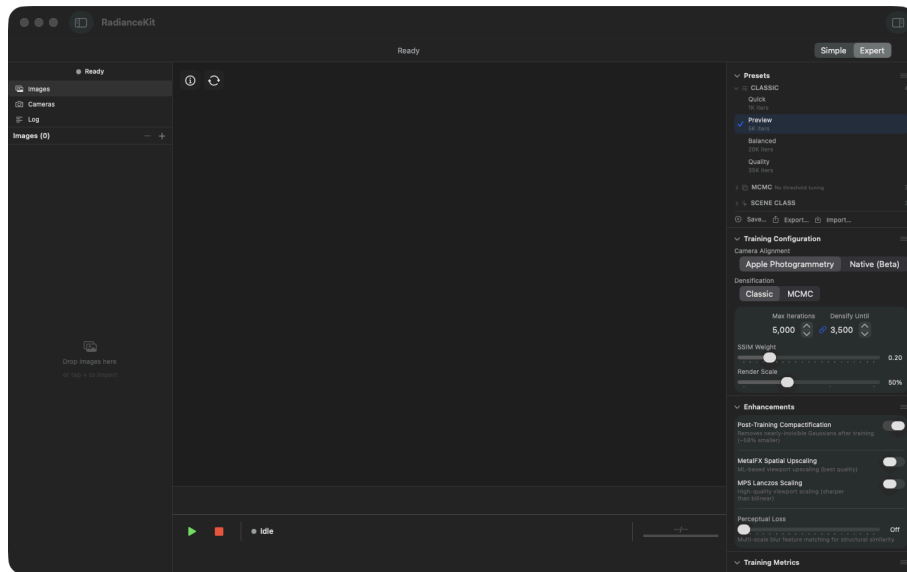
Pozycja menu	Skrót
File > Open Scene...	⌘O
File > Save Scene...	⌘S
File > Import COLMAP / Metashape Workspace...	⇧⌘I
File > New Project	⇧⌘N
Mode > Simple Mode	⌘1
Mode > Expert Mode	⌘2
Training > Start Training	⇧⌘T
Viewport > Enter/Exit Edit Mode	⇧⌘E
Viewport > Toggle Auto-Rotation	⌘⌥T
Viewport > Save Screenshot	⇧⌘S
Export > 3D Formats > PLY	⌘E
Help > User Guide	⌘?
Help > Keyboard Shortcuts	⌘/
Help > Open Training Logs...	⇧⌘L
Help > Pareto Dashboard...	⇧⌘D
Help > Holdout Analysis...	⇧⌘H
Help > BayesOpt Console...	⇧⌘B

Menu Edit (dostarczane przez system, w Expert Mode przy aktywnej selekcji Project Navigator):

Akcja	Skrót
Undo Remove Image	⌘Z
Remove Selected Image	Backspace / Delete

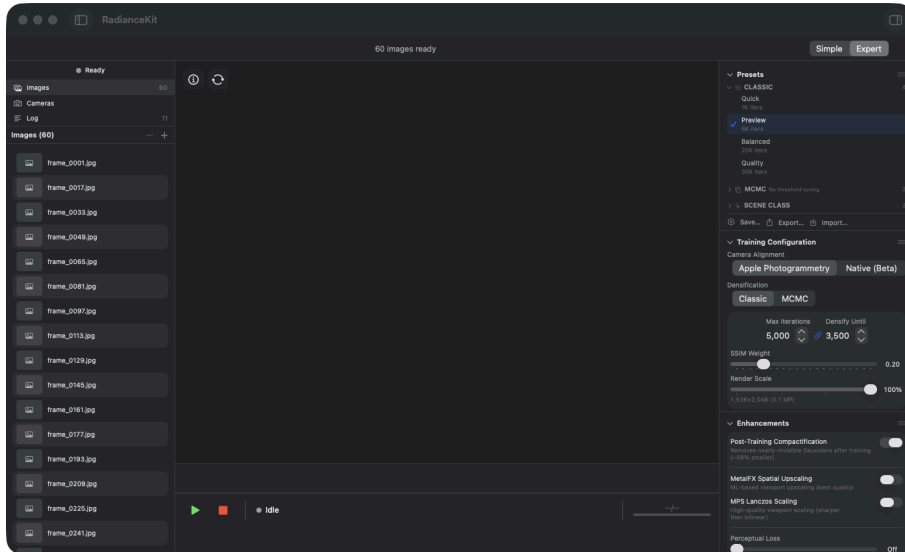
ROZDZIAŁ

Rozdział 2 — Inspector (Expert View)



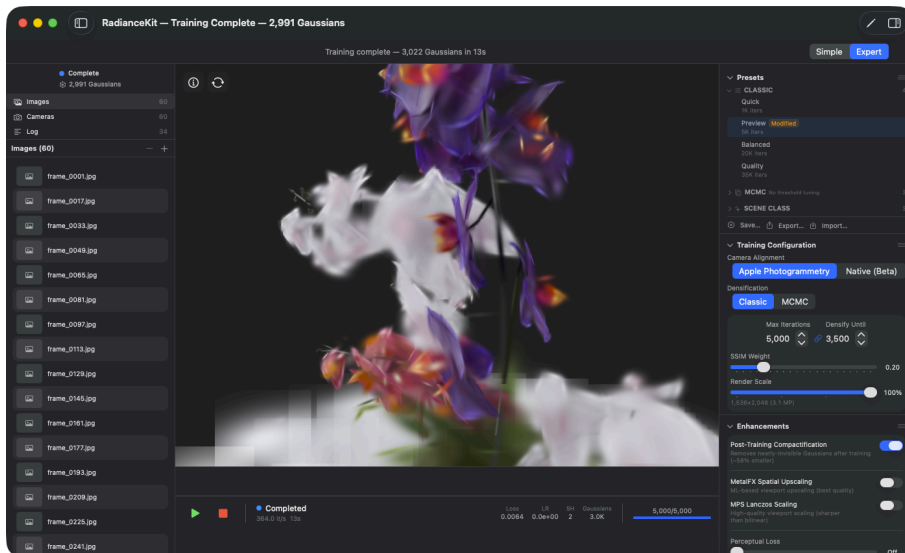
Rysunek 7: Tryb Expert pusty — Project Navigator po lewej (Images 0, Cameras, Log), pusty widok pośrodku, Inspector po prawej z sekcjami Presets/Training Configuration/Enhancements/Training Metrics

Pusty Inspector przed importem: Lewy pasek boczny pokazuje licznik Images 0 i wskazówkę „Drop images here / or tap + to import„. Inspector po prawej jest w pełni funkcjonalny, ale ustawienia wstępne są tylko informacyjne (brak aktywnego treningu). Domyślne ustawienie „Preview” (5K iter.) jest oznaczone. Camera Alignment na Apple Photogrammetry, Densification Classic, SSIM Weight 0.20, Render Scale 50%. Stany puste w Training Metrics („Start training to see live metrics„) i Loss History („Loss curve will appear during training“).



Rysunek 8: Inspector z załadowanymi 60 obrazami flowers — pasek boczny obrazów pokazuje pierwsze nazwy plików frame_0001.jpg ff, nagłówek „60 images ready„

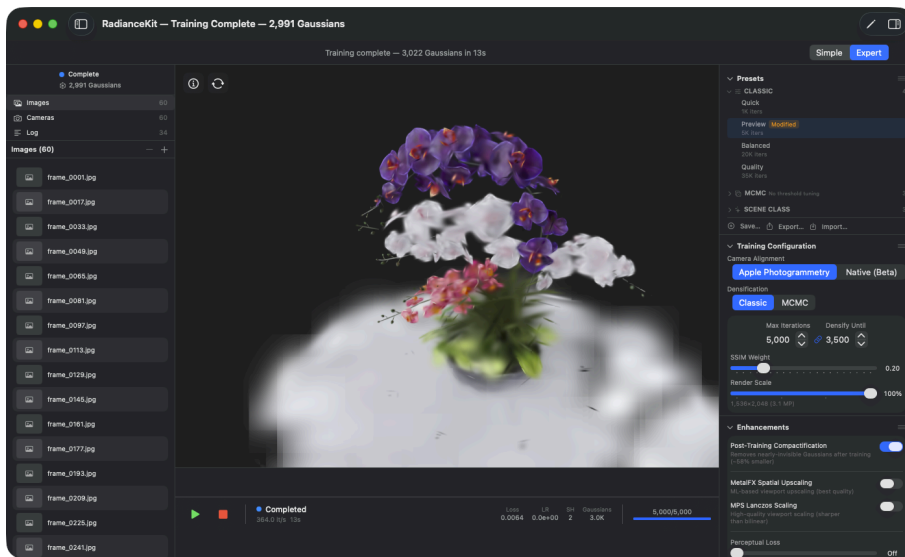
Inspector po imporcie: Status nagłówka „60 images ready„. Pasek obrazów listuje wszystkie 60 zaimportowanych klatek (frame_0001.jpg do frame_0945.jpg , każda 16. klatka z zestawu 960-Cam-Bouquet jako podzbiór do szybkich iteracji). Logika Auto-Render-Scale sprawdza rozdzielczość obrazu ($1536 \times 2048 = 3.1$ MP) i odpowiednio dostosowuje Render Scale. Przycisk Play (zielony, dolny lewy) jest teraz aktywny i startuje trening z aktywnym ustawieniem wstępnym.



Rysunek 9: Inspector w trakcie treningu — Live-Viewport pokazuje rekonstrukcję bukietu flowers, pasek metryk u dołu (Loss / LR / liczba gaussianów / iteracje), karta ustawienia „Preview„ z odznaką „Modified“ jeśli parametry zostały zmienione

Inspector podczas treningu: Pasek tytułu pokazuje globalny postęp „RadianceKit — Training NN %„. Widok renderuje działającą rekonstrukcję gaussianów w czasie rzeczywistym (aktualizowane co 50 iteracji — interwał Live-Preview ustawialny w Settings → General → Training → Live Preview). Pasek metryk pod widokiem: bieżący Loss, Learning Rate, liczba gaussianów i licznik iteracji (np. 1,600/5,000 przy ustawieniu Preview).

Karta ustawienia „Preview” w Inspectorze nosi odznakę „Modified”, gdy tylko jakiś parametr odbiega od domyślnej wartości wbudowanej. Pasek boczny „Log” zbiera zdarzenia etapów SfM i treningu.



Rysunek 10: Inspector po zakończeniu treningu — widok pokazuje gotową rekonstrukcję bukietu flowers (2,991 gaussianów po 5K iteracjach w 13s), pasek tytułu „Training Complete — 2,991 Gaussians,”

Inspector po treningu: Pasek tytułu pokazuje finalną liczbę gaussianów (tutaj 2 991 — bardzo zwarte, ponieważ syntetyczna scena bukietu z Blendera na jasnym tle ma prostą geometrię). Widok pokazuje gotową chmurę punktów — orbitalna nawigacja przeciągnięciem aktywna (obraca wokół środka sceny). Sekcja Training Metrics jest teraz wypełniona wartościami finalnymi, wykres Loss History pokazuje przebieg wszystkich 5 000 iteracji. Sekcja eksportu poniżej jest teraz aktywna (wszystkie przyciski formatów włączone).

Inspector to prawy pasek boczny w Expert Mode (§2). Skupia wszystkie parametry istotne dla treningu w siedmiu zwijalnych sekcjach. Domyślna kolejność z góry na dół przy pierwszym uruchomieniu to: Look, Presets, Konfiguracja treningu, Metryki, wykres straty, Enhancements i Export. Sekcja „Look,” (post-treningowe korekty obrazu) to faktyczna zmiana nazwy w UI dawnej sekcji „Finishing” — jej wewnętrzny `rawValue` enum-a pozostaje ze względów persystencji „Finishing”, a wyświetlany nagłówek brzmi „Look”. Każdą sekcję można zwinać kliknięciem na nagłówek, kolejność zmienić przeciągnięciem (InspectorView.swift:81–97). **Przy pierwszym uruchomieniu wszystkie siedem sekcji jest zwinięte** (inspectorCollapsedSections domyślnie przyjmuje `Set(InspectorSection.allCases)`); stan aplikacji zapisuje potem preferencje zwinięcia i kolejności między uruchomieniami.

Szereg kontrolki z Inspectora pojawia się w niemal identycznej formie również w ustawieniach (rozdział 3) — typowo backend SfM, Sky-Masking i podobne wartości domyślne. Rozdział jest celowy: ustawienia dostarczają globalny szablon aplikacji dla nowotworzonych projektów, Inspector nadpisuje te wartości dla bieżącego otwartego projektu. Kto raz zna logikę obsługi jednej strony, może drugą używać na ślepo.

Lewa kolumna w Expert Mode — Project Navigator — nie należy do Inspectora, ale jest jego bezpośrednim sąsiadem. Tam można wybierać zaimportowane obrazy kliknięciem,

oglądać spacją w Quick-Look i usuwać przyciskiem minus lub klawiszem Delete (z Cmd-Z do cofania). Inspector podąża za bieżącą selekcją paska bocznego z kontekstowymi informacjami szczegółowymi, siedem głównych sekcji pozostaje jednak zawsze dostępne.

Sekcja Look (L1–L5)

Sekcja Look (wewnętrzny `rawValue` nadal „Finishing„) to najwyższa sekcja Inspector'a i zbiera w jednym miejscu **post-treningowe** korekty obrazu. Wszystkie suwaki działają **niedestrukcyjnie**: każdy suwak nakłada `FinishingPass` na nowo na niezmieniony, pierwotny snapshot (oryginalny kolor DC, opacity, skalowanie) — korekta jest więc **idempotentna**, nie kumulatywna. Wynik pojawia się **na żywo w widoku** (WYSIWYG, dokładnie tak, jak późniejszy eksport) i jest **wpiekany w każdy eksport**. Sekcja jest dostępna dopiero **po zakończeniu przebiegu treningu** (wcześniej widnieje „Available after a training run completes.“); jej wartości są **resetowane przy każdym nowym treningu**. Dopóki trwa eksport, wszystkie suwaki są **zablokowane** — pojawia się wskazówka blokady „Locked while exporting — the file uses the current settings.“ i GroupBox jest wyłączony.

L1 Suwak Saturation

GDZIE

Inspector → sekcja Look → GroupBox → Saturation.

TECHNICZNIE

Suwak 0.5–1.2, wyświetlanie dwucyfrowe (np. „1.00“). Skaluje chrominancję SH-DC każdego Splata względem wartości luminancji: 1.0 = bez zmian, < 1.0 = odbarwione (kolor ściągnięty do skali szarości), > 1.0 = bardziej nasycone. Matematycznie kolor DC jest przeliczany wstecz z pierwotnego snapshotu (`desaturateDC`), tak że wielokrotne przesuwanie się nie sumuje. Zwalidowane na materiale z dronów DJI (wiadukt Pensford), który ma tendencję do przerysowania — domyślna wartość dla dronów to 0.82. Działa tylko na bazę koloru (SH-stopień 0), wyższe współczynniki SH pozostają nietknięte.

PROSTYMI SŁOWAMI

Jak nasycone są kolory gotowego Splata. 1.00 zostawia wszystko tak jak wytrenowane, wartości poniżej ściągają kolor w stronę szarości — dobre dla materiału z dronów lub wideo, który często wychodzi przesycony. Wartości powyżej 1.0 czynią go bardziej nasyconym. Możesz dowolnie przesuwac tam i z powrotem, bez „nakręcania się„ efektu, bo aplikacja zawsze przelicza od nowa z niezmienionego oryginalnego stanu. Widoczne na żywo w widoku i dokładnie tak w eksporcie.

L2 Suwak Splat length

GDZIE

Inspector → sekcja Look → GroupBox → Splat length.

TECHNICZNIE

Suwak 0.3–1.0, wyświetlanie dwucyfrowe. Ściąga trzy osie skalowania każdego gaussiana w przestrzeni logarytmicznej w stronę ich wartości średniej (`shortenScale` , współczynnik `alpha`): 1.0 = bez zmian, mniejsze wartości czynią wydłużone „igłowe”, Splaty bardziej okrągłymi, 0 to czyste kule. Działa na igłowe, przeciągnięte Splaty bez zmiany rozmiaru całkowitego i redukuje przez to typowe artefakty „konfetti”. Nakładane od pierwotnego snapshotu (oryginalne skalowanie logarytmiczne), więc idempotentne. Komutuje ze Splat size (L3), bo oba działają w przestrzeni logarytmicznej.

PROSTYMI SŁOWAMI

Czyni przedłużone, drzazgowate Splaty bardziej okrągłymi. 1.00 zostawia kształt tak jak wytrenowany, niższe wartości ściskają wydłużone „igły”, w bardziej okrągławe kleksy — to uspokaja ziarniste, gnębione artefaktami konfetti rekonstrukcje. Rozmiar całkowity pozostaje ten sam, chodzi tylko o wydłużenie. Można bezpiecznie łączyć ze Splat size (L3).

L3 Suwak Splat size

GDZIE

Inspector → sekcja Look → GroupBox → Splat size.

TECHNICZNIE

Suwak 0.5–2.0, wyświetlanie dwucyfrowe. Skaluje każdy gaussian na **wszystkich** trzech osiach jednolicie (`sizeScale`): 1.0 = bez zmian, < 1.0 = mniejsze/gęstsze/ostrzejsze, > 1.0 = większe/„puszystsze”, (wypełnia luki między Splatami). Ponieważ skalowania leżą w przestrzeni logarytmicznej, mnożenie jest realizowane jako addytywny offset `log(factor)` — to komutuje ze Splat length (L2), bo stały offset zostawia odchylenie-od-średniej niezniknięte. Od pierwotnego snapshotu, więc idempotentne. Nowość w tej wersji.

PROSTYMI SŁOWAMI

Skaluje wszystkie Splaty równomiernie większe lub mniejsze. 1.00 to stan wytrenowany, wartości poniżej czynią chmurę punktów ciaśniejszą i ostrzejszą, wartości powyżej zakrywają luki między Splatami (działa miękce/„puszystej”). Praktyczne, by optycznie domknąć dziurawą rekonstrukcję albo odwrotnie odsłonić więcej detalu. Bez problemu współgra ze Splat length (L2) — oba suwaki nie wpływają na siebie nawzajem.

L4 Fade far region (z sub-suwakami)

GDZIE

Inspector → sekcja Look → GroupBox → przełącznik „Fade far region„ plus sub-suwaki „Fade start xradius” i „Fade floor„.

TECHNICZNIE

Przełącznik aktywujący radialny spadek opacyty wraz z dystansem od środka ciężkości kamer — słabo obserwowane „konfetti dali„ w tle są wygaszane. **Tylko dla ujęć orbitalnych:** przełącznik jest wyłączony, gdy `finishingContext.fadeEligible` jest `false` (loty liniowe, zbyt mało lub zdegenerowane kamery); wtedy zamiast sub-suwaków pojawia się wskazówka „Far-fade applies only to orbit captures (not this scene).” Przydatność jest ustalana przez pokrycie azymutalne pozycji kamer (orbita okrąża środek ciężkości i wypełnia wiele sektorów kompasu, lot liniowy tylko ~2). Dwa sub-suwaki sterują geometrią: **Fade start xradius** (1.0–3.0) ustawia promień wewnętrzny jako wielokrotność promienia orbity, w obrębie którego obowiązuje pełna opacyty; **Fade floor** (0.0–1.0) to współczynnik opacyty daleko poza promieniem zaniku. Ważne: zanik **pomija obszar Sky-Dome** (zamrożone gaussiany o indeksach [0, frozenCount]), żeby celowa kopuła tła nie była przyciemniana.

PROSTYMI SŁOWAMI

Wygasza miękkie resztki na zewnętrznej krawędzi sceny — dokładnie te grudki „konfetti dali„, które przy ujęciach dookolnych unoszą się daleko z tyłu. Działa tylko przy prawdziwych ujęciach orbitalnych/okrążających; przy prostych lotach dronem lub zbyt małej liczbie kamer przełącznik jest wyszarzony, a wskazówka wyjaśnia dlaczego. Gdy jest aktywny, dochodzą dwa regulatory precyzyjne: „Fade start xradius” określa, od jakiej odległości (jako wielokrotność promienia okręgu) zaczyna się wygaszanie, „Fade floor„ jak bardzo dalekie Splaty pozostają na końcu jeszcze widoczne (0 = całkiem zniknęły, 1 = bez zmian). Celowo zrekonstruowany Sky-Dome (I44) nigdy nie jest przy tym ruszany — niebo zostaje zachowane.

L5 Przycisk Reset finishing



GDZIE

Inspector → sekcja Look → GroupBox → „Reset finishing„ (na dole, mały przycisk).



TECHNICZNIE

Resetuje wszystkie ustawienia Look do wartości domyślnych (`FinishingPass.Settings() = Saturation 1.0, Fade wyłączony, Splat length 1.0, Splat size 1.0`) i natychmiast wyzwala ponowne finishing, tak że widok wraca do niezmienionego wytrenowanego stanu. `controlSize(.small)`. Ponieważ cały stos Look przelicza idempotentnie od pierwotnego snapshotu, „powrót do domyślnych„ to dokładnie pierwotny wynik treningu — brak utraty jakości przez wielokrotne przesuwanie tam i z powrotem. Jak wszystkie suwaki sekcji zablokowany podczas trwającego eksportu.



PROSTYMI SŁOWAMI

Jednym kliknięciem przywraca wszystkie suwaki Look do standardu (Saturation 1.00, Fade wyłączony, oba suwaki Splat na 1.00) — widok pokazuje potem znów dokładnie świeżo wytrenowany wynik. Praktyczne, gdy się zagrałeś i chcesz czysto zacząć od nowa. Ponieważ aplikacja zawsze przelicza od oryginalnego stanu, nie ma przy tym żadnej utraty jakości. Gdy trwa eksport, przycisk (jak suwaki) jest zablokowany.

Sekcja Presets (I1–I11)

Sekcja Presets to najszybsza droga do zastosowania przetestowanej konfiguracji. Wbudowane ustawienia (Capture Class, Classic, MCMC, Hybrid) dostarczają powtarzalne punkty startowe z 560+ udokumentowanych eksperymentów; własne ustawienia można zapisywać, eksportować, importować i udostępniać. Lista jest pogrupowana według kategorii (Capture Class, Classic, MCMC, Hybrid, Custom), a więcej niż jedna kategoria może być rozwinięta jednocześnie. Przez mechanizm menu kontekstowego (prawy klik na wierszu) dostępne są eksport, duplikowanie i — przy własnych ustawieniach — usuwanie.

I1 Przycisk Save...

GDZIE

Inspector → sekcja Presets → przycisk Save... (pasek akcji na dole).

TECHNICZNIE

Otwiera popover z polem tekstowym i przyciskami Save/Cancel. Bieżący stan TrainingConfig jest persystowany jako nowe ustawienie zdefiniowane przez użytkownika (kodowane JSON, zapisywane na poziomie aplikacji). Proces zapisu kopiuje wszystkie 81 parametrów treningu plus bieżącą strategię densyfikacji. Ustawienie automatycznie łąduje w kategorii Custom, niezależnie od tego, z którego wbudowanego ustawienia zostało wyprowadzone. Puste nazwy i wejścia tylko z białymi znakami są odrzucające. Istniejące już nazwy nie są odrzucające — każde ustawienie ma własny wewnętrzny ID, podwójne nazwy są technicznie dozwolone, ale praktycznie mylące.

PROSTYMI SŁOWAMI

Zabezpiecza twoją bieżącą konfigurację jako wielokrotnie używane ustawienie wstępne. Naciśnij przycisk, podaj w popoverze nazwę i kliknij Save — wszystkie 81 parametrów wraz ze strategią densyfikacji łąduje pod wybraną nazwą w kategorii Custom. Potrzebne, gdy włożyłeś wysiłek i nie chcesz przy następnym projekcie znów zaczynać od zera. Szczególnie praktyczne dla powtarzających się setupów jak „Dron 4K„ lub „Indoor szybko“. Podwójne nazwy są technicznie dozwolone, ale praktycznie mylące — weź lepiej coś mówiącego.

I2 Pole tekstowe Preset Name

GDZIE

Popover Save → pole tekstowe „Preset Name„.

TECHNICZNIE

Proste pole tekstowe z zaokrągloną ramką, szerokie. Wartość jest przejmowana jako nazwa ustawienia przy kliknięciu na przycisk Save. Brak ograniczenia długości w UI, ale zapisana nazwa musi być kodowalna JSON i przedstawialna w listach UI — emoji i znaki diakrytyczne działają. Zawartość jest automatycznie resetowana do pustego stringa przy otwarciu popovera. Przycisk Save pozostaje dezaktywowany, dopóki pole jest puste po przycięciu. Brak Auto-Suggest i brak preset domyślnych z nazwą aktualnie aktywnego ustawienia.

PROSTYMI SŁOWAMI

Tutaj wpisujesz nazwę dla twojego ustawienia. Wybierz coś mówiącego jak „Dron 4K 30fps„ lub „Wnętrze szybko“ — pomaga ci to później w odnajdywaniu w kategorii Custom. Emoji i znaki diakrytyczne dozwolone, twardego limitu długości brak. Dopóki pole jest puste lub składa się tylko ze spacji, przycisk Save pozostaje wyszarzony. Przy ponownym otwarciu popovera pole jest znów puste — brak preset domyślnych z aktywnym ustawieniem.

I3 Przycisk Cancel (Save-Dialog)

GDZIE

Popover Save → przycisk Cancel (po lewej).

TECHNICZNIE

Zamyka popover bez zapisywania. Odrzuca zawartość pola tekstowego — przy następnym otwarciu jest znów resetowany do pustego przez logikę przycisku Save... (I1). Standardowy styl przycisku, brak dialogów potwierdzenia, brak skrótów. Bieżący TrainingConfig pozostaje niezmienny, ponieważ ścieżka Save w ogóle nie została wykonana.

PROSTYMI SŁOWAMI

Zamyka popover Save bez zapisywania czegokolwiek. Jeśli zmieniłeś zdanie, popełniłeś literówkę lub przypadkowo otworzyłeś dialog — po prostu kliknij Cancel. Twoja bieżąca konfiguracja treningu pozostaje niezmienną, ponieważ jeszcze nic nie zostało zapisane. Przy następnym otwarciu popovera pole nazwy startuje znów puste. Brak pytania bezpieczeństwa, brak skrótów — po prostu kliknięcie i zniknęło.

I4 Przycisk Save (Save-Dialog)

GDZIE

Popover Save → przycisk Save (po prawej, prominentny styl).

TECHNICZNIE

Wyzwała właściwe persistowanie. Waliduje jeszcze raz niepustą nazwę (defensywne sprawdzenie) i wtedy zapisuje bieżący TrainingConfig jako JSON do pamięci aplikacji. Następnie zamyka popover. Wyróżniony niebieskim, wyszarzony dopóki pole tekstowe jest puste. Jeśli zapis się nie uda (np. ponieważ pamięć aplikacji jest pełna — bardzo nieprawdopodobne), obecnie nie ma widocznego dialogu błędu; ustawienie po prostu nie pojawiłoby się przy następnym starciu aplikacji.

PROSTYMI SŁOWAMI

Kliknięciem Save przejmujesz nazwę i zapisujesz bieżący setup jako nowe ustawienie wstępne. Popover się zamyka, ustawienie natychmiast pojawia się w kategorii Custom listy ustawień wstępnych i od teraz można je aktywować kliknięciem. Przycisk jest wyróżniony niebieskim (`borderedProminent`) i pozostaje wyszarzony, dopóki pole nazwy jest puste. Jeśli zapis się nie uda (np. UserDefaults pełne), brak widocznego dialogu błędu — ustawienie po prostu zniknęłoby przy następnym starciu aplikacji.

I5 Przycisk Export...

GDZIE

Inspector → sekcja Presets → pasek akcji → przycisk Export...

TECHNICZNIE

Eksportuje aktualnie wybrane ustawienie jako plik `.radiancepreset` (wewnętrznie JSON). Dezaktywowany, gdy nie ma wybranego ustawienia. Przy kliknięciu aplikacja otwiera dialog Save z predefiniowaną nazwą pliku (nazwa ustawienia + rozszerzenie `.radiancepreset`). Zapisany format zawiera kompletny TrainingConfig plus metadane (nazwa, kategoria, ID, flaga Built-in). Podwójne kliknięcie w Finderze otwiera aplikację — ale **nie** automatycznie import; użytkownik musi użyć przycisku Import (I6).

PROSTYMI SŁOWAMI

Wybierz ustawienie z listy i kliknij Export — wtedy możesz je zapisać jako plik `.radiancepreset` i np. wysłać koledze lub przenieść na drugiego Maca. Odbiorca wczytuje go po drugiej stronie przyciskiem Import... (I6). Działa dla wbudowanych i własnych ustawień Custom równie dobrze. Przycisk jest wyszarzony, dopóki w liście nic nie jest kliknięte. Wskazówka: przez menu kontekstowe (I8) idzie jeszcze szybciej — nie musisz ustawienia najpierw zaznaczać.

I6 Przycisk Import...

GDZIE

Inspector → sekcja Presets → pasek akcji → przycisk Import...

TECHNICZNIE

Otwiera dialog pliku dopuszczający tylko pliki `.radiancepreset` (wybór wielu dezaktywowany). Przy wybraniu plik JSON jest ładowany, walidowany i wstawiany do kategorii Custom — z nowym wewnętrznym ID, by nie powstały kolizje z wbudowanymi. Import automatycznie ustawia kategorię na Custom, nawet jeśli wyeksportowane ustawienie pierwotnie było np. wbudowane. Uszkodzone lub niezgodne ze starszą wersją schematu pliki są po cichu odrzucane, bez dialogu błędu (log konsoli daje jednak informację).

PROSTYMI SŁOWAMI

Wczytuje plik `.radiancepreset` z dysku. Użyteczne, gdy ktoś przesyła ci wypróbowany setup lub sam chcesz utrzymać swoje ulubione ustawienia w synchronizacji na kilku Macach. Zaimportowane ustawienia zawsze lądują w kategorii Custom — nawet gdy pierwotnie były eksportowane z wbudowanych. Uszkodzone lub przestarzałe pliki są po cichu ignorowane; w logu konsoli stoi wtedy powód. Wybór wielu w dialogu jest dezaktywowany, więc na kliknięcie tylko jeden plik.

I7 Wiersz ustawienia (aktywacja kliknięciem)

GDZIE

Inspector → sekcja Presets → każdy wiersz ustawienia w każdej kategorii.

TECHNICZNIE

Kliknięcie na wiersz ustawienia zastępuje wszystkie pola TrainingConfig wartościami z ustawienia, zapamiętuje ID aktywnego ustawienia i resetuje status Modified. Aktywne ptaszki przed wierszem pojawiają się tylko, gdy ustawienie jest wybrane i niezmodyfikowane. Gdy tylko wartość w TrainingConfig zostanie zmieniona (suwak, stepper, przełącznik w innych sekcjach Inspektora), za nazwą pojawia się pomarańczowa odznaka „Modified„. Wbudowane ustawienia nie mogą być nadpisywane — przy modyfikacji trzeba przez przycisk Save (I1) założyć własną kopię.

PROSTYMI SŁOWAMI

Kliknięcie na wiersz aktywuje ustawienie i przejmuje wszystkie zapisane tam wartości do bieżących ustawień treningu. Ptaszek przed nazwą pokazuje, które ustawienie jest obecnie aktywne. Gdy tylko potem przekręcisz jakiś suwak, stepper lub przełącznik w innych sekcjach, za nazwą pojawia się pomarańczowa odznaka „Modified„ — ponieważ twój setup teraz odbiega od ustawienia. Wbudowanych ustawień nie da się nadpisać; jeśli chcesz zachować zmiany, załóż przez przycisk Save... (I1) własną kopię lub zduplikuj ustawienie (I9).

I8 Menu kontekstowe „Export...„

GDZIE

Prawy klik na każdy wiersz ustawienia → pierwsza pozycja „Export...„.

TECHNICZNIE

Identyczna funkcjonalność jak I5 (przycisk Export...), ale wygodniej dostępna — bez konieczności wcześniejszego zaznaczenia ustawienia. Eksportuje bezpośrednio ustawienie klikniętego wiersza. Działa dla wszystkich kategorii ustawień równo (wbudowane lub Custom), bez ograniczenia. Eksport zawiera flagę Built-in i oryginalną kategorię, ale przy reimportcie kategoria jest mapowana na Custom, jak opisano w I6.

PROSTYMI SŁOWAMI

Szybka droga do eksportu — prawy klik na pożądane ustawienie i wybierz „Export...„. Oszczędza obejście przez wcześniejsze kliknięcie i potem naciśnięcie przycisku Export. Działa dla wszystkich kategorii równo, również dla wbudowanych. Utworzony plik `.radiancepreset` jest identyczny z tym z I5; przy późniejszym reimportcie ładuje automatycznie w kategorii Custom.

I9 Menu kontekstowe „Duplicate„

GDZIE

Prawy klik na każdy wiersz ustawienia → druga pozycja „Duplicate„.

TECHNICZNIE

Klonuje ustawienie do kategorii Custom. Tworzy nowy wewnętrzny ID, dokleja „ Copy„ do nazwy i zapisuje kopię. Działa także dla wbudowanych ustawień — klon jest wtedy edytowalny. Oryginał pozostaje nietknięty. TrainingConfig jest kopiowany wartość po wartości (JSON roundtrip), tak by nie istniały referencyjne powiązania między oryginałem a kopią.

PROSTYMI SŁOWAMI

Tworzy edytowalną kopię ustawienia w kategorii Custom. Praktyczne, gdy np. chcesz wbudowane ustawienie „Quality„ jako punkt wyjścia, a potem chcesz tylko nieco przesunąć suwak SSIM. Workflow: duplikuj, zmień nazwę (menu kontekstowe lub nowy przebieg Save...), dostosuj, gotowe. Oryginał pozostaje nietknięty — możesz w każdej chwili do niego wrócić. Działa także dla wbudowanych, co jest jedyną drogą przejęcia ich wartości jako bazy i jednocześnie uczynienia ich edytowalnymi.

I10 Menu kontekstowe „Delete„

GDZIE

Prawy klik na własne wiersze ustawień → ostatnia pozycja „Delete„ (czerwony, destrukcyjny).

TECHNICZNIE

Widoczna tylko dla ustawień Custom. Wbudowanych nie można usuwać. Pozycja jest oznaczona jako destrukcyjna, pojawia się w menu kontekstowym na czerwono i jest oddzielona dzielnikiem, aby nie klikać jej przypadkowo. Nie ma **żadnego** dialogu potwierdzenia — jedno kliknięcie usuwa ustawienie natychmiast. Usunięte ustawienie nie jest do odzyskania (Cmd-Z tu nie działa — Undo istnieje w bieżącym buildzie tylko dla listy obrazów, nie dla operacji na ustawieniach). Jeśli usunięte ustawienie było właśnie aktywne, bieżący TrainingConfig pozostaje niezmienny, tylko aktywny wybór ustawienia jest wyzerowany.

PROSTYMI SŁOWAMI

Usuwanie własnych ustawień. Przy wbudowanych (Quick, Preview, Balanced, Quality, Ultra Detail, Drone / Aerial, 360° Walkaround, Photo / Object itd.) „Delete„ w ogóle nie jest widoczne — nie możesz ich przypadkowo zabić. Uwaga: brak pytania bezpieczeństwa i brak Undo, kliknięcie i ustawienie znika. Jeśli nie jesteś pewien, wcześniej przez Export... (I5/I8) pociągnij na dysk kopię bezpieczeństwa — możesz ją w każdej chwili ponownie zaimportować. Jeśli ustawienie było właśnie aktywne, twój TrainingConfig pozostaje niezmienny, znika tylko ptaszek.

I11 Nagłówek kategorii (zwijanie/rozwijanie)

GDZIE

Inspector → sekcja Presets → każdy nagłówek kategorii (Capture Class, Classic, MCMC, Hybrid, Custom).

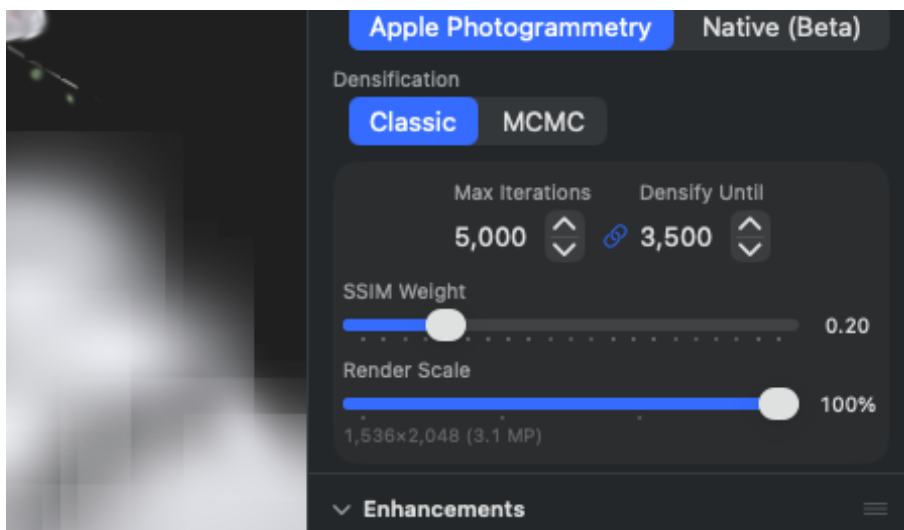
TECHNICZNIE

Status zwinięcia na kategorię z różnymi domyślnymi: kuratowana grupa Capture Class startuje **rozwinięte**, Classic, MCMC, Hybrid i Custom startują **zwinione**. Status nie jest persistowany — przy restarcie aplikacji wszystkie kategorie są znów w stanie domyślnym. Strzałka chevron rotuje z animacją. Liczba po prawej w nagłówku pokazuje liczbę ustawień w tej kategorii. Obszar trafienia kliknięcia obejmuje cały obszar nagłówka.

PROSTYMI SŁOWAMI

Zwijanie i rozwijanie kategorii, aby utrzymać listę ustawień przejrzystą. Przy starcie aplikacji grupa Capture Class jest otwarta, Classic, MCMC, Hybrid i Custom zamknięte. Kliknięcie na nagłówek (cały obszar jest klikalny) i lista wjeżdża z krótką animacją chevron na górę lub dół. Mała liczba po prawej pokazuje, ile ustawień leży w kategorii. Po restarcie aplikacji jest znów stan domyślny — aplikacja celowo nie zapisuje tego ustawienia zwinięcia.

Sekcja Konfiguracja treningu (I12–I22)



Rysunek 11: Wycinek sekcji Konfiguracja treningu — Camera Alignment (Apple Photogrammetry aktywne, Native (Beta) nieaktywne), Densification (Classic aktywne), Max Iterations 5,000 / Densify Until 3,500 z symbolem Link, suwak SSIM Weight 0.20, suwak Render Scale na 100 % (1,536×2,048 = 3.1 MP)

Tu łądzą centralne dźwignie: który backend SfM ma liczyć, jak działa densyfikacja, ile iteracji, jak duża waga SSIM. Przy strategii MCMC pojawiają się dwa dodatkowe przełączniki („MCMC Quality„ i „Auto-scale by scene“), ukryte w trybie Classic. Przy backendzie Native SfM dochodzi pole FOV-Override, używane tylko dla klatek wideo bez ogniskowej EXIF.

I12 Selektor Camera Alignment

GDZIE

Inspector → Konfiguracja treningu → Camera Alignment (selektor segmentowy u góry).

TECHNICZNIE

Selektor segmentowy z dwiema opcjami: Apple Photogrammetry i Native (Beta). Wybór określa używany backend SfM przy następnej rekonstrukcji kamer. Wpływa jednocześnie na to, które inne elementy Inspector'a są widoczne: Native pokazuje dodatkowo FOV-Override (I13), potrzebny tylko przy klatkach wideo bez EXIF. Wskazówka: dla bardzo dużych ujęć plenerowych możesz wczytać wynik zewnętrznego narzędzia (Metashape lub COLMAP) przez import workspace — zobacz rozdział 1 (M5) i rozdział 9 (Q3, Q6).

PROSTYMI SŁOWAMI

Tutaj wybierasz, jak rekonstruowane są pozycje kamer — najważniejszy przełącznik dla końcowej jakości. Apple Photogrammetry to szybki standard i wystarcza w pełni dla większości skanów obiektów. Native (Beta) to własne zgodne z App Store rozwiązanie, dobre dla orbit i scen obrotowych, a przy klatkach wideo bez EXIF potrzebuje FOV-Override (I13). Przy bardzo dużych zestawach plenerowych możesz alternatywnie policzyc kamery w Metashape lub COLMAP i wczytać wynik przez import workspace. Szczegóły i rekomendacje na typ sceny znajdziesz w rozdziale 9.

I13 Pole FOV Override (Native SfM)

GDZIE

Inspector → Konfiguracja treningu → FOV Override (widoczne tylko przy Camera Alignment = Native).

TECHNICZNIE

Numeryczne pole tekstowe (zakres 0-170°), domyślnie 0 = automatyczne ustalenie z EXIF lub heurystyki. Ręczne wpisanie jest potrzebne, gdy obrazy wejściowe zostały wyekstrahowane z wideo bez metadanych ogniskowej. Typowe wartości: iPhone Wide $\approx 73^\circ$, DJI Mavic Wide-Crop $\approx 70^\circ$, dron z sensorem pełnoklatkowym $\approx 84^\circ$. Wartość jest ograniczana do [0, 170] — wartości poza są bezpośrednio cofane. Wpływa tylko na natywny potok SfM (Q4/Q5); Apple Photogrammetry kompletnie ignoruje tę wartość.

PROSTYMI SŁOWAMI

Jeśli twoje obrazy nie mają EXIF (typowo przy wyekstrahowanych klatkach wideo), wpisujesz tutaj poziomy zakres widzenia kamery w stopniach. Wartości orientacyjne: iPhone Wide $\approx 73^\circ$, DJI Mavic Wide-Crop $\approx 70^\circ$, dron z sensorem pełnoklatkowym $\approx 84^\circ$. 0 pozwala aplikacji samej zgadnąć — często działa dobrze, ale przy rzadkich obiektywach może pójść źle. Wartości powyżej 170° są automatycznie cofane. Pole jest widoczne i działa tylko, gdy wybrałeś Native jako Camera Alignment (I12) — Apple Photogrammetry kompletnie je ignoruje.

I15 Selektor Densification

GDZIE

Inspector → Konfiguracja treningu → Densification (selektor segmentowy, zawsze widoczny).

TECHNICZNIE

Przełącza między dwiema strategiami densyfikacji: Classic (oryginalna procedura 3DGS z Clone/Split/Prune i progiem gradientu) i MCMC (Stochastic Gradient Langevin Dynamics z relokacją, NeurIPS 2024). Przy zmianie z Classic na MCMC aplikacja automatycznie ustawia specyficzne dla MCMC pola na sprawdzone wartości domyślne (Reg-Weights = 0, MCMC-Cap-Multiplier 3.0, harmonogram Sample/Noise). Bez tej automatycznej inicjalizacji sesje ze starymi ustawieniami cierpiały na błąd 1.4.4-MCMC-Collapse (460K→5 gaussianów, kill watchdoga). Wybór selektora dodatkowo określa, które elementy Inspektora są widoczne — przy MCMC pojawiają się I16/I17. Szczegółowe działanie pól w rozdziale 6, T11–T16 (Classic) i T61–T73 (MCMC).

PROSTYMI SŁOWAMI

Centralny wybór strategii dla rośnięcia liczby gaussianów. Classic jest dobrze dostrojone z 459 eksperymentów, generuje szybkie i wysokiej jakości wyniki i nie musi znać pól MCMC. MCMC to nowsze podejście (NeurIPS 2024), bardziej powtarzalne i rezygnuje z ręcznej regulacji progów — za to liczy około 6x dłużej przy porównywalnej jakości. Przy przełączaniu na MCMC aplikacja automatycznie ustawia bezpieczne domyślne, aby trening nie wszedł w 1.4.4-Collapse. Szczegóły do pól strategii stoją w rozdziale 6 (T11–T16 Classic, T61–T73 MCMC).

I16 Przełącznik MCMC Quality

GDZIE

Inspector → Konfiguracja treningu → MCMC Quality (tylko przy Densification = MCMC).

TECHNICZNIE

Przełącza Gradient-Accumulation na 2 kroki (aktywne) lub 1 krok (nieaktywne). Akumuluje gradienty z dwóch następujących po sobie widoków kamery, zanim wykonany zostanie krok optimizera. Empirycznie (Session 33, V544a) redukuje to finalny błąd L1 o ok. 6% (0.0246 z Quality vs 0.0261 bez, przy uśrednieniu 3 prób na Horse-Full-MCMC). Cena: podwojony czas treningu. Przy bardzo długich treningach (200K iteracji) prowadzi to do dodatkowych 10+ minut oczekiwania — więc opłaca się tylko, gdy ostatnie parę procent jakości naprawdę jest potrzebne. Wpływa tylko na trening, nie na format eksportu czy prezentację widoku.

PROSTYMI SŁOWAMI

Tryb Quality dla MCMC z Gradient-Accumulation po dwóch widokach. Robi wynik końcowy empirycznie ok. 6% lepszym (L1 0.0246 zamiast 0.0261 w teście Horse), kosztuje za to dwa razy tyle. Jeśli i tak już jedziesz trening 200K-MCMC (bywa 2 godziny), dochodzi znowu szczupła godzina. Opłaca się przy finalnych renderingach showcase lub pod koniec sesji Quality-Sweep, w codziennym workflow raczej nie. Widoczne tylko, gdy Densification stoi na MCMC (I15).

I17 Przełącznik Auto-scale by scene

GDZIE

Inspector → Konfiguracja treningu → Auto-scale by scene (tylko przy MCMC).

TECHNICZNIE

Gdy aktywny, skaluje efektywną górną granicę Max-Gaussians ze SfM-Init-Point-Count × MCMC-Cap-Multiplier (domyślnie 3.0). Przykład: SfM dostarcza 250K punktów początkowych, podstawowy cap = 150K, multiplier 3.0 → efektywna górna granica = $\max(150K, 750K) = 750K$. Gdy dezaktywowany, ściśle obowiązuje tylko podstawowy. Wprowadzone dla v1.4.5, ponieważ duże ujęcia plenerowe z ponad 1000 klatkami i odpowiednio wysoką gęstością punktów SfM przy sztywnym domyślnym 150K-Cap zagłodziły densyfikację — nadmiarowe punkty zostały, nowe nie mogły powstać. Domyślnie OFF w ustawieniach Custom, ON w wbudowanych MCMC. Wpływa tylko na czas treningu, nie na eksport.

PROSTYMI SŁOWAMI

Pozwala maksymalnej liczbie gaussianów rosnąć z rozmiarem sceny (dokładniej: z liczbą punktów inicjalnych SfM). W małych scenach prawie nie zauważasz różnicy, w dużych scenach plenerowych jest często decydujące dla jakości — inaczej trening „dusi się”, ponieważ domyślna górna granica 150K jest dla sceny zbyt niska. Wprowadzone specjalnie dla v1.4.5, gdy bardzo duże zestawy plenerowe (ponad 1000 klatek) wyraźnie wisiły na cap. Przy wbudowanych ustawieniach MCMC już z góry włączone; w własnych ustawieniach domyślnie wyłączone.

I18 Stepper Max Iterations

GDZIE

Inspector → Konfiguracja treningu → GroupBox → Max Iterations.

TECHNICZNIE

Stepper z zakresem 1 000–100 000, krok 1 000. Określa łączną liczbę iteracji optimizera. Liniowo skorelowane z czasem treningu (połowa = ok. 50% czasu). Empiryczne sweet spoty: 20K (Classic Balanced, $L1 \approx 0.028$), 40K (Classic Quality, $L1 \approx 0.023$), 200K (MCMC Full, $L1 \approx 0.0246$). Powyżej 40K przy Classic średnio daje zaledwie poprawę — Diminishing Returns. Przy zmianie, jeśli funkcja Link (I19) jest aktywna, Densify Until jest proporcjonalnie podciągany (domyślny stosunek: 0.5, tj. Densify-Until = $\text{Max}/2$).

PROSTYMI SŁOWAMI

Ile kroków treningu jest wykonywanych — więcej jest lepiej, ale kosztuje też liniowo więcej czasu. Reguła kciuka: 20 000 dla dobrej jakości, 40 000 dla optimum przy strategii Classic (powyżej średnio prawie nic nie daje). MCMC potrzebuje znacznie więcej, 200 000 jest tu standardem. Podwojenie iteracji z grubsza podwaja czas treningu. Przy aktywnym przycisku Link (I19) Densify Until jest proporcjonalnie podciągany — praktycznie zawsze to, czego chcesz.

I19 Przycisk Link/Unlink (Densify ↔ Iterations)

GDZIE

Inspector → Konfiguracja treningu → GroupBox → mały przycisk Link między Max Iterations a Densify Until.

TECHNICZNIE

Przycisk przełączający, który zamraża stosunek Densify Until do Max Iterations. Przy aktywnym (ikona Link wyróżniona) przy każdej zmianie Max Iterations Densify Until jest proporcjonalnie podciągany. Przy Unlink (ikona Link-Plus) wartości pozostają niezależne. Domyślnie linked, ponieważ odzwierciedla to typową korelację — gdy ciągniesz trening na podwójne iteracje, zwykle chcesz też densyfikację proporcjonalnie dłużej. Stosunek jest obliczany przy ustawieniu przycisku Link z bieżącej wartości; typowy stosunek to 0.5 (Densify-Until = połowa liczby iteracji).

PROSTYMI SŁOWAMI

Mały przycisk klamry między Max Iterations a Densify Until. Gdy aktywny (ikona Link wyróżniona), obie wartości wędrują razem — podwajasz Iterations, podwaja się też Densify Until w tym samym stosunku. Gdy nie (ikona `link.badge.plus`), możesz je ustawiać niezależnie. Standardem jest zlinkowane, ponieważ odzwierciedla to typową korelację — dłuższy trening zwykle też chce dłuższej fazy densyfikacji. W 99% przypadków zostaw zatrzaśnięte.

I20 Stepper Densify Until

GDZIE

Inspector → Konfiguracja treningu → GroupBox → Densify Until.

TECHNICZNIE

Stepper z zakresem 500–50 000, krok 500. Określa indeks iteracji, od którego nie dochodzą nowe gaussiany przez Clone/Split (Classic) lub relokację (MCMC). Po osiągnięciu udoskonalane są tylko pozycja i kolor. Wyższe wartości = więcej gaussianów = większy plik, dłuższy czas na iterację (+30–60% GPU-Time na krok). Typowe wartości: 15K (dla 30K Max-Iter), 20K (dla 40K), 100K (dla 200K MCMC). Przy aktywnym Link (I19) automatycznie skalowane razem. Inaczej wpływa przy Classic vs MCMC: Classic zatrzymuje całkowicie wzrost, MCMC zatrzymuje logikę relokacji, ale adaptacja Sample/Noise działa dalej.

PROSTYMI SŁOWAMI

Do której iteracji wolno dodawać nowe gaussiany — przy Classic przez Clone/Split, przy MCMC przez relokację. Potem chodzi już tylko o udoskonalenie koloru i formy istniejących punktów. Wyżej = więcej detalu, ale też większy plik i +30–60% czasu GPU na krok. Typowe wartości: 15K (dla 30K Max-Iter), 20K (dla 40K), 100K (dla 200K MCMC). Wisi normalnie przez Link (I19) na Max Iterations — rzadko sensowne ręczne oddzielanie.

I21 Suwak SSIM Weight

GDZIE

Inspector → Konfiguracja treningu → GroupBox → SSIM Weight.

TECHNICZNIE

Suwak 0.0–1.0 w krokach 0.05, wyświetlanie jako „0.20”. Miesza L1-Loss (0.0) i SSIM-Loss (1.0). L1 napina jasność na piksel, SSIM podobieństwo strukturalne (krawędzie, lokalne statystyki). Domyślnie 0.2 to wartość z oryginalnego artykułu 3DGS (Kerbl 2023) i reverse-engineered jako solidny kompromis w licznych sesjach. Wyższe wartości (0.5+) preferują zachowanie detalu, mogą jednak ignorować lokalne błędy jasności. Niższe wartości (< 0.1) prowadzą do utraty detalu na ostrych krawędziach. Obliczanie SSIM działa w shaderze z oknem Gaussa 11×11. Wydajność: przy 0.0 (tylko L1) trening jest ok. 8-12% szybszy, ponieważ obliczanie SSIM w shaderze jest pomijane.

PROSTYMI SŁOWAMI

Jak silnie ważne jest podobieństwo strukturalne obrazu (krawędzie, lokalne wzory) wobec czystego porównania jasności. 0.2 to standard z oryginalnego artykułu 3DGS i wystarcza prawie dla wszystkich scen. Wyżej (0.5+) przy drobnych strukturach jak włosy, futro lub roślinność — tam pomaga większa waga struktury. Niżej (0.0) sprawia, że trening jest ok. 8-12% szybszy, ponieważ obliczanie SSIM w shaderze jest pomijane, ale kosztuje detal na ostrych krawędziach. Kto nie ma dobrego powodu do zmiany, zostawia 0.2.

I22 Suwak Render Scale

GDZIE

Inspector → Konfiguracja treningu → GroupBox → Render Scale.

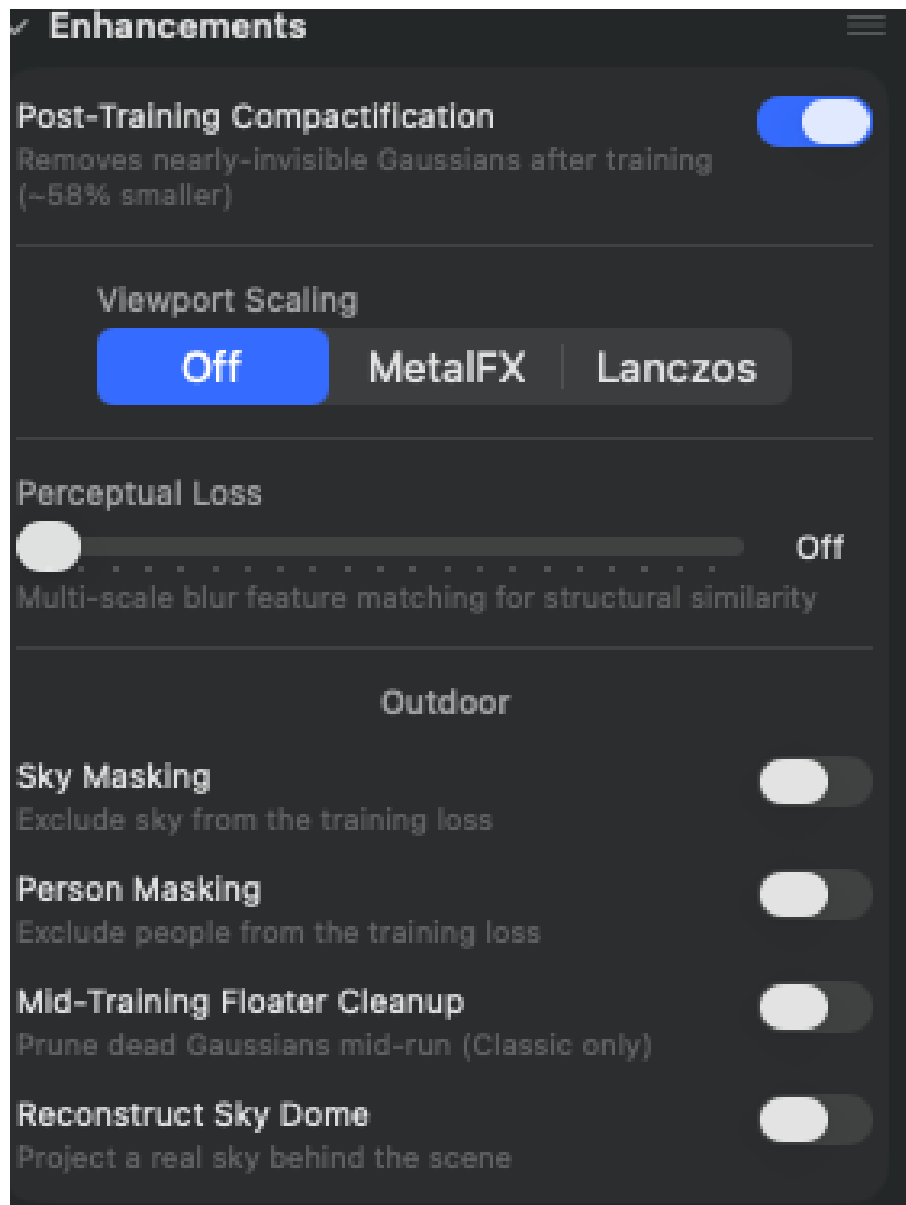
TECHNICZNIE

Suwak 0.25–1.0 w krokach 0.25, wyświetlanie jako „100%”. Skaluje rozdzielczość renderingu treningu względem wielkości obrazu źródłowego. Największa dźwignia na wydajność: 50% redukuje czas GPU o ok. 75% (ponieważ 4× mniej pikseli), 25% o ok. 94%. Próg gradientu jest automatycznie skalowany razem. Pod suwakiem pojawia się wyświetlacz rozdzielczości na żywo w MP (np. „2304×1296 (3.0 MP)”). Jeśli bieżąca wartość odbiega od rekomendowanej, w pomarańczowym tekście wyświetla się „— recommended: 50%”. Rekomendacja celuje w ~3 MP efektywnej rozdzielczości — najbardziej efektywnie przetwarzalny zakres przez GPU Apple Silicon. Obrazy źródłowe 4K dostają np. automatycznie rekomendację 25%, obrazy FullHD 100%. Zmiana wyzwała dodatkowo realokację bufora.

PROSTYMI SŁOWAMI

Z jaką rozdzielczością trening renderuje — jedna z największych dźwigni wydajności. Pełna (100%) daje najlepszą jakość, ale przy dużych obrazach kosztuje dużo czasu GPU. Połowa (50%) oszczędza około 75% czasu GPU, ponieważ liczone jest cztery razy mniej pikseli — idealne dla źródeł 4K. Pod suwakiem widzisz efektywną rozdzielczość w megapikselach; aplikacja celuje w około 3 MP, ponieważ to na Apple Silicon działa najefektywniej. Jeśli twoja wartość odbiega od tego, aplikacja wyświetla pomarańczową wskazówkę „recommended”, — zwykle opłaca się za tym podążać.

Sekcja Enhancements (I26–I29, I42–I44)



Rysunek 12: Wycinek sekcji Enhancements — trzy wiersze: Post-Training Compactification (przełącznik włączony), Viewport Scaling (segmentowany picker Off/MetalFX/Lanczos), Perceptual Loss (suwak na „Off”). Każdy wiersz z podtytułem wyjaśnia funkcję

Sekcja Enhancements grupuje trzy funkcje poprawiające jakość obrazu bez zmieniania samej rdzeniowej pętli treningu. Pierwsze dwie (I26–I27) to etapy **Post-Training** lub **Viewport**: Compactification sprzęta po końcu treningu, Viewport Scaling to czysty renderer widoku, który nie wpływa na działający trening. Perceptual Loss (I29) mimo przynależności do sekcji jest składnikiem treningu — jest aktywowane podczas treningu jako dodatkowy człon loss, stąd oddzielenie od przełączników widoku przez dzielnik. Od wersji v1.6 sekcja ma też grupę Outdoor (I42–I44: Sky Masking, Mid-Training Floater Cleanup, Reconstruct Sky Dome) — opcje treningowe przeciwko floaterom nieba, które dawniej mieszkały w oknie ustawień, a teraz siedzą tutaj per projekt.

I26 Przełącznik Post-Training Compactification **GDZIE**

Inspector → Enhancements → Post-Training Compactification.

 **TECHNICZNIE**

Aktywuje post-processing V443: po zakończeniu iteracji treningu gaussiany z opacności poniżej 0.01 (1% widoczności) są usuwane. Empirycznie redukuje to rozmiar pliku o ~55-58% przy zerowej widocznej utracie jakości — ponieważ te gaussiany wizualnie i tak nie wnoszą. Compactification działa jako GPU-Compact-Pass i trwa zależnie od liczby gaussianów ułamki sekund do kilku sekund. Nie wpływa na wydajność treningu. Gdy ten przełącznik jest wyłączony, eksportowane są również niewidoczne gaussiany — istotne tylko, gdy chcesz użyć formatu do dalszego etapu treningu (Continue Training), inaczej marnotrawstwo pamięci.

 **PROSTYMI SŁOWAMI**

Sprząta po treningu gaussiany, których i tak nie widzisz (opacności poniżej 1%). Robi pliki eksportu o około połowę mniejszymi (55-58% redukcji rozmiaru) bez widocznej utraci jakości. Działa jako krótki przebieg GPU po ostatniej iteracji, trwa tylko ułamki sekund do kilku sekund. Praktycznie zawsze powinno być włączone — jedyny powód do wyłączenia to gdy chcesz później kontynuować trening przez Continue Training i musisz zachować również niewidoczne gaussiany. Przy normalnych przepływach eksportu po prostu zostaw włączone.

I27 Picker Viewport Scaling **GDZIE**

Inspector → Enhancements → Viewport Scaling (segmentowany picker z trzema opcjami: Off, MetalFX, Lanczos).

 **TECHNICZNIE**

Jeden segmentowany picker wybierający upscaler widoku — trzy opcje są **wzajemnie wykluczające**. Gdy rozdzielczość treningu (przez I22 Render Scale) jest niższa niż wielkość widoku, wybrany tryb skaluje wyrenderowaną klatkę do wielkości wyświetlania. **Off** = proste rozciąganie bilinearne. **MetalFX** = bazowany na ML Spatial Upscaler od Apple, najostrożniejsza opcja (model ML zoptymalizowany na ostre krawędzie), narzut ok. 1-2ms na klatkę na GPU M3. **Lanczos** = Apple Metal Performance Shaders z 8-Tap-Sinc-resamplingiem, klasycznie bez ML, minimalny narzut (< 0.5ms), jakość poniżej MetalFX, ale bez typowego dla ML „wygładzania” drobnych struktur liniowych. Potok renderera jest przy przełączaniu na żywo rekonfigurowany — widoczne natychmiast, bez restartu. **Tło**: Wcześniej były to dwa osobne przełączniki (MetalFX + Lanczos), które mogły być włączone jednocześnie — stan sprzeczny, w którym MetalFX po cichu wygrywał z Lanczosem. Picker usuwa ten stan; ewentualnie odziedziczony ze starszych sesji stan „oba włączone” sam się leczy przy następnym przełączeniu do MetalFX. Wpływa **tylko** na widok na żywo, nie na renderowane eksporty (Orbit-Video, zrzuty ekranu) — te są renderowane w pełnej rozdzielczości źródła.

 **PROSTYMI SŁOWAMI**

Wyostrza obraz na żywo w widoku — szczególnie użyteczne, gdy pracujesz ze zredukowaną rozdzielczością treningu (Render Scale 50%, zobacz I22). Trzy stopnie, z których zawsze aktywny jest tylko jeden: „Off”, po prostu rozciąga piksele, „MetalFX” używa machine learningu Apple dla najostrzejszych krawędzi (praktycznie zawsze najlepszy wybór), „Lanczos”, to klasyczny filtr bez ML — weź go jako fallback, jeśli MetalFX wygładza ci w jakiejś scenie linie lub pokazuje artefakty. Działa na żywo, bez restartu. Wpływa tylko na widok na żywo, nie na eksportowane wideo orbitalne ani zrzuty ekranu — te są renderowane w pełnej rozdzielczości źródła. Inaczej niż wcześniej nie możesz już przez pomyłkę wybrać dwóch trybów jednocześnie.

I29 Suwak Perceptual Loss**GDZIE**

Inspector → Enhancements → Perceptual Loss.

TECHNICZNIE

Suwak 0.0–0.2 w krokach 0.01, wyświetlanie przy 0.0 jako „Off”, inaczej jako „0.05” itd. Aktywuje dodatkowy człon loss, porównujący multi-skalowany Gaussian-Blur renderingu z obrazem ground-truth (3 skale rozmycia). Wychwytuje strukturalne różnice, których L1+SSIM samo nie widzi. Implementacja V460. Empirycznie wartość 0.05–0.1 poprawia ocenę L1 w sesjach o kilka procent, kosztuje jednak ~5% czasu treningu (dodatkowy forward-pass przez jądra rozmycia). Powyżej 0.15 trening staje się niestabilny i L1 znowu się pogarsza (człon loss dominuje optymalizację). Działa **podczas** treningu, nie w post-processingu — mimo pozycji w sekcji „Enhancements”, to nie jest więc czysta wartość dodatkowa post-factum.

PROSTYMI SŁOWAMI

Dodatkowy udział loss sprawdzający strukturalne podobieństwo obrazu przez trzy różne stopnie rozmycia. Pomaga szczególnie w scenach z drobnymi strukturami jak włosy, tkanina lub roślinność, ponieważ wychwytuje wzory, których L1+SSIM samo nie widzi. Mniejsze wartości są bezpieczniejsze — 0.05 do 0.1 to sweet spot, powyżej 0.15 trening staje się niestabilny i loss znowu się pogarsza. Na 0 (Off) funkcja jest całkowicie wyłączona i nic nie kosztuje; aktywna pochłania ok. 5% czasu treningu na dodatkowy forward-pass przez jądra rozmycia. Działa mimo sekcji „Enhancements”, bezpośrednio podczas treningu, nie dopiero w post-processingu.

I42 Sky Masking GDZIE

Inspector → Enhancements (grupa Outdoor) → przełącznik „Sky Masking„. Bound: `AppState.trainingConfig.skyMaskingEnabled` (per projekt, `@DefaultFalse`). Default: `false` .

 TECHNICZNIE

Aktywuje pre-trainingową segmentację pikseli nieba opartą na Apple Vision. Przed startem treningu dla każdej kamery wejściowej obszar nieba jest ekstrahowany przez maskę pierwszego planu Apple Vision (Sky = Background) i przypisywany jako maska na piksel do danej kamery. Podczas treningu wkład w loss na piksel jest mnożony przez dopełnienie maski nieba — piksele nieba wnoszą 0 do gradientu, więc gaussiany rzutujące w niebo nie otrzymują sygnałów optymalizacji i tym samym nie stają się „gęstsze„ ani „jaśniejsze“. Znacząco redukuje floatery (ciemne grudki na niebie) w scenach plenerowych/dronowych. Kosztuje ~3% regresji L1 przy klasycznym treningu 40K (zobacz `memory/dev_outdoor-floater-reduction.md`). Sensowne tylko w scenach plenerowych z wyraźnie widocznym niebem; w scenach wewnątrz lub z białym tłem segmentacja nieba identyfikuje błędne obszary i blokuje ważne sygnały loss. Wartość jest teraz przechowywana per projekt (już nie globalnie dla aplikacji) i podąża za ustawieniem wstępnym / plikiem sceny.

 PROSTYMI SŁOWAMI

Przy ujęciach plenerowych z niebem w kadrze powstają często czarne lub kolorowe grudki na niebie — tzw. „floatery„. Ta opcja automatycznie wykrywa, gdzie jest niebo, i mówi treningowi: „Zostaw niebo w spokoju.“ Działa bardzo dobrze przy lotach dronem i scenach krajozobrazowych. Przy wnętrzach lub ciemnych tłach może pogorszyć obraz — więc włączaj tylko, gdy widać prawdziwe niebo. Szczegóły: `memory/dev_outdoor-floater-reduction.md`.

I43 Mid-Training Floater Cleanup

GDZIE

Inspector → Enhancements (grupa Outdoor) → przełącznik „Mid-Training Floater Cleanup„. Bound: `AppState.trainingConfig.floaterCleanupEnabled` (per projekt, `@DefaultFalse`). Default: `false`.

TECHNICZNIE

Włącza przy treningu Classic-40K (ustawienie „P4 Quality„) dwa dodatkowe przebiegi kontroli gęstości: przy iteracji 20,000 i przy iteracji 30,000. Oba przebiegi przeszukują wszystkie gaussiany według trzech kryteriów: (a) bardzo niskie opacity (standard 0.005), (b) niewielki rozmiar w przestrzeni ekranu, (c) brak wkładów do loss w ostatnich 1000 iteracjach. Gaussiany spełniające wszystkie trzy warunki są usuwane. Efekt: ~5–15% mniej gaussianów na końcu treningu, widocznie mniej ciemnych grudek na niebie w scenach dronowych/plenerowych. Kosztuje ~1–3% regresji L1 w scenach wewnątrz close-up, dlatego nie aktywowane domyślnie. Wartość jest zapamiętywana między restartami (w przeciwieństwie do S7). Dwie iteracje cleanupu (20K, 30K) są twardo zdefiniowane i obecnie nie można ich zmienić przez UI; przy krótszych treningach (np. P2 Preview 5K) przełącznik nie ma efektu, ponieważ nigdy nie dochodzi do oznaczeń iteracji. **Nowość:** przełącznik jest teraz operowalny tylko wtedy, gdy aktywne ustawienie używa densyfikatora **Classic** (`densificationStrategy == .classic`). Przy MCMC lub Hybrid jest **disabled** i pojawia się wskazówka inline, ponieważ te strategie i tak same obsługują martwe gaussiany (MCMC przez relokację, Hybrid przez połączoną logikę reloc/noise) — manualne przebiegi cleanupu byłyby tam nieskuteczne lub kontrproduktywne. Referencja w kodzie: `RadianceKitApp.swift`, karta General. Szczegóły: `memory/dev_outdoor-floater-reduction.md`.

PROSTYMI SŁOWAMI

Podczas treningu powstają czasem „martwe„ punkty gaussowskie, które nic już nie wnoszą do jakości obrazu, ale zajmują pamięć. Ta opcja sprząta dwukrotnie podczas długiego treningu (przy 20K i 30K iteracji) i usuwa te zwłoki. W scenach plenerowych z niebem jest to szczególnie sensowne, ponieważ tam zbiera się większość floaterów. Przy małych treningach lub zbliżeniach mebli raczej niepotrzebne. Przełącznik da się włączyć tylko wtedy, gdy twoje ustawienie używa densyfikatora Classic — przy ustawieniach MCMC lub Hybrid jest wyszarzony (wraz z krótkim wyjaśnieniem), ponieważ one same sprzątają swoje martwe punkty.

I 44 Reconstruct Sky Dome**GDZIE**

Inspector → Enhancements (grupa Outdoor) → przełącznik „Reconstruct Sky Dome„. Bound: `AppState.trainingConfig.skyDomeEnabled` (per projekt, `@DefaultFalse`). Default: `false` .

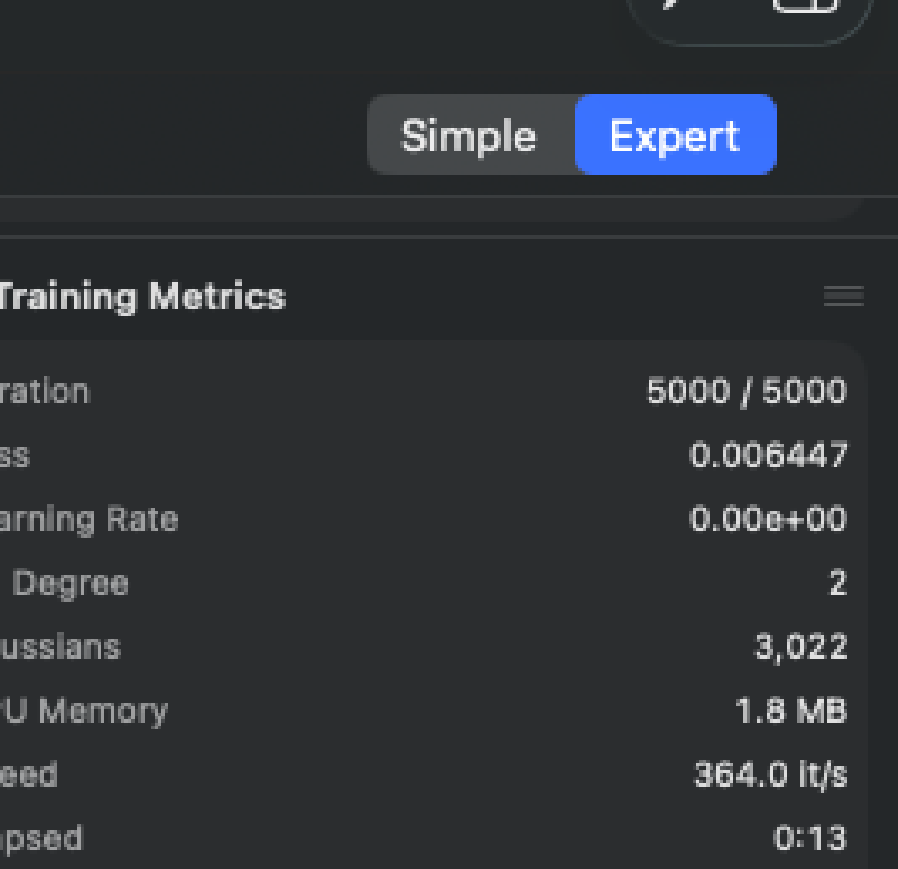
TECHNICZNIE

Aktywuje projekcję Sky-Dome przed treningiem (MVP V549e). Po SfM i przed startem treningu dla każdej kamery wejściowej maska nieba Apple Vision wspólna z S7 jest ekstrahowana z obrazu, piksele nieba są ze współczynnikami wewnętrznymi kamery odrzutowywane na powierzchnię wirtualnej kuli (standardowy promień 8× promień sceny). Na tej kuli inicjalizowanych jest ~5000 nowych gaussianów ze średnimi kolorami z odrzutowanych pikseli nieba, bardzo dużą skalą (1.0 w jednostkach sceny) i początkową opacnością 0.95. Te 5000 gaussianów to nie jest klasyczna maska nieba — są trenowane jak wszystkie inne, ale przez wysoką początkową opacność pozostają w cienkiej skorupie. Wynik: przy widokach Novel-View 360° w scenach plenerowych/dronowych zamiast ciemnych grudek confetti pojawiają się faktyczne kolory nieba i struktury chmur. Wartość jest zapamiętywana między restartami. Sensowne tylko w scenach plenerowych z co najmniej 360° pokrycia kamer; przy czystych Object-Capture bez widoku nieba nie ma efektu. Status: eksperymentalne, szersza walidacja A/B na dalszych zestawach plenerowych jeszcze przed nami.

PROSTYMI SŁOWAMI

Zamiast tego, by trening próbował „zgadnąć„ niebo z kilku widocznych pikseli (co prowadzi do floaterów), RadianceKit rzutuje piksele nieba bezpośrednio na wirtualną kulę wokół sceny przed startem treningu. Gdy potem obrócisz gotową scenę w 360°, widzisz prawdziwe niebo zamiast czarnych grudek. Działa tylko przy ujęciach plenerowych, gdzie faktycznie niebo jest w kadrze. Przy skanach salonu lub ujęciach studyjnych nic nie daje.

Sekcja Metryki (I30–I38)



The screenshot shows the 'Training Metrics' section in the Inspector (Expert View). At the top, there are two tabs: 'Simple' and 'Expert', with 'Expert' selected. Below the tabs, the 'Training Metrics' section is displayed, showing a list of metrics and their values. The metrics are: Iteration (5000 / 5000), Loss (0.006447), Learning Rate (0.00e+00), Degree (2), Gaussians (3,022), GPU Memory (1.8 MB), Speed (364.0 It/s), and Elapsed (0:13).

Metric	Value
Iteration	5000 / 5000
Loss	0.006447
Learning Rate	0.00e+00
Degree	2
Gaussians	3,022
GPU Memory	1.8 MB
Speed	364.0 It/s
Elapsed	0:13

Rysunek 13: Wycinek sekcji Training Metrics po zakończonym treningu na bukcie (5K iteracji, 2 991 gaussianów finalnie) — tabela z metrykami treningu (Iteration, Loss, SSIM Loss, Combined Loss, Gaussian Count, Learning Rate, Elapsed, ETA)

Podczas działania treningu sekcja metryk pokazuje dziewięć wartości na żywo z pętli treningu. Przed startem treningu sekcja jest pusta („Start training to see live metrics”). Wszystkie wartości są aktualizowane co ~30 iteracji (częstotliwość aktualizacji strumienia). Sekcja jest read-only — żaden element nie jest klikalny ani zmienialny. Do głębszej analizy weź logi JSONL treningu z `~/Documents/RadianceKit/Logs/` (skrypt `python3 scripts/analize_logs.py best 5`).

I30 Iteration



GDZIE

Inspector → Metryki → Iteration. Read-only.

TECHNICZNI

Wyświetlanie w formacie „4523 / 40000” — bieżąca iteracja przez łączną planowaną. Liczy synchronicznie z pętlą treningu, która pushuje wartości co ~30 iter. Druga liczba odpowiada wartości Max-Iterations w momencie startu; nie zmienia się więcej, nawet gdy użytkownik przekręci stepper — działający przebieg używa własnej kopii migawki. Gdy aplikacja przez menu Training dosypuje iteracji (Continue Training +5K/+10K/+20K), mianownik się zwiększa.

PROSTYMI SŁOWAMI

Gdzie trening obecnie stoi. „4523 / 40000” oznacza: 4523 z 40 000 kroków przeszło, więc około 11%. Lewa liczba odlicza w takcie sekundowym; gdy stoi przez minuty, trening zawiesił się — zazwyczaj wskazówka na GPU-Throttling lub konkurencyjną aplikację. Prawa liczba odpowiada wartości Max-Iterations (I18) przy starcie treningu i nie zmienia się więcej, nawet gdy później przekręcisz stepper. Przy Continue Training (+5K/+10K/+20K) rośnie o dodatkowe kroki.

I31 Loss



GDZIE

Inspector → Metryki → Loss. Read-only.

TECHNICZNI

Wartość float z sześcioma miejscami po przecinku (np. „0.024385”). Mierzy kombinowany Loss L1+SSIM (mix kontrolowany przez I21 SSIM Weight) plus opcjonalnie Perceptual Loss (I29) i inne regularizery. Skala nie jest absolutna, lecz zależna od sceny — wymaga dla większości porównań tego samego zestawu danych. Typowe wartości końcowe przy dobrych konfiguracjach: - Classic Quality 40K iter: 0.022–0.025 (Horse, Truck, Garden) - MCMC Full 200K iter: 0.024–0.028 - Outdoor Dron 30K: 0.030–0.060 (gorsze z powodu geometrii) - Indoor Apartments: 0.018–0.025

Wartości powyżej 0.10 po 5K iteracjach wskazują na problemy SfM (złe pozy kamer) — przerwij i policz SfM od nowa.

PROSTYMI SŁOWAMI

Jak bardzo wyrenderowany obraz odbiega jeszcze od oryginału — kombinacja z L1, SSIM i ewentualnie Perceptual Loss. Mniejsze jest lepiej. Poniżej 0.03 jest zwykle naprawdę dobrze, poniżej 0.05 jeszcze okej, sceny plenerowe leżą z powodu geometrii raczej przy 0.03–0.06. Powyżej 0.10 po kilku tysiącach iteracji to sygnał ostrzegawczy — zwykle przyczyna leży w rekonstrukcji kamer (SfM nie zadziałało czyisto). Skala nie jest absolutna, lecz zależna od sceny; porównania robić tylko w ramach tego samego zestawu danych. Gdy liczba nagle skacze w górę, zwykle wydarzyło się zdarzenie eksplozji gradientu.

I32 Learning Rate



GDZIE

Inspector → Metryki → Learning Rate. Read-only.



TECHNICZNIE

Wyświetlanie w notacji naukowej (np. „1.60e-04”). Bieżący współczynnik uczenia dla parametrów pozycji (3DGS ma sześć niezależnych LR dla pozycji, SH-DC, SH-Rest, opacity, skali, rotacji — wyświetlany jest tu LR pozycji jako wartość reprezentatywna). Domyślna wartość początkowa 1.6e-4, opadająca przez Exponential-Decay do ~1.6e-6 na końcu treningu. Spadek można dostosować przez pole LR-Schedule w konfiguracji treningu (pole T w rozdz. 6). Gdy LR pozostaje niezwykle wysokie (np. 1e-3 lub więcej po 10K iteracji), może to wskazywać na błędnie załadowaną konfigurację.

PROSTYMI SŁOWAMI

Jak duże są obecnie kroki optymalizacji — konkretnie współczynnik uczenia dla pozycji gaussianów. Startuje przy 1.60e-04 i opada wykładniczo do około 1.60e-06 na końcu treningu („1.60e-06” = 0.0000016). Przebieg działa automatycznie, niczego tu nie musisz dostrajać. Gdy wartość po 10 000+ iteracjach jest wciąż większa niż 1e-3, prawdopodobnie została załadowana błędna config — przerwij trening i wybierz ustawienie wstępne na nowo. Wewnętrznie 3DGS ma sześć niezależnych współczynników uczenia (pozycja, SH-DC, SH-Rest, opacity, skala, rotacja); tutaj widzisz tylko LR pozycji jako zastępcę.

I33 SH Degree



GDZIE

Inspector → Metryki → SH Degree. Read-only.



TECHNICZNIE

Liczba całkowita 0-3. Stopień Spherical Harmonics dla reprezentacji koloru. Zaczyna od 0 (tylko komponenta DC, tj. niezależny od kierunku kolor na gaussiana — czyli tylko stała RGB) i rośnie progresywnie podczas treningu do 3. Standardowy harmonogram podnosi stopień przy 1000/2000/3000 iteracjach o po

1. SH-3 odpowiada 48 współczynnikom koloru na gaussiana (3

kanały RGB × 16 funkcji bazowych SH). Wyższy stopień SH = więcej zależnego od kierunku odbicia (błyszczące powierzchnie pod różnymi kątami widzenia wyglądają poprawnie różnie), ale też więcej pamięci i wolniejszy trening.

PROSTYMI SŁOWAMI

Jak złożona jest obecnie reprezentacja koloru na gaussiana. Startuje przy 0 (tylko jeden niezależny od kierunku kolor na punkt) i jest stopniowo podciągana na 3 — typowo przy iteracji 1000, 2000 i 3000. Stopień 3 oznacza 48 współczynników koloru na gaussiana i pozwala na zależne od kierunku odbicia, czyli że błyszczące powierzchnie wyglądają poprawnie różnie z różnych kątów widzenia. Nie musisz aktywnie ruszać, harmonogram działa automatycznie. Wyższy stopień kosztuje więcej pamięci i lekko spowalnia trening — ale to cena za realistyczne refleksy.

I34 Gaussians

GDZIE

Inspector → Metryki → Gaussians. Read-only.

TECHNICZNIE

Bieżąca liczba gaussianów w modelu, sformatowana z separatorem Locale (np. „524.318”). Wzrost: - Classic: startuje od punktów początkowych SfM (typowo 50K-300K), rośnie przez Clone/Split aż krótko przed Densify Until, potem statyczne do końca treningu (modulo Pruning) - MCMC: punkty Sample są dodawane do MCMC-Cap, potem tylko relokacja

Zdrowe wartości końcowe: - Classic Quality: 400K-700K (Horse 524K, Garden 800K) - MCMC Full: dokładnie na cap (domyślnie 150K, z Auto-Scale Multiplier × SfM-Count zależnie od sceny 500K-1.5M)

Przy MCMC liczba spada do < 60% cap → anomalia (wskaźnik kolapsu, wskazuje na zbyt agresywne regularizery).

PROSTYMI SŁOWAMI

Ile punktów gaussianów ma obecnie model 3D. Rośnie podczas treningu, aż osiągnięte zostanie Densify Until (I20); potem liczba pozostaje praktycznie stała. Więcej punktów = więcej detalu, ale też większy plik i wolniejsze renderowanie w widoku. 500.000 gaussianów to typowa średnia wartość dla Classic-Quality na średniej scenie; MCMC Full łąduje w zależności od Auto-Scale (I17) przy 500K do 1.5M. Gdy liczba przy MCMC nagle spada poniżej 60% cap, to wskaźnik kolapsu — zwykle zbyt agresywne regularizery.

I35 GPU Memory

GDZIE

Inspector → Metryki → GPU Memory. Read-only.

TECHNICZNIE

Oszacowanie zużycia pamięci bufora gaussianów jako liczba gaussianów × 616 bajtów (sformatowane w stylu Memory). 616 bajtów to empiryczny rozmiar w pełni wyposażonego gaussiana (pozycja, skala, rotacja, opacity, współczynniki SH stopień 3, akumulator gradientu). Wyświetlanie **nie** obejmuje narzutu renderera (bufor kafelków, bufor sortowania, bufor backward) — realne zapotrzebowanie na pamięć GPU leży typowo 2-3x powyżej tej wartości. Przy 500K gaussianów: wyświetlane ~290 MB, realnie ~700 MB. Przy 1.5M gaussianów: wyświetlane ~880 MB, realnie ~2.5 GB. Na M3 Max z 64+ GB Unified Memory niekrytyczne, na M3 Pro z 18 GB już limit.

PROSTYMI SŁOWAMI

Oszacowanie, ile pamięci GPU zajmują same gaussiany — około 616 bajtów na punkt. Faktyczne zużycie GPU jest 2-3x wyższe niż wyświetlane, ponieważ renderer dokłada jeszcze własne bufory kafelków, sortowania i backward. Przy MacBooku z 16-18 GB Unified Memory powinieneś pozostać poniżej 500K gaussianów; z M3 Max lub Studio (64+ GB) możesz luźno jechać 1.5M i więcej. Gdy trening nagle się wywala lub system swapuje, zwykle granica jest tu osiągnięta — Render Scale (I22) w dół lub Densify Until (I20) zredukować.

I36 Speed GDZIE

Inspector → Metryki → Speed. Read-only.

 TECHNICZNIE

Iteracje na sekundę z jednym miejscem po przecinku („24.3 it/s”). Obliczone przez trener jako średnia krocząca z ostatnich ~100 iteracji. Typowe wartości:

- Quick Preset (1K iter): 80-120 it/s (krótko, brak steady-state)
- Classic 20K @ 1.0 Render Scale (scena Truck, M3 Max): 25-35 it/s
- Classic 20K @ 0.5 Render Scale: 80-120 it/s
- MCMC 200K @ 0.5 Render Scale: 25-50 it/s (wolniejsze z powodu relokacji)
- Przy 1M+ gaussianów i pełnej rozdzielczości: < 10 it/s

Spadająca prędkość w trakcie treningu jest normalna — więcej gaussianów = więcej compute na iterację. Nagłe spadki (np. z 30 → 5 it/s) wskazują na GPU-Thermal-Throttling lub konkurujące aplikacje.

 PROSTYMI SŁOWAMI

Jak szybko trening działa, w iteracjach na sekundę. Stoi typowo przy 20-50 it/s, przy zredukowanej Render Scale (50%) i małych scenach również czasem 80-120 it/s. Spada w trakcie treningu zupełnie naturalnie, ponieważ więcej gaussianów = więcej pracy na iterację. Nagłe spadki (np. 30 → 5 it/s) wskazują na GPU-Thermal-Throttling lub konkurujące aplikacje — karty przeglądarki z wideo, backup Time Machine, indeksowanie Photos. Trzymanie aplikacji na pierwszym planie i zamykanie programów w tle często pomaga. Przy 1M+ gaussianów i pełnej rozdzielczości poniżej 10 it/s są normalne.

I37 Elapsed GDZIE

Inspector → Metryki → Elapsed. Read-only.

 TECHNICZNIE

Już upłynęły czas jako „4:23” (m:ss) lub „1:23:45” (h:mm:ss). Przełączenie formatu od 1 godziny. Mierzy tylko czysty czas treningu, nie poprzedzające fazy (obliczenia SfM, import obrazów). Przy Pause/Resume zegar idzie dalej — jest więc wall-clock, nie czas CPU.

 PROSTYMI SŁOWAMI

Jak długo trening już działa, jako czysty stoper (czas wall-clock). Format to „m:ss”, do jednej godziny, potem „h:mm:ss”. Nie „czas CPU”, tylko „jak długo już czekamy” — więc też czasy paury się liczą. Mierzy tylko czystą fazę treningu, nie poprzedzające obliczenia SfM ani import obrazów. Pomocne do porównania z ETA (I38) — gdy Elapsed znacząco wystrzela poza pierwotne ETA, trening gdzieś stał się wolniejszy niż planowane.

I38 ETA**GDZIE**

Inspector → Metryki → ETA. Read-only.

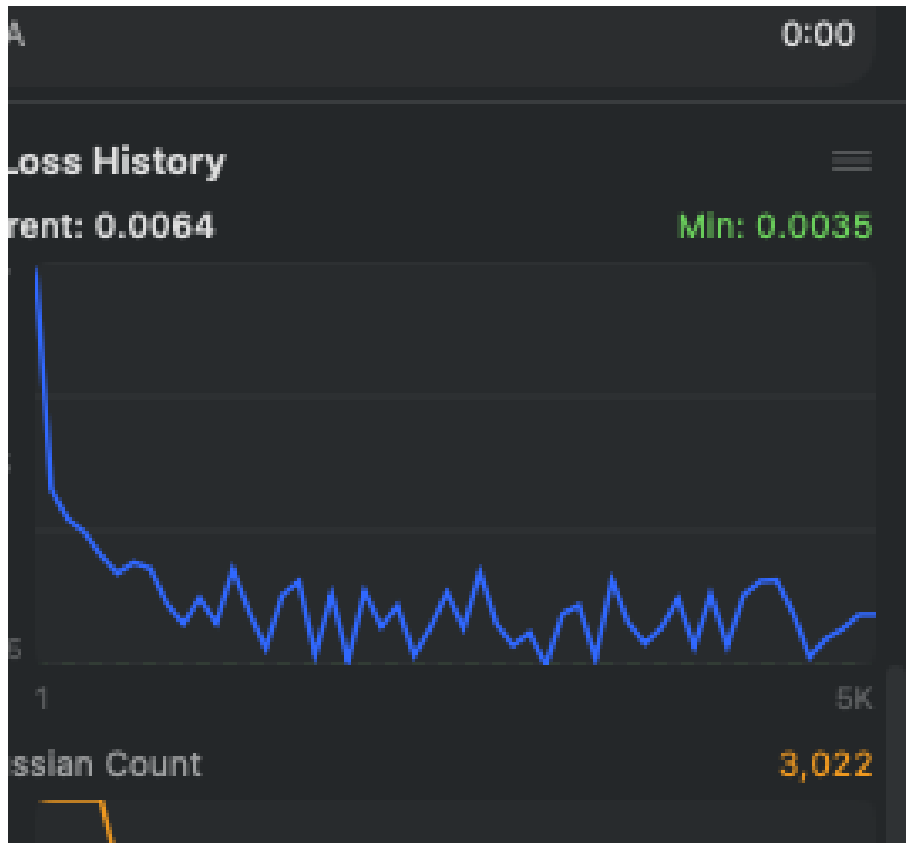
**TECHNICZNIE**

Szacowany czas pozostały jako „17:42” lub „1:12:35”. Obliczenie: (Max Iterations – bieżąca iteracja) / iteracje na sekundę. Pokazuje „-”, gdy prędkość jest właśnie zero (zupełnie na początku lub przy pauzie). Oszacowanie **nie** jest dostosowywane do typowego spowolnienia pod koniec treningu — zwłaszcza przy MCMC i Classic z dużymi wartościami Densify-Until trening ma tendencję stawać się wolniejszym, ponieważ coraz więcej gaussiów wchodzi do obrazu. Realnie pozostaje typowo 10-20% powyżej początkowego ETA.

**PROSTYMI SŁOWAMI**

Jak długo prawdopodobnie jeszcze trzeba czekać — obliczone z pozostałych iteracji i bieżącej prędkości (I36). Zgrubne oszacowanie: gdy Mac nagle staje się wolniejszy (więcej gaussiów od fazy densyfikacji, Thermal-Throttling, inne aplikacje), może trwać dłużej niż wyświetlane. Aplikacja nie wlicza typowego spowolnienia pod koniec treningu, dlatego prawdziwy koniec ląduje zwykle 10-20% powyżej początkowego ETA. Doliczyć plus 15%, wtedy zwykle pasuje. Pokazuje „-”, gdy prędkość jest właśnie 0 (początek treningu lub pauza).

Sekcja wykresu straty (I39–I41)



Rysunek 14: Wycinek sekcji Loss History po zakończonym treningu — Current 0.0064, Min 0.0035 (zielony), niebieski przebieg z 0.027 (iteracja 1) do 0.0035 (iteracja 5K) z charakterystycznym zagięciem ok. iter 200, pod spodem wykres Gaussian Count pomarańczowy

Sekcja wykresu straty wizualizuje przebieg treningu w czasie. Składa się z dwóch wykresów: wykresu krzywej Loss (duży, u góry, niebieski) i wykresu liczby gaussianów (mniejszy, na dole, pomarańczowy). Oba są budowane na żywo podczas treningu i persistują do następnego startu treningu. Przed pierwszym treningiem obszar jest pusty („Loss curve will appear during training”). Wykresy są czystymi rysunkami SwiftUI-Path (brak frameworku Swift-Charts), aby również przy 100K+ punktach renderowały się płynnie.

I39 Current Loss (wyświetlanie)

GDZIE

Inspector → wykres straty → lewy obszar etykiety „Current: 0.0287”. Read-only.

TECHNICZNIE

Wartość float ostatniego punktu próbki Loss, sformatowana z czterema miejscami po przecinku. Identyczna z I31 (Loss w sekcji Metrics), tylko sformatowana tu bardziej zwarto. Źródłem jest historia Loss — lista, która otrzymuje wpis na ~30 iteracji. Tylko wartości skończone są wpisywane na listę — NaN/Infinity (bardzo rzadkie, w przypadku błędu eksplozji gradientu) są filtrowane.

PROSTYMI SŁOWAMI

Bieżąca wartość Loss w krótszym zapisie niż w sekcji Metrics (cztery miejsca po przecinku). Treściowo identyczna z I31, ale tutaj wyświetlanie siedzi bezpośrednio przy wykresie Loss i daje ci przy obserwacji krzywej dokładną wartość liczbową. Aktualizowane jak wszystkie metryki na żywo co 30 iteracji. Wartości NaN lub Infinity (skrajnie rzadkie przy błędach eksplozji gradientu) aplikacja automatycznie odfiltrowuje. Użyteczne, aby przy patrzeniu na wykres nie musieć skakać do innej sekcji.

I40 Min Loss (wyświetlanie)

GDZIE

Inspector → wykres straty → prawy obszar etykiety „Min: 0.0245” (zielony). Read-only.

TECHNICZNIE

Minimum wszystkich kiedykolwiek widzianych wartości Loss bieżącego przebiegu treningu. Rekomputowane na żywo z historii Loss — brak osobnej persistencji. Wyświetlane zielonym tekstem, ponieważ „Min„ = „Best so far“. Przerywana zielona linia na dolnej krawędzi wykresu wizualnie zaznacza tę pozycję Y. Przy sesjach Continue-Training śledzenie minimum startuje od nowa — stara historia jest w UI zastępowana nową (nie doczepiana). Gdy bieżący trening idzie gorzej niż poprzedni, wyświetlanie Min może więc być większe niż poprzedni wynik końcowy.

PROSTYMI SŁOWAMI

Najniższa wartość Loss, którą ten trening dotychczas widział — wyświetlana zielonym, ponieważ „best so far„. Przerywana zielona linia na dolnej krawędzi wykresu zaznacza tę pozycję również wizualnie. Gdy bieżąca krzywa leży właśnie wyraźnie powyżej, jest szansa jeszcze poprawy; zwykle Min jest jednak wskaźnikiem wyniku końcowego, który cię później interesuje. Przy sesjach Continue-Training śledzenie Min startuje od nowa, ponieważ stara historia w UI jest zastępowana nową — wartość Min może przez to wyglądać gorzej niż poprzedni wynik końcowy.

I41 Wykres Gaussian Count

GDZIE

Inspector → wykres straty → drugi wykres pod spodem (pomarańczowy). Read-only.

TECHNICZNIE

Wykres liniowy liczby gaussianów w iteracjach treningu. Źródło: historia Gaussian-Count (lista par (Iter, Count), wypełniana przez trener co ~30 iter). Skala Y dynamiczna między minimum a maksimum historii. Przy strategii Classic krzywa wygląda zwykle tak: stale rośnie do Densify Until, potem płasko (z małymi wahaniami pruningu). Przy MCMC: stromy wzrost do cap, potem horyzontalna linia (relokacja utrzymuje liczbę stałą). Gdy krzywa **opada** mimo aktywnego treningu, densyfikacja zbyt agresywnie pruninguje — wskazówka na błędne domyślne lub znany błąd MCMC-Collapse (temat v1.4.4-Hotfix).

PROSTYMI SŁOWAMI

Jak liczba gaussianów rozwija się w czasie treningu — mniejszy pomarańczowy wykres pod krzywą Loss. Przy strategii Classic linia stale rośnie, aż osiągnięte zostanie Densify Until (I20), potem pozostaje płasko z małymi wahaniami pruningu. Przy MCMC strzela stromo do cap i pozostaje horyzontalna, ponieważ relokacja utrzymuje liczbę stałą. Gdy krzywa mimo aktywnego treningu nagle zagina się w dół, densyfikacja zbyt agresywnie pruninguje — klasyczna oznaka błędu MCMC-Collapse z v1.4.4. Wtedy pomaga aktualizacja aplikacji lub przełączenie z powrotem na Classic.

Jak czytać krzywą Loss?

Wykres Loss jest najważniejszym narzędziem diagnostycznym w Inspectorze — żaden inny wskaźnik nie pokazuje tak bezpośrednio, czy trening posuwa się użytecznie do przodu, czy zawisł. Typowa zdrowa forma to szybki spadek w pierwszych 1000-3000 iteracji (z ~0.15 na ~0.05), po którym następuje powolny, równomierny spadek do końca treningu (na 0.020-0.030). Logarytmicznie krzywa wygląda przy tym jak gładka przekątna.

Co oznacza plateau przy Loss? Gdy krzywa pozostaje płaska przez kilka tysięcy iteracji, są dwie możliwe lektury: (a) trening jest „zbieżny”, — Loss nie może już znacząco spadać, ponieważ model jest tak dobry, jak może być z danymi danych i ustawieniami. Jest to pożądane; to „gotowe”. (b) trening „wisi”, — Loss mógłby właściwie jeszcze spadać, ale optymalizacja stagnuje (minimum lokalne, współczynnik uczenia za mały, densyfikacja wyłączona). Rozróżnić: gdy wartość Loss leży w typowym dobrym zakresie (0.020-0.030 przy Indoor/Object, 0.040-0.060 przy plenerze) i krzywa od 5K iteracji jest płaska, jest zbieżne. Gdy wartość jest wyraźnie wyższa niż przy porównywalnych scenach (np. 0.08), zawisła.

Uwaga plateau gaussianów ≠ plateau Loss. Plateau w liczbie gaussianów **nie** oznacza „trening jest gotowy”. Oznacza tylko, że densyfikacja przestała dodawać nowe punkty — albo dlatego, że osiągnięto (Classic), albo dlatego, że cap MCMC jest pełny. Trening działa potem dalej i udoskonala już tylko istniejące punkty. Właściwy sygnał „gotowe” czytasz na krzywej Loss i wskaźniku iteracji (I30), nie tutaj.

Reguła kciuka do przerywania: gdy krzywa Loss po 5000+ iteracjach leży powyżej 0.08 i prawie już nie spada, z wysokim prawdopodobieństwem rekonstrukcja SfM jest skrzy-

wiona. Przerwij trening, w rozdziale 9 sprawdź, czy wybrany backend SfM pasuje do sceny, ew. przełącz na COLMAP/Native, potem startuj na nowo. Lepiej zainwestować 10 minut w lepsze SfM niż 2 godziny treningu ze złym wyrównaniem kamer.

Kiedy sięgnąć po Inspector?

Szybka referencja: która sekcja + które kontrolki dla którego typowego use-case?

Typowe zadanie	Sekcja	ID kontrolek
Odbarwić kolory gotowego Splata	Look	L1 (Saturation)
Zaokrąglić igłowe/konfetti Splaty	Look	L2 (Splat length)
Wypełnić dziurawą chmurę / powiększyć Splaty	Look	L3 (Splat size)
Wygasić dalekie „konfetti daleki”, przy orbitach	Look	L4 (Fade far region)
Odrzucić korekty Look	Look	L5 (Reset finishing)
Wczytać predefiniowany setup	Presets	I7 (klik na wiersz)
Zapisać własny setup	Presets	I1 → I2 → I4
Udostępnić setup kolegom	Presets	I5 (Export) lub I6 (Import)
Zmienić backend SfM (np. ponieważ Apple-PG zbyt niestabilne)	Konfiguracja treningu	I12 (zobacz rozdz. 9)
Przetwarzać klatki wideo bez ogniskowej EXIF	Konfiguracja treningu	I13 (FOV Override)
Wydajność COLMAP: GLOMAP zamiast klasycznego	Konfiguracja treningu	I14
Z Classic na MCMC przełączyć	Konfiguracja treningu	I15
Trening dłużej działać	Konfiguracja treningu	I18 (Max Iter) + I20 (Densify Until) — sprzężone przez I19
Czas GPU zmniejszyć o połowę	Konfiguracja treningu	I22 (Render Scale na 50%)
Jakość treningu +6% (MCMC)	Konfiguracja treningu	I16 (MCMC Quality)
Scena plenerowa z wieloma punktami SfM	Konfiguracja treningu	I17 (Auto-scale by scene)
Skonfigurować/zmienić ścieżkę COLMAP	Konfiguracja treningu	I23 / I24 / I25
Zmniejszyć pliki eksportu	Enhancements	I26 (zawsze zostawić włączone)
Widok ostrzejszy bez dodatkowego czasu treningu	Enhancements	I27 (Viewport Scaling → MetalFX)
MetalFX zbyt wygładza → alternatywa	Enhancements	I27 (Viewport Scaling → Lanczos)
Ostatnia kropla detalu przy drobnych strukturach	Enhancements	I29 (Perceptual Loss 0.05-0.1)
Redukcja czasu treningu przy większej rozdzielczości	Loss	I30 (Perceptual Loss 0.05-0.1) (SfM tylko)

ROZDZIAŁ

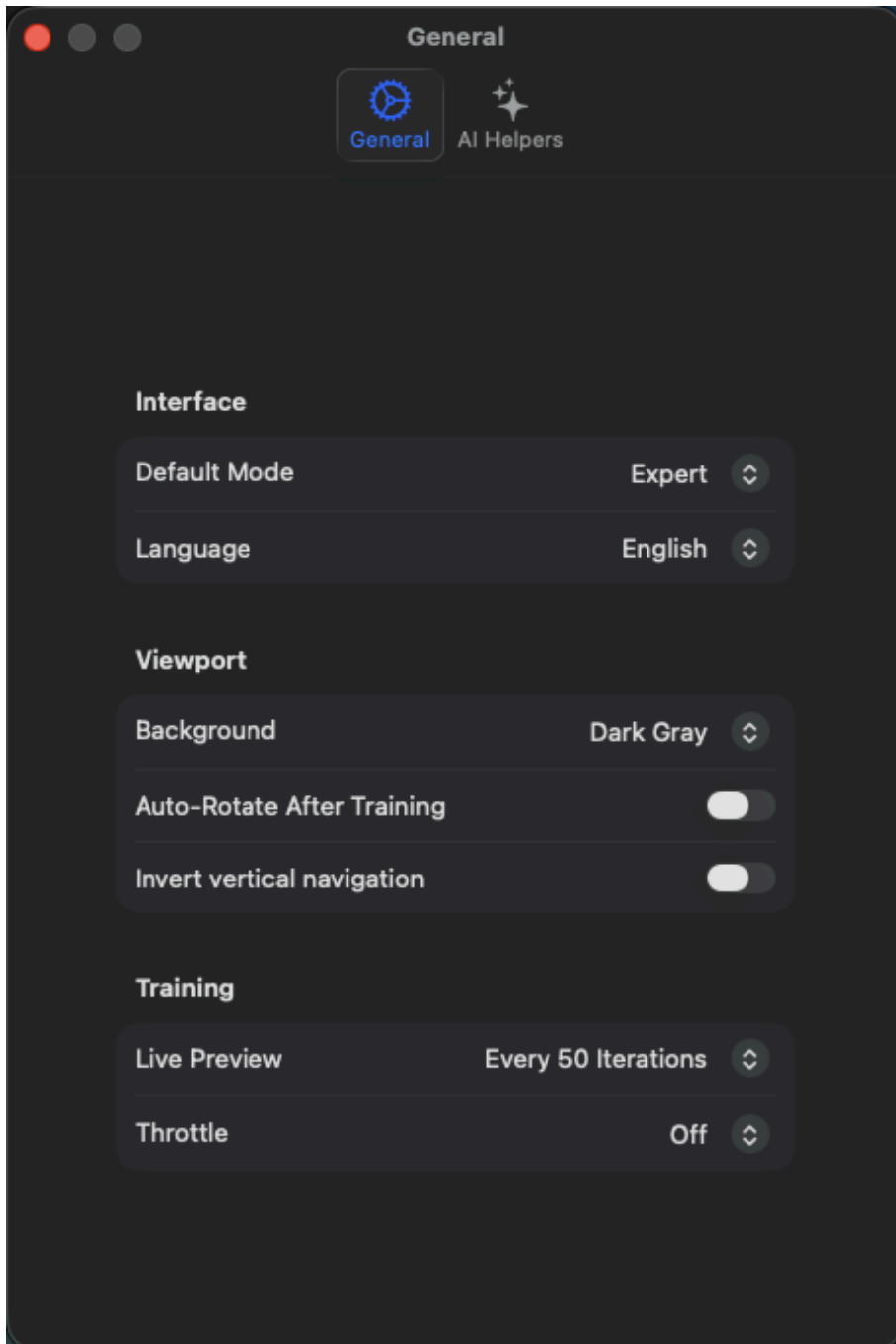
Rozdział 3 — Ustawienia

Okno ustawień otwiera się przez RadianceKit → Ustawienia... lub standardowy skrót klawiszowy **⌘, .** Zawiera dwie karty: **General** i **AI Helpers**. W przeciwieństwie do wartości Inspector z rozdziału 2 ustawienia z tego okna działają **globalnie dla aplikacji** (we wszystkich projektach) — są trwale zapisywane i przeżywają restarty aplikacji.

Karta General grupuje trzy sekcje tematyczne: Interface, Viewport i Training. Karta AI Helpers włącza pomocniki on-device do machine learningu (Vision, CoreML) dla wstępnego przetwarzania SfM i treningu. (Trzy przełączniki redukcji floaterów plenerowych — Sky Masking, Mid-Training Floater Cleanup, Reconstruct Sky Dome — które dawniej tu mieszkały, od wersji v1.6 przeniosły się do sekcji Enhancements w Inspectorze eksperckim, gdzie są teraz przechowywane per projekt; zobacz rozdział 2, 142–144.)

Wcześniejsze elementy sterujące do zbiorczego włączania lub wyłączenia wszystkich AI-Helpers w obecnej wersji już nie istnieją — odpowiednio nie są tu dokumentowane. Również wcześniejszy obszar „Coming Soon„ dla jeszcze niewysłanych pomocników został usunięty i nie jest tu referowany.

Karta General



Rysunek 15: Ustawienia → karta General z sekcjami Interface, Viewport, Training i Experimental

S1 Default Mode

GDZIE

Settings → General → Interface → selektor Default Mode. Bound: . Default: `.simple`.

TECHNICZNIE

Steruje, w którym z dwóch trybów UI aplikacja otwiera się po następnym uruchomieniu. „Simple Mode„ to prowadzony przepływ kreatora w 4 krokach (Import → Processing → Preview → Export, udokumentowany w rozdziale 10 pod Z1–Z4), „Expert Mode” to klasyczny układ trzech paneli z navigatorem, widokiem 3D i Inspektorem ekspertским z rozdziału 2. Wartość jest zapamiętywana między restartami. Identyczny efekt jak menu Mode → Simple Mode (⌘1) / Mode → Expert Mode (⌘2), z tym że menu przełącza bieżącą sesję, podczas gdy ten selektor ustala wartość domyślną dla przyszłych sesji. Oba tryby sięgają do tego samego stanu projektu — projekty, kamery i konfiguracja treningu pozostają zachowane przy zmianie trybu. Specyficzne dla trybu przyciski paska narzędzi są natychmiast ponownie renderowane.

PROSTYMI SŁOWAMI

Tutaj wybierasz, z jakim interfejsem RadiancKit startuje przy następnym uruchomieniu. „Simple Mode„ to tryb początkujący: cztery przejrzyste kroki, predefiniowane ustawienia, prawie żadnych opcji. „Expert Mode” to pełny układ skrzynki z narzędziami ze wszystkimi pokrętkami, jakie widzisz w rozdziale 2. Możesz w każdej chwili przełączać się przez menu „Mode„ tam i z powrotem, bez utraty obrazów ani postępu treningu.

S2 Language

GDZIE

Settings → General → Interface → selektor Language. Bound: . Default: `.system` (podąża za językiem macOS).

TECHNICZNIE

Wybiera język wyświetlania całego UI aplikacji, niezależnie od języka systemu macOS. RadiancKit jest zlokalizowany w 17 językach (`de` , `en` , `pl` , `en-AU` , `ar-SA` , plus 12 dalszych). Przy „System„ aplikacja podąża za językiem macOS. Przy jawnym wyborze ustawienie języka jest zapamiętywane między restartami; pełne działanie wymaga z reguły restartu aplikacji, ponieważ pakiety lokalizacji są ładowane tylko przy starcie. 298 udokumentowanych kluczy lokalizacji w projekcie są wszystkie uwzględniane, włącznie ze wszystkimi tekstami w sub-widokach i podpowiedziach pomocy.

PROSTYMI SŁOWAMI

Jeśli twój Mac działa po angielsku, ale wolisz niemiecki interfejs RadiancKit (lub odwrotnie), ustaw to tutaj. Większość tekstów zmienia się natychmiast. Niektóre okna dialogowe pojawiają się w nowym języku dopiero po restarcie aplikacji.

S3 Viewport Background

GDZIE

Settings → General → Viewport → selektor Background. Bound:.. Default: `.darkGray` (RGB 0.1, 0.1, 0.1).

TECHNICZNIE

Ustawia domyślny kolor tła dla widoku 3D. Trzy opcje: „Dark Gray„ (RGB 0.1, 0.1, 0.1 — domyślne), „Black” (0, 0, 0) i „White„ (1, 1, 1). Ustawienie utrwała domyślną wartość dla nowych projektów i sesji przez restarty i jednocześnie natychmiast aktualizuje działający renderer Metal. Identyczne opcje znajdują się w menu Viewport → Background (M21, M22, M23), ale selektor ustawień ustawia domyślne, podczas gdy menu przełącza bieżący wyświetlany obraz. Ważne dla zrzutów ekranu i wideo demonstracyjnych: białe tła silniej uwypuklają zielone/niebieskie floatery, ciemne tła są lepsze do czystych ujęć renderowanych.

PROSTYMI SŁOWAMI

Kolor za twoimi modelami 3D w oknie podglądu. Ciemnoszare jest standardem i pasuje do większości scen. Biały jest dobry do zrzutów ekranu, czarny działa bardziej elegancko przy ujęciach renderowanych. Możesz kolor w każdej chwili przełączyć przez menu „Viewport → Background„ dla bieżącej sceny — to ustawienie określa tylko, jaki kolor ma być aktywny przy następnym otwarciu.

S4 Auto-Rotate After Training

GDZIE

Settings → General → Viewport → przełącznik „Auto-Rotate After Training„. Bound:.. Default: `false`.

TECHNICZNIE

Startuje bezpośrednio po zakończeniu treningu ciągłą rotację typu turntable kamery widoku wokół środka ciężkości sceny (standardowa prędkość obrotu ~0.3 rad/s). Praktycznie użyteczne do sesji demonstracyjnych, porównań A/B oraz do bezpośredniej oceny w widoku 360°, czy na brzegu sceny powstały „floatery„. Efekt jest wizualnie identyczny z menu Viewport → Toggle Auto-Rotation (M16, $\text{⌘}T$), z tym że przełącznik tutaj wyzwala zachowanie automatycznie po zakończeniu treningu zamiast ręcznie. Można w dowolnej chwili przerwać przez menu lub kliknięcie w widok (co pauzuje rotację). Nie ma wpływu na wydajność treningu — rotacja startuje dopiero, gdy trening jest gotowy.

PROSTYMI SŁOWAMI

Gdy włączone, scena 3D obraca się automatycznie, gdy tylko trening się zakończy — jak karuzela. Miło, gdy przy nocnym treningu rano widzisz wynik już w ruchu, bez konieczności samodzielnego klikania. Przy długich sesjach, w których tylko nadzorujesz trening, pozostaw lepiej wyłączone.

S5 Live Preview Interval **GDZIE**

Settings → General → Training
→ selektor Live Preview. Bound:
AppState.trainingConfig.livePreviewInterval.
Default: 0 (Off).

 **TECHNICZNIE**

Określa, w jakim odstępie iteracji bieżący snapshot treningu jest renderowany do widoku 3D. Cztery dyskretne wartości: 0 („Off”), 50, 250, 1000 iteracji. Przy aktywnym Live Preview trener kopiuje bufor gaussianów z GPU do osobnego bufora renderowania i wyzwala odświeżenie widoku. Przy „Off” widok jest aktualizowany dopiero po zakończeniu treningu. Koszt wydajności: co 50 iteracji ~5–10% wolniej na M3 Ultra, co 250 iteracji ~1–2% wolniej, co 1000 iteracji niemierzalne. Narzut pamięci stały ~2 GB dla bufora snapshotu, niezależnie od interwału. Wartość służy jako domyślna dla nowych treningów; po starcie treningu Inspector treningu pokazuje rzeczywistą bieżącą wartość tego treningu. Przy interwale 50 wrażenie wizualne to płynne „dorastanie,” chmury punktów, przy 1000 wygląda to skokowo.

 **PROSTYMI SŁOWAMI**

Podczas treningu możesz wybrać, jak często widok 3D jest aktualizowany. „Off” oznacza: brak aktualizacji podczas treningu (najszybsze). „Every 50 Iterations” pokazuje niemal w czasie rzeczywistym, jak powstaje twoja scena (nieco wolniejsze). Do wygodnego obserwowania przy małych treningach „Every 250” to dobry kompromis.

S6 Throttle Delay



GDZIE

Settings → General → Training → selektor Throttle.
Bound: `AppState.trainingConfig.throttleDelayMs`.
Default: 0 (Off).



TECHNICZNIE

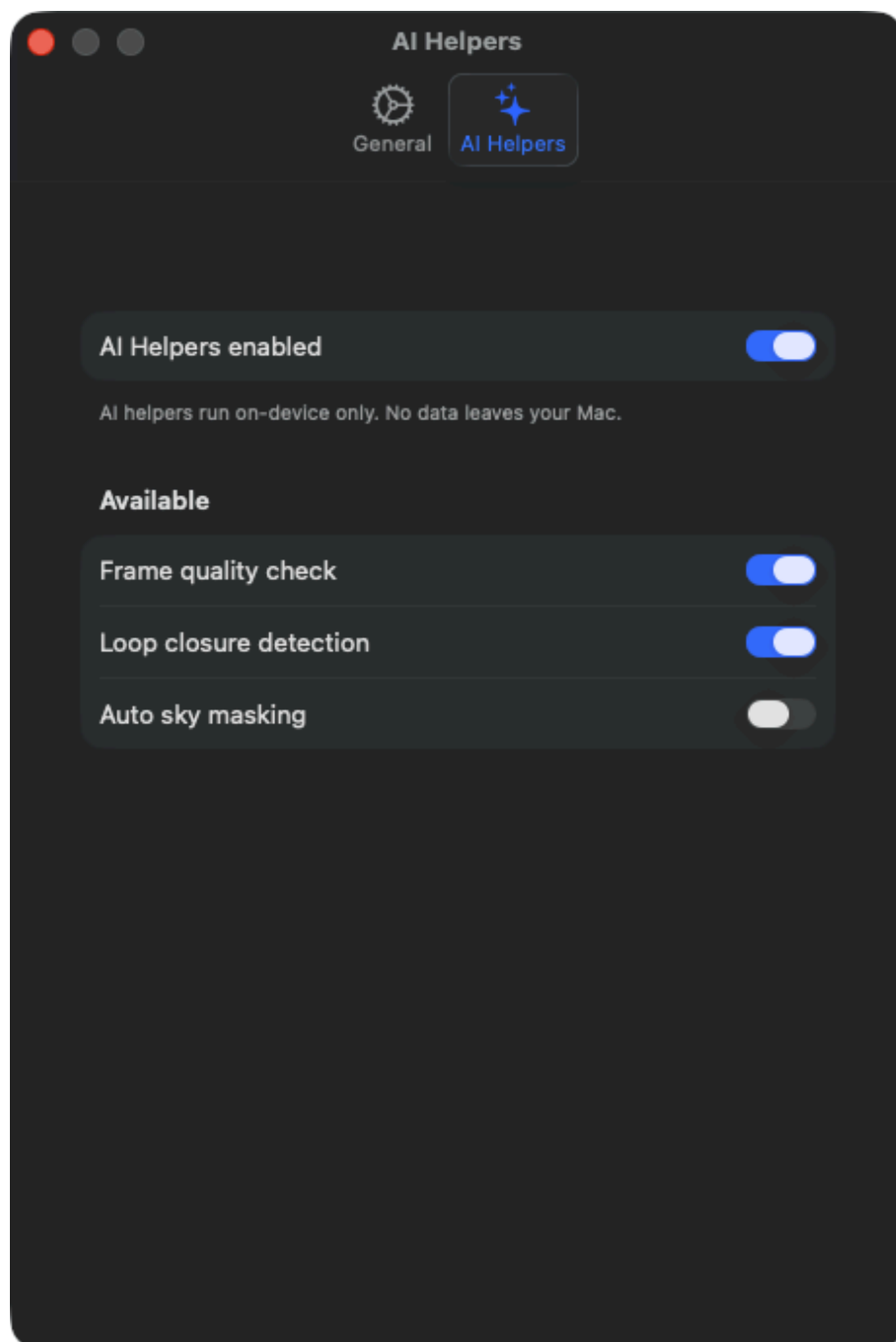
Wstawia między iteracjami treningu sztuczne opóźnienie w milisekundach. Cztery dyskretne wartości: 0 („Off”), 2 („Light”), 5 („Moderate”), 10 („Eco”).
Sens: przy dłuższych treningach (kilka godzin) GPU jest inaczej obciążone do 100%, co prowadzi do zauważalnie wolniejszego UI systemu (kursor myszy zacina się, inne aplikacje stają się ociężałe). Opóźnienie throttle daje GPU przerwy, w których mogą być wykonywane inne zadania. Koszt wydajności jest znaczny: przy 5 ms throttle typowy trening 40K trwa około 50–80% dłużej niż bez throttle. W trybie wydajności „Eco” (10 ms) opóźnienie na iterację jest dłuższe niż sama iteracja — współczynnik 2–3× wolniej. Przy aktywnym throttle pod selektorem pojawia się wskazówka: „Throttle is on. Training will be slower than usual.” Sama aplikacja nie reaguje zauważalnie lepiej — tylko inne aplikacje korzystają.



PROSTYMI SŁOWAMI

Jeśli twój Mac podczas długiego treningu robi się zbyt gorący lub inne programy stają się zbyt ociężałe, włącz tutaj hamulec. „Off” daje GPU pełny gaz (najszybsze). „Light” robi małą przerwę między każdym krokiem (nieco wolniejsze, ale system reaguje lepiej). „Eco” to najmocniejszy hamulec — dobre dla nocnych treningów na MacBooku, które nie powinny się zbyt grzać.

Karta AI Helpers



Rysunek 16: Ustawienia → karta KI-Helper z głównym przełącznikiem i podprzełącznikami

S11 AI Helpers enabled (Master)**GDZIE**

Settings → AI Helpers → pierwsza sekcja → przełącznik „AI Helpers enabled„. Bound: Default: `true` .

**TECHNICZNIE**

Główny przełącznik nad wszystkimi funkcjami AI-Helpers w potoku. Gdy wyłączony, potok importu i SfM całkowicie pomija wszystkie etapy wstępnego przetwarzania oparte na ML — brak wywołania Apple Vision, brak ładowania modelu CoreML, brak przebudzenia NPU. Gdy włączony, konsultowane są indywidualne podprzełączniki (S12–S13). Wartość jest zapamiętywana między restartami. Ma wpływ na następujące etapy: (a) Frame-Quality-Pre-Check przed SfM (S12), (b) wykrywanie zamknięcia pętli (S13). Ważne: gdy wyłączony, dwa podprzełączniki są dezaktywowane i wizualnie wyszarzone. Wskazówka w stopce podkreśla, że wszystkie AI-Helpers działają ściśle on-device — brak przesyłania obrazów, brak przetwarzania w chmurze. Gwarancja prywatności wynika z użycia wyłącznie frameworku Apple Vision (lokalnie na Neural Engine) oraz modeli CoreML, które leżą bezpośrednio w pakiecie aplikacji.

**PROSTYMI SŁOWAMI**

Główny przełącznik dla wszystkich funkcji, które wewnętrznie używają AI/machine learningu. Standard to „włączone„, ponieważ pomocniki oszczędzają wiele czasu bez konieczności, by twoje obrazy opuszczały Maca. Jeśli chcesz je całkowicie wyłączyć (np. aby oszczędzać prąd lub ponieważ twój Mac nie ma NPU), wyłącz tutaj — wtedy dwie opcje poniżej automatycznie szarzeją i nic już nie robią.

S12 Frame quality check**GDZIE**

Settings → AI Helpers → sekcja Available → przełącznik „Frame quality check„. Bound: Default: `true` .

**TECHNICZNIE**

Aktywuje screener jakości klatek (Faza 3.11), który przed wywołaniem SfM analizuje każdą zaimportowaną klatkę. Kroki potoku na klatkę: (a) filtr wariancji Laplasjana z Apple Vision (wykrywanie rozmycia — próg ~150), (b) sprawdzenie przeoraku / niedoekspozycji oparte na histogramie (próg: >5% pikseli przy 0 lub 255), (c) wykrywanie pustych klatek (odchylenie standardowe < 5 na wszystkich pikselach). Klatki przechodzące wszystkie trzy testy idą bezpośrednio. Klatki nie przechodzące co najmniej jednego testu wywołają modalny dialog potwierdzenia, który listuje każdą problematyczną klatkę z miniaturą i uzasadnieniem i pyta, czy usunąć. Ważne: brak automatycznego usuwania — dialog jest zawsze wymagany, użytkownik zachowuje ostateczną decyzję. Wydajność: ~50 ms na klatkę na M3 Ultra, działa równolegle. Gdy wyłączony, wszystkie klatki idą bez sprawdzenia do SfM. Przy wyłączonym Masterze (S11) ten przełącznik jest wizualnie wyszarzony i bez efektu. Status wysłany według pamięci: SHIP-PED 2026-05-23.

**PROSTYMI SŁOWAMI**

Przed właściwym treningiem aplikacja patrzy na każde zdjęcie: czy jest rozmazane? całkowicie ciemne lub białe? puste? Jeśli tak, pyta cię, czy chcesz obraz wyrzucić — nigdy nie usuwa nic automatycznie. To oszczędza później wiele godzin, ponieważ jedno totalnie rozmazane zdjęcie może czasem zepsuć cały trening. Standard to „włączone„, ponieważ koszt jest prawie zero, a korzyść duża.

S13 Loop closure detection**GDZIE**

Settings → AI Helpers → sekcja Available → przełącznik „Loop closure detection„. Bound: Default: `true`.

**TECHNICZNIE**

Aktywuje wykrywanie zamknięcia pętli oparte na Apple Vision Feature-Print. Dla każdej zaimportowanej klatki obliczany jest ~768-wymiarowy wektor cech, reprezentujący osadzenie neuronowe zawartości obrazu. Następnie wszystkie feature-prints są parami porównywane przez podobieństwo cosinusowe. Pary z podobieństwem > 0.85 i dystansem w indeksie klatek > 50 (czyli klatki niesąsiadujące) są identyfikowane jako „kandydaci na zamknięcie pętli„ i zapisywane do sidecar pliku JSONL w folderze projektu. Tylko informacyjne — zaimportowana sekwencja obrazów nie jest modyfikowana. Sens: daje solverowi SfM (zwłaszcza COLMAP) wskazówkę, że te klatki w przestrzeni 3D należą do siebie. Dla natywnego SfM informacja sidecar jest obecnie tylko dokumentująca; COLMAP używa wskazówek wewnętrznie przez plik niestandardowych dopasowań (integracja ręczna możliwa, nie podpięta automatycznie). Wydajność: ~200 ms na klatkę na M3 Ultra, działa równolegle. Gdy wyłączony, nie są generowane feature-printy. Przy wyłączonym Masterze (S11) wizualnie wyszarzony.

**PROSTYMI SŁOWAMI**

Gdy podczas fotografowania obchodzisz obiekt dookoła i na końcu ponownie lądujesz w punkcie startowym, ogromnie pomaga komputerowi o tym wiedzieć. Ta opcja automatycznie wykrywa, które z twoich zdjęć były robione „prawie z tego samego miejsca„, i zapisuje to do małego pliku pomocniczego. Narzędzia SfM (przede wszystkim COLMAP) mogą wykorzystać tę informację do dostarczenia czystszej rekonstrukcji 3D. Standard to „włączone“, ponieważ działa bez twojego udziału i nic nie zmienia w twoich obrazach.

Ustawienia lustra Inspectora

Pozostałe wpisy ustawień (S17–S33) z tabeli inwentarza są lustrami z Inspectora ekspertskiego i są udokumentowane w rozdziale 2 (kontrolki Inspectora I12–I29). Nie pojawiają się fizycznie w oknie Settings, lecz były wymienione w inwentarzu tylko dlatego, że działają przez właściwości `TrainingConfig`, które są persistowane przez i o tyle formalnie mają charakter ustawień. Po wyjaśnienia treściowe patrz tam.

Kiedy co?

Ustawienie	Zakres działania	Trwałość
S1 Default Mode	Aplikacja globalnie	Restart aplikacji
S2 Language	Aplikacja globalnie	Restart aplikacji
S3 Viewport Background	Aplikacja globalnie (Default) + Runtime	Restart aplikacji
S4 Auto-Rotate After Training	Aplikacja globalnie	Restart aplikacji
S5 Live Preview Interval	Domyślne dla nowych treningów	Restart aplikacji
S6 Throttle Delay	Domyślne dla nowych treningów	Restart aplikacji
S11 AI Helpers Master	Aplikacja globalnie	Restart aplikacji
S12 Frame quality check	Aplikacja globalnie	Restart aplikacji
S13 Loop closure detection	Aplikacja globalnie	Restart aplikacji

Aplikacja globalnie = działa na wszystkie projekty. Domyślne dla nowych treningów = działa tylko na następny utworzony trening, bieżące sesje pozostają niezmienione. Bieżący trening = działa natychmiast na bieżącą konfigurację treningu, ale nie persistuje bez jawnego reimportu.

ROZDZIAŁ

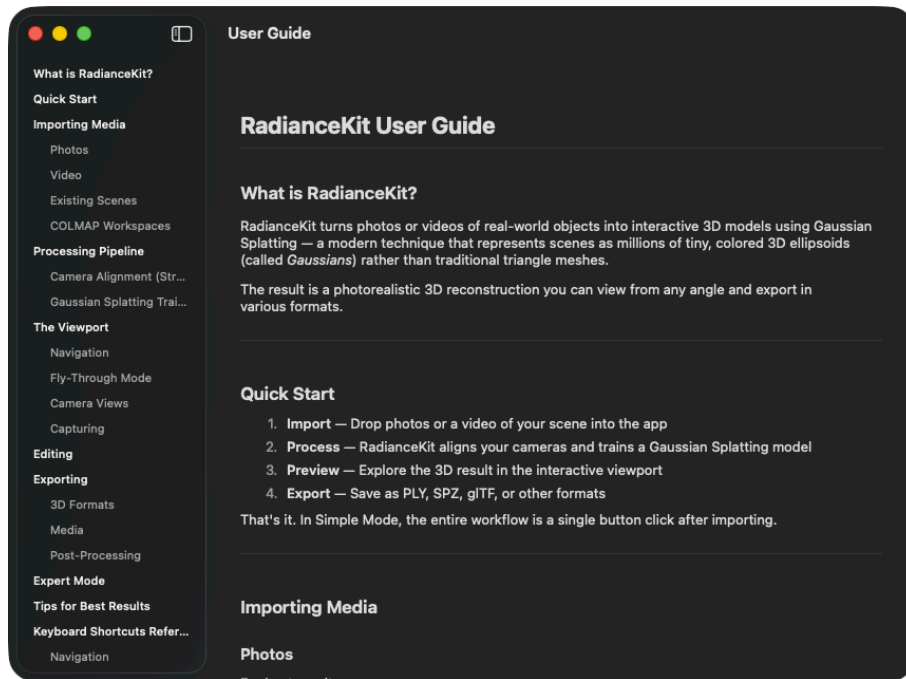
Rozdział 4 — Okna pomocnicze

Obok okna głównego (widok 3D plus Inspector) RadianceKit zarządza siedmioma kolejnymi oknami, wszystkie otwierane przez menu Help. Lista od góry do dołu: User Guide (⌘?), Keyboard Shortcuts (⌘/), Open Training Logs... (nie otwiera okna aplikacji, lecz Finder; dlatego nie omawiane tu dalej), Manage Storage..., Pareto Dashboard... (⇧⌘D), Holdout Analysis... (⇧⌘H), BayesOpt Console... (⇧⌘B). Trzy z nich — Dashboard, Holdout, BayesOpt — to samodzielne narzędzia analityczne. Każde z nich ma własny stos view-model, czyta lub zapisuje pliki JSON na dysku, a dla każdego istnieje argument CLI, którym możesz przy starcie aplikacji wskazać okno na konkretny plik (`--dashboard-dir` , `--holdout-file` , `--bayesopt-autorun`).

Cztery proste okna (User Guide, Keyboard Shortcuts, Manage Storage, plus podpozycje Open Training Logs / Open Exports Folder) otrzymują krótki wpis na element sterujący. Trzy okna analityczne są udokumentowane bardziej szczegółowo — każde z wprowadzeniem wyjaśniającym, co widzisz w oknie, kiedy je otworzyć i jak interpretować wyświetlany obraz.

Na końcu rozdziału znajduje się sekcja odsyłaczy do Inspektora okna głównego: co możesz sensownie odczytać w wykresie Loss na żywo i we wskaźniku liczby gaussianów podczas działającego treningu.

User Guide (W1–W4)



Rysunek 17: Okno User Guide z paskiem bocznym po lewej i wyrenderowaną treścią Markdown po prawej

Co to jest: Wbudowane okno pomocy, które renderuje dostarczony z aplikacją plik `guide_<język>.md`. Język jest wyprawdany z ustawień (karta General → Language) lub, gdy stoi tam „System”, z preferencji języka macOS. Układ jest klasyczny: po lewej pasek boczny ze wszystkimi nagłówkami, po prawej tekst ciągły.

📍 Gdy potrzebujesz szybkiego przypomnienia o pojedynczym punkcie — czyli jako zamiennika hasła. Szczegółowa referencja to ten podręcznik; wbudowane okno pomocy jest raczej tym, czym byłoby `--help` w wierszu poleceń. Jest aktualizowane z każdym wydaniem aplikacji, ale treściowo trzymane na poziomie bardziej powierzchownym.

W1 NavigationSplitView (pasek boczny + szczegóły)



GDZIE

Help → User Guide (⌘?).



TECHNICZNIE

Układ dwukolumnowy z wąskim paskiem bocznym (co najmniej 180 pt szerokości) dla drzewa treści i przewijalnym obszarem szczegółów dla właściwej treści Markdown. Okno ma minimalny rozmiar 700 x 500 pt. Przy pierwszym otwarciu okno ładuje pasujący `guide_<lang>.md` z pakietu aplikacji (fallback `guide_en.md`), parsuje go na rekordy bloków (nagłówki H1–H4, akapity, listy, tabele, linie podziału) i ekstrahuje osobno strukturę nagłówków dla paska bocznego. Formatowanie inline (Bold, Italic, Code-Span) jest renderowane przez wbudowany silnik Markdown. Język czytany jest z ustawień aplikacji, ze specjalnym przypadkiem chińskiego (`zh-Hans`) i brazylijskiego portugalskiego (`pt-BR`), które są utrzymywane jako pełne tagi locale, ponieważ te warianty różnią się od `zh` lub `pt`.

PROSTYMI SŁOWAMI

Wbudowany tekst pomocy, po lewej lista tematów, po prawej treść. Język ustawia się automatycznie według twoich ustawień systemu. Działa offline, ale celowo jest tylko wersją krótką — pełna referencja to ten podręcznik.

W2 List (pasek boczny nagłówków)



GDZIE

Lewa kolumna w oknie User Guide.



TECHNICZNIE

Lista wszystkich nagłówków H2 i H3 bieżącego dokumentu Markdown. Wpisy H2 pojawiają się bez wcięcia ze średnią grubością czcionki, wpisy H3 z wcięciem 16 pt po lewej i zredukowanym stylem pierwszego planu. H4 i głębiej są ignorowane, ponieważ głębia inaczej czyni pasek boczny nieprzejrzystym. Identyfikatory kotwic są generowane z tekstu nagłówka przez slugifikację (lowercase + spacje na myślniki + filtrowanie na litery/cyfry/myślniki — ten sam algorytm, którego GitHub używa do swoich kotwic Markdown, dzięki czemu również zewnętrzne URL-e do dokumentacji potencjalnie łądowałyby na tej samej kotwicy). Lista używa natywnego stylu macOS.

PROSTYMI SŁOWAMI

Pasek nawigacji po lewej stronie. Stuknij wpis i przeskakujesz do sekcji.

W3 Button (nagłówek → skok do kotwicy)

GDZIE

Na każdy wiersz paska bocznego jeden przycisk.

TECHNICZNIE

Każdy wpis paska bocznego to przycisk ustawiający bieżącą kotwicę, ale wyglądający optycznie jak wpis listy. Zmienna obserwatora wyzwala skok przewijania do odpowiedniej kotwicy z łagodną animacją przez 0,3 s. Po skoku wartość kotwicy jest resetowana, aby następne kliknięcie tej samej kotwicy znów zadziało (inaczej obserwator nie wyzwoliłby ponownie, bo wartość się nie zmieniła).

PROSTYMI SŁOWAMI

Kliknięcie prowadzi cię do odpowiedniego miejsca w tekście po prawej.

W4 ScrollView (treść szczegółowa)

GDZIE

Prawa kolumna.

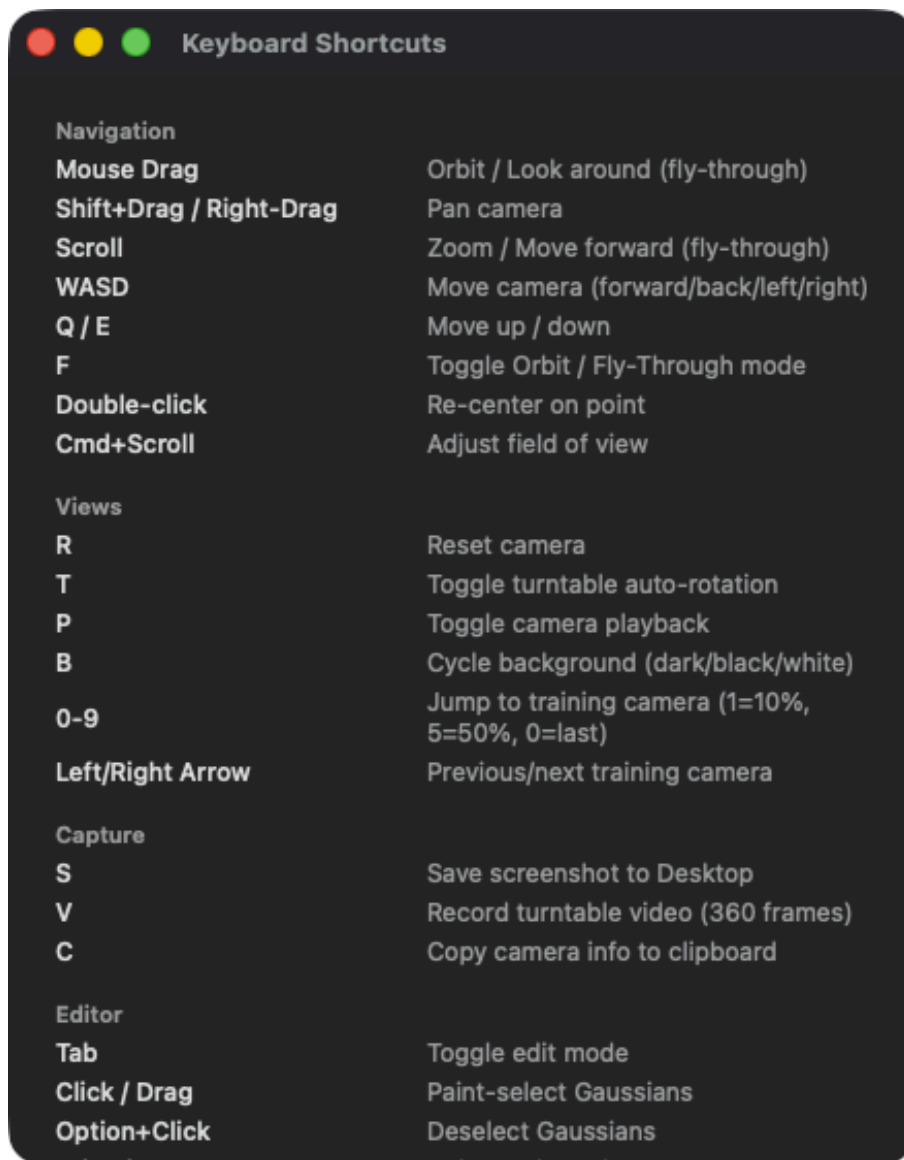
TECHNICZNIE

Przewijalny, pionowo stackujący obszar treści z lazy renderingiem, ponieważ dłuższe przewodniki łatwo mogą mieć ponad 200 bloków Markdown — wariant non-lazy instancjowałby je wszystkie jednocześnie. Każdy blok otrzymuje własne ID, albo kotwicę nagłówka (dla skalalnych H1–H3), albo placeholder indeksu. Maksymalna szerokość 720 pt, padding 32 poziomo / 24 pionowo, dzięki czemu długie linie zachowują dobrze czytelny układ. Tabele są renderowane komórka po komórce z poziomymi stosami i liniami podziału; kod inline przez wbudowany silnik Markdown. Prawdziwe bloki kodu są obecnie traktowane jako akapit — znane ograniczenie okna pomocy.

PROSTYMI SŁOWAMI

Właściwy tekst pomocy. Przewijalny, dobrze czytelnej szerokości, czysta typografia.


Keyboard Shortcuts (W5–W6)



Rysunek 18: Okno Keyboard Shortcuts — pięć grup Navigation/Views/Capture/Editor/Training z kolumną skrótów po lewej i opisem po prawej

CO WIDAĆ NA OBRAZIE Statyczna lista referencyjna w pięciu sekcjach. **Navigation:** Mouse Drag (Orbit/Fly), Shift+Drag/Right-Drag (Pan), Scroll (Zoom), WASD (ruch fly-through), Q/E (Up/Down), F (przełącz Orbit/Fly), podwójne kliknięcie (wycentrowanie), Cmd+Scroll (dostosowanie FoV). **Views:** R (reset kamery), T (auto-rotacja), P (Camera Playback), B (cykl tła), 0–9 (skok do kamery treningowej 1=10%/5=50%/0=last), strzałki lewo/prawo (poprzednia/następna kamera). **Capture:** S (zrzut ekranu na pulpit), V (Turntable-Video), C (kopiowanie informacji o kamerze). **Editor:** Tab (tryb edycji), Click/ Drag (Paint-Select), Option+Click (deselect), X / Delete (usuń selekcję), Cmd-Z (cofnij ostatnie usunięcie), [/] (rozmiar pędzla mniejszy/większy), Esc (zwolnij selekcję). **Training:** Start, Pause/Resume, Cancel, Continue +5K/+10K/+20K przez skróty menu w M9–M14.

Co to jest: Prosty statyczny przegląd wszystkich skrótów klawiszowych — Navigation, Views, Capture, Editor, Training. Treść jest hardkodowana, brak ładowania Markdown.

 Gdy szukasz najszybszej drogi do zrobienia czegoś w widoku. WASD-Fly-Through, R dla reset kamery, B dla cyklu tła — wszystkie stoją tu.

W5 ScrollView (obszar treści)

GDZIE

Help → Keyboard Shortcuts (⌘/).

TECHNICZNIE

Prosty obszar przewijania z pionową listą w środku. Padding 20 wokół, brak drzewa nawigacji bocznej (lista jest dostatecznie krótka). Treści zgrupowane w pięciu sekcjach (Navigation, Views, Capture, Editor, Training). Na kombinację klawiszy jeden wiersz z tłumaczalnym tekstem w obu kolumnach. Lewa kolumna (kod klawisza) zafiksowana na 180 pt szerokości, dzięki czemu opisy po prawej pozostają wyrównane pionowo. Brak interakcji poza przewijaniem — kliknięcie wiersza nic nie wyzwała, skróty klawiszowe są rzeczywistymi modyfikatorami klawiatury w menu i w widoku.

PROSTYMI SŁOWAMI

Tabela wszystkich klawiszy skrótów. Statyczna ściągawka do szybkiego zerknięcia.

W6 VStack (sekcje skrótów)

GDZIE

Wewnątrz ScrollView.

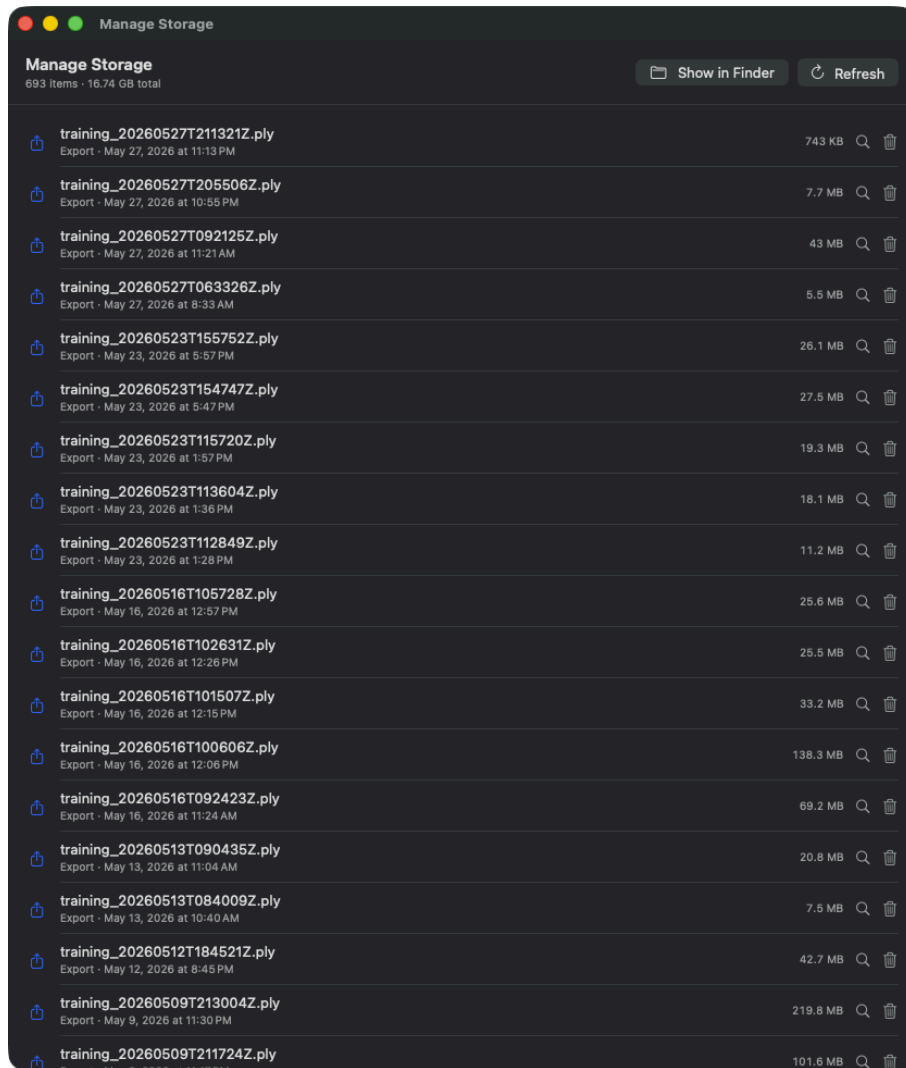
TECHNICZNIE

Sekcje stackowane do lewej z odstępem 16 pt. Wewnątrz pięciu sekcji nagłówek + sekwencja wierszy. Nagłówki używają drugorzędowego stylu Subheadline — celowo nie format Title, ponieważ sekcje nie muszą być nawigowalne. Treść jest celowo płaska (brak disclosure, brak wyszukiwania, brak filtru), aby komponent działał niezmiennie w każdej wersji macOS i plik pozostał czytelny.

PROSTYMI SŁOWAMI

Grupowanie klawiszy według funkcji (Navigation, Views, Editor i tak dalej).

Manage Storage (W7–W12)




Rysunek 19: Okno Manage Storage — nagłówek pokazuje „693 items · 16.74 GB total”, tabela z plikami eksportu PLY posortowana według daty, każda pozycja z pigułką formatu + nazwa pliku + rozmiar + data

CO WIDAĆ NA OBRAZIE Widok tabelaryczny wszystkich zarządzanych przez RadianceKit plików. Nagłówek liczy 693 pozycje, łączny rozmiar 16.74 GB. Pasek narzędzi u góry: „Show in Finder,”

1. „Refresh,”. Każdy wiersz: ikona PLY, nazwa pliku (np.

training_20260527T211321Z.ply), data eksportu, rozmiar (waha się 7 KB do 218 MB), ikona lupy (Reveal) i ikona kosza (Move to Trash). Pliki są posortowane według daty, najnowsze u góry. W tym ujęciu demonstracyjnym dominują eksporty PLY, ponieważ dużo pracowano z `--benchmark`.

Co to jest: Przegląd użycia dysku dla wszystkiego, co RadianceKit składa pod `~/Documents/RadianceKit/` — logi, eksporty, sceny, pakiety capture (od towarzysza iOS), importy (staging-kopie obrazów wejściowych). Na pozycję rozmiar w bajtach i dwa przyciski: „pokaż w Finderze,” i „przenieś do kosza”. NIE jest to automatyczne sprzątnięcie — aplikacja sama niczego nie usuwa; ty decydujesz na pozycję.

 Gdy dysk się zapełnia. Przede wszystkim logi się zbierają (jeden JSONL na próbę treningu, plus `_qualityMetrics.json`); eksporty oczywiście też (PLY 100% surowych danych, jeden na eksport). Również użyteczne po crashu, gdy katalog staging importów ma jeszcze stare kopie obrazów wejściowych (zobacz „Disk-pressure incident„ w `dev_v549f-needle-reduction.md`).

W7 Przycisk „Show in Finder„

GDZIE

Nagłówek u góry po prawej w oknie przeglądarki storage.

TECHNICZNIE

Otwiera cały katalog RadianceKit (`~/Documents/RadianceKit/`) w Finderze, dzięki czemu możesz bezpośrednio zobaczyć strukturę folderu i też manipulować nią z samego Findera. Akcja otwiera nowe okno Findera i nie przechodzi do kontenera sandbox aplikacji — `~/Documents/RadianceKit/` to regularnie dostępna dla aplikacji domena Documents, nie ścieżka kontenera sandbox.

PROSTYMI SŁOWAMI

Otwiera katalog w Finderze, abyś mógł sam operować plikami.

W8 Przycisk „Refresh„

GDZIE

Nagłówek, obok przycisku Finder.

TECHNICZNIE

Wyzwała skanowanie w tle, które działa na zainicjowanym przez użytkownika asynchronicznym zadaniu, dzięki czemu skanowanie dużych drzew katalogów nie blokuje UI. Właściwe chodzenie przebiega przez każdy znany podkatalog (Logs, Exports, Scenes, Captures, Imports) i generuje pozycję storage na bezpośrednie dziecko. Na pozycję ustalany jest rekursywny rozmiar — preferowane jest faktyczne zużycie dysku (włącznie ze współdzieleniem hardlinków APFS) z fallbackiem na logiczny rozmiar pliku.

PROSTYMI SŁOWAMI

Wczytuje listę na nowo, gdy w międzyczasie w Finderze coś usunąłeś lub dodałeś.

W9 List (wpisy storage) **GDZIE**

Główna treść pod nagłówkiem.

 **TECHNICZNIE**

Lista z układem na wiersz: kategoria-specyficzna ikona SF Symbol (dokument dla logów, strzałka uploadu dla eksportów, sześćcian dla scen, taca dla importów), nazwa + podtytuł (etykieta kind + sformatowana data modyfikacji), licznik bajtów po prawej (wyrównany do prawej, monospaced), przycisk Reveal (symbol lupy), przycisk Trash (kosz). Sortowanie: pierwotnie według kind (sceny najpierw, potem eksporty, logi, captures, importy, inne), drugorzędnie według daty modyfikacji malejąco (najnowsze u góry). Gdy skanowanie wciąż trwa, w miejscu wyświetlany jest postęp „Scanning...”. Gdy nic nie znaleziono, wyświetlanie empty-state z ikoną tacy.

 **PROSTYMI SŁOWAMI**

Lista wszystkich twoich danych RadianceKit, posortowana według typu i aktualności. Na pozycję widzisz rozmiar i możesz bezpośrednio usunąć.

W10 Przycisk wiersza „Reveal in Finder,” **GDZIE**

Na wiersz, symbol lupy po prawej.

 **TECHNICZNIE**

Otwiera Finder i zaznacza specyficzny element (plik lub folder). Różnica wobec W7: W7 otwiera katalog root; W10 zaznacza dokładnie tę jedną pozycję. Praktyczny workflow: zidentyfikuj duży wpis, kliknij na lupę, potem skopiuj go na przykład na zewnętrzny wolumin.

 **PROSTYMI SŁOWAMI**

Skacze w Finderze bezpośrednio na ten wpis, abyś szybko go znalazł.

W11 Przycisk wiersza „Move to Trash„ GDZIE

Na wiersz, symbol kosza po prawej obok lupy.

 TECHNICZNIE

Wyzwała okno potwierdzenia dialogu (W12). Dopiero po potwierdzeniu działa standardowa operacja macOS „przenieś do kosza„ (czyli odwracalna, nie bezpośrednio usunięcie). Po udanym trash wpis jest usuwany z listy i licznik łącznych bajtów aktualizowany. Przy błędach wyświetlany jest modalny dialog błędu.

 PROSTYMI SŁOWAMI

Przenosi pozycję do kosza. Dialog wcześniej pyta.

W12 ConfirmationDialog (potwierdzenie usunięcia) GDZIE

Wyzwalane przez W11, prezentowane jako sheet macOS.

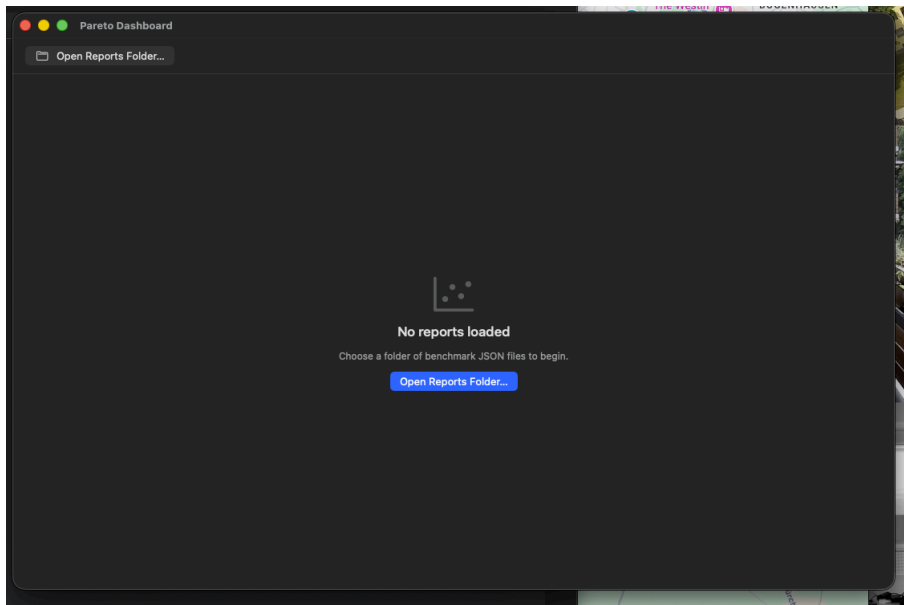
 TECHNICZNIE

Standardowy dialog potwierdzenia z dynamicznym tytułem „Delete <name>?„ i wierszem wiadomości jawnie wskazującym, że pozycja ląduje w koszu i z niego jest przywracalna (do czasu opróżnienia kosza). Dwa przyciski: „Move to Trash” jako akcja destrukcyjna (prezentowana czerwono) i „Cancel„ z automatycznym przypisaniem Esc. Dialog jest non-modal w tym sensie, że blokuje tylko to okno, nie całą aplikację — to standard macOS dla odwracalnych usunięć.

 PROSTYMI SŁOWAMI

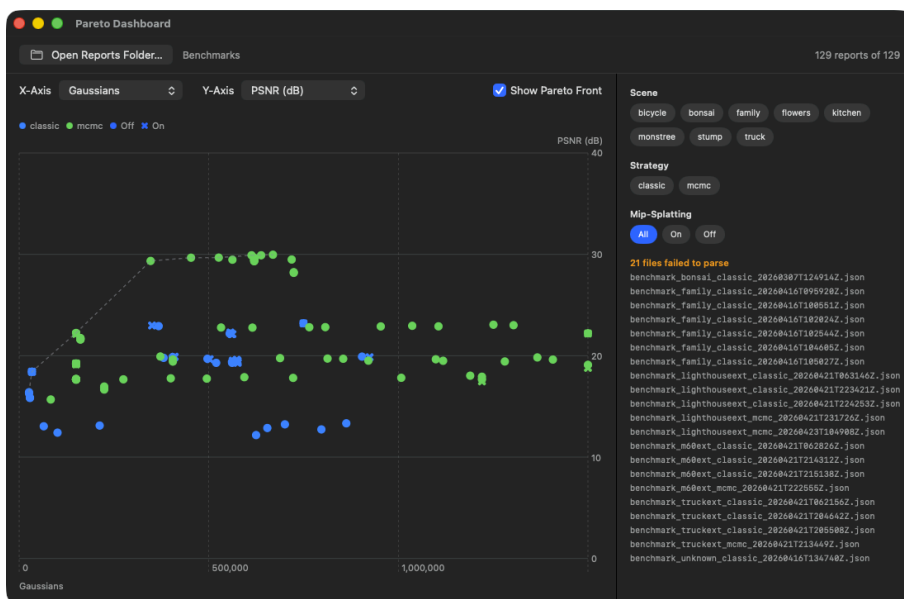
Pytanie bezpieczeństwa przed usunięciem. „Move to Trash„ jest odwracalne — dopóki kosz nie zostanie opróżniony.

Pareto Dashboard (W13–W22)



Rysunek 20: Pareto Dashboard — pusty stan przed importem raportu

Pusty stan (po pierwszym otwarciu) — empty-state z call-to-action „Open Reports Folder...”. Punkty danych pojawiają się, gdy tylko raporty treningu są wczytane, zobacz następane zdjęcie.




Rysunek 21: Pareto Dashboard z 129 wczytanymi raportami benchmark — Gaussians vs PSNR z frontem Pareto, filtry Scene/Strategy/Mip

CO WIDAĆ NA OBRAZIE Pasek narzędzi nagłówka pokazuje „129 reports of 129” (wszystkie raporty w wybranym folderze pomyślnie sparsowane — 21 dodatkowych plików nie udało się sparsować z powodu starszego formatu, zobacz listę wskazówek po prawej). Osie: selektor X-Axis na `Gaussians`, selektor Y-Axis na `PSNR (dB)`. Wykres scatter: zielone punkty = strategia Classic, niebieskie punkty = MCMC. Przerwana linia frontu


Pareto biegnie wzdłuż najlepiej osiągniętych wartości PSNR i osiada na plateau ok. PSNR≈30 dB od ok. 500K gaussianów. Pigułki filtrów po prawej: 7 scen (bicycle, bonsai, family, flowers, kitchen, stump, truck), 2 strategie (classic, mcmc), 3 opcje Mip-Splatting (All, On, Off). Obecnie wszystkie filtry otwarte, stąd gęsty klaster punktów.

Co to jest: Narzędzie porównawcze wielu przebiegów. Trenowałeś w przeszłości kilka scen lub tę samą scenę z różnymi ustawieniami wstępnymi — każdy taki przebieg treningu produkuje (gdy podałeś `--benchmark` lub uruchomiłeś przez funkcję `benchmark`) plik JSON-raportu, zawierający m.in. final-PSNR, SSIM, LPIPS, liczbę gaussianów i czas wallclock. Dashboard wczytuje cały folder takich raportów jednocześnie i plotuje je jako 2D scatter z wybieralnymi osiami. Dodatkowo front Pareto (zbiór niezdominowanych punktów) jest rysowany jako przerywana linia.

 Po założeniu co najmniej trzech lub czterech raportów treningu. Z mniejszą liczbą punktów linia frontu nie jest wymowna. Typowy use-case: próbowałeś zrekonstruować scenę plenerową i kolejno wypróbowałeś P3 Balanced (Classic), P4 Quality (Classic), P7 MCMC Quality i P9 Outdoor (tuned) — teraz chcesz wiedzieć, która konfiguracja daje najlepszą PSNR na sekundę czasu treningu lub która potrzebuje najmniej gaussianów dla danej PSNR.

JAK INTERPRETOWAĆ Obie osie są swobodnie wybieralne (X: `psnr`, `ssim`, `lpips`, ...; Y tak samo). Logika frontu Pareto w `ParetoFront2D.indices` wie dla każdej metryki, czy „mniejsze = lepsze„ (np. LPIPS, Loss, Time) czy „większe = lepsze” (PSNR, SSIM) — linia biegnie więc w zależności od wyboru osi z lewego dolnego do prawego górnego rogu lub z lewego górnego do prawego dolnego, zawsze wzdłuż najlepszej osiągniętej kombinacji. Punkt jest Pareto-optymalny, gdy ŻADEN inny punkt w OBU wymiarach nie jest przynajmniej równie dobry (czyli żaden inny go nie dominuje). Pareto-optymalne punkty leżą na linii, inne punkty na prawo/powyżej (zależnie od orientacji osi) od niej. Punkty NA linii to prawdziwi kandydaci na „najlepsze ustawienie wstępne„; punkty DALEKO od linii to zmarnowany czas treningu.

PIGUŁKI FILTRÓW Możesz ograniczyć wybór do konkretnej sceny (gdy np. chcesz porównywać tylko przebiegi plenerowe), do konkretnej strategii (Classic lub MCMC) lub do Mip-Splatting on/off (istotne po Fazie Q1.5, gdzie Mip pozostaje jako opt-in Advanced Flag).

 Masz trzy raporty dla sceny „truck„ pod `~/Documents/RadianceKit/Reports/`: Run A (P4 Quality, 40K iter, 524K Gs, 105 s, PSNR 23.4), Run B (P7 MCMC, 200K iter, 150K Gs, 693 s, PSNR 24.6), Run C (P9 Outdoor, 100K iter, 1.25M Gs, 312 s, PSNR 25.8). Ustaw oś X na `trainingTime`, oś Y na PSNR. Run B leży po prawej u góry, Run C jeszcze bardziej w prawo u góry, Run A po lewej u dołu. Front Pareto łączy A i C — oba niezdominowane. Run B jest „lost” (C jest lepsze w Time i PSNR). Wniosek: dla „truck„ nie opłaca się domyślne MCMC; albo szybko+ok (A), albo długo+bardzo dobrze (C). Zapisz konfigurację z C jako własne ustawienie wstępne (Inspector → I1 Save Preset).

Następna akcja: Najlepszą konfigurację zapisać jako ustawienie wstępne. Konkretnie: spójrz na punkty Pareto (hover pokazuje PSNR/SSIM/LPIPS/Gs/Time w tooltipie), zdecyduj, który najlepiej pasuje ci pod kątem trade-off Time-vs-Quality, otwórz odpowiedni raport (nazwa pliku zawiera timestamp przebiegu), skopiuj jego konfigurację treningu

w nowym przebiegu lub zapisz ją po następnej sesji treningu jako ustawienie wstępne przez Inspector.

W13 Przycisk „Open Reports Folder...”

GDZIE

Pasek narzędzi u góry po lewej.

TECHNICZNI

Otwiera dialog wyboru folderu z prośbą „Select a folder containing benchmark .json reports”. Po potwierdzeniu działa zadanie w tle parsujące sekwencyjnie wszystkie pliki `.json` w folderze. Błędne raporty (zepsute JSON, błędny schemat) są zbierane i wyświetlane na dole paska bocznego jako „N file failed to parse” — bez crasha. Gdy drugie kliknięcie nastąpi podczas trwającego pierwszego ładowania, poprzednie zadanie jest anulowane, aby dwa wyniki jednocześnie nie pisały do stanu.

Także przez CLI: `--dashboard-dir /ściezka/do/raportow` ładuje folder już przy starcie aplikacji.

PROSTYMI SŁOWAMI

Wybiera folder, w którym leżą twoje raporty benchmark. Standardowa ścieżka to `~/Documents/RadianceKit/Reports/`. Ładuje wtedy wszystkie JSONy naraz.

W14 Selektor „X-Axis”

GDZIE

Nad wykresem, po lewej.

TECHNICZNI

Selektor menu ze wszystkimi dostępnymi osiami metryk modułu dashboard (PSNR, SSIM, LPIPS, liczba gaussianów, czas treningu itd.). Domyślnie liczba gaussianów. Przy zmianie hover-punkt jest resetowany, ponieważ wcześniej podświetlona pozycja w starym układzie współrzędnych osi po zmianie osi nie ma więcej sensu. Selektor jest ograniczony do szerokości zawartości, aby nie ciągnął się przez całą szerokość.

PROSTYMI SŁOWAMI

Która metryka ma stać na osi poziomej. Zwykle „czas treningu”, lub „liczba gaussianów”, ponieważ to „koszty”, które chcesz porównywać.

W15 Selektor „Y-Axis„ GDZIE

Nad wykresem, obok X-Axis.

 TECHNICZNIE

Identyczne do W14, tylko że domyślnie PSNR. Wybór osi jest zapisywany niezależnie, więc użytkownik może też wybierać bezsensowne kombinacje ($X=PSNR$, $Y=PSNR$ — wyrzuciłoby wszystkie punkty na przekątną). Takie kombinacje nie są jednak wyłapywane; celowa decyzja, ponieważ porównanie „SSIM vs PSNR„ jest jak najbardziej interesujące, by zobaczyć jak konsystentne są metryki.

 PROSTYMI SŁOWAMI

Co stoi na osi pionowej. Normalnie „PSNR„ lub „SSIM” jako miara jakości.

W16 Przełącznik „Show Pareto Front„ GDZIE

Po prawej obok selektorów osi.

 TECHNICZNIE

Standardowy przełącznik macOS. Gdy aktywny, na wykresie Pareto dodatkowo do chmury punktów rysowana jest linia z obliczonym frontem Pareto 2D. Styl: przerywana (wzór 4–4), półprzezroczysta szarość, grubość linii 1,5 pt. Obliczenia Pareto działają na głównym wątku — przy typowej liczbie raportów ($\leq \sim 50$) jest to bez problemu szybkie. Gdy przełącznik jest wyłączony, linia jest pominięta, dzięki czemu pozostają tylko gołe punkty.

 PROSTYMI SŁOWAMI

Pokazuje linię prowadzącą przez „dotychczas najlepsze„ punkty. Gdy linia ci przeszkadza (np. ponieważ chcesz porównywać tylko pojedyncze trade'y), wyłącz ją.

W17 Pigułki filtra „Scene„**GDZIE**

Prawy pasek boczny w oknie dashboard.

**TECHNICZNIE**

Pigułki filtra dla każdej sceny występującej w wczytanych raportach. Własny układ flow, który automatycznie przepakowuje pigułki na kilka wierszy, gdy tylko szerokość zostanie wyczerpana. Aktywne pigułki otrzymują akcentowe tło, nieaktywne neutralne standardowe tło Material. Wybór wielu jest możliwy (semantyka set); gdy żadna pigułka nie jest wybrana, wszystkie sceny są uznawane za „przepuszczone„ — tzn. logika set jest „pusty wybór = wszystko“, nie „pusty wybór = nic„.

**PROSTYMI SŁOWAMI**

Kliknięcie na nazwę sceny filtruje punkty na tylko tę scenę. Wybór wielu możliwy. Puste = wszystkie sceny.

W18 Pigułki filtra „Strategy„**GDZIE**

Pod filtrem Scene na pasku bocznym.

**TECHNICZNIE**

Dokładnie jak W17, ale dla strategii treningu — typowo dwie wartości „classic„ i „mcmc“, wyprowadzone z pola Strategy z JSON-raportów benchmark. Pomocne, gdy zmieszasz raporty obu strategii i chcesz widzieć tylko jeden rodzaj (np. „pokaż tylko MCMC, ponieważ Classic już wykluczyłem„).

**PROSTYMI SŁOWAMI**

Filtr według Classic lub MCMC. Standardowo oba aktywne.

W19 Pigułki filtra „Mip-Splatting„ GDZIE

Pod filtrem Strategy na pasku bocznym.

 TECHNICZNIE

Filtr trójwartościowy (zamiast set jak W17/W18): „All„ / „On„ / „Off„. Tło: Mip-Splatting został w Fazie Q1.5 oceniony jako eksperymentalna poprawa wieloskalowa i finalny werdykt brzmiał „brak konsekwentnego ładnego wygranego; zachować jako opt-in Flag“. Gdy robisz porównania Mip-on/off, często chcesz móc bardzo ostro rozdzielić. Stąd dedykowany filtr ternarny ze stanami „przepuść wszystko„, „tylko Mip włączone“, „tylko Mip wyłączony„. Sekcja paska bocznego jest pokazywana tylko, gdy w zestawie danych jest co najmniej jeden raport Mip i co najmniej jeden bez Mip (inaczej filtrowanie nie ma sensu).

 PROSTYMI SŁOWAMI

Gdy chcesz porównywać Mip-Splatting on/off, tutaj trójdzielny filtr. Inaczej ignoruj.

W20 ChipButton (przełącznik filtra, all/on/off) GDZIE

Komponent pomocniczy, używany w W17/W18/W19.

 TECHNICZNIE

Minimalistyczny wrapper przycisku. Treść: tekst etykiety z grubością Caption i padding 10 poziomo / 5 pionowo. Tło warunkowe: gdy aktywne → kolor akcentu aplikacji z białym tekstem; w przeciwnym razie neutralne standardowe tło Material z czarnym tekstem. Forma to Capsule (pigułka). Plain-Button-style, aby Material Capsule nie był pokrywany przez systemową ramkę.

 PROSTYMI SŁOWAMI

Same okrągłe przyciski filtra. Optycznie jak tag iOS.

W21 Chart (Pareto-Scatter)

GDZIE

Środkowy obszar dashboardu.

TECHNICZNI

Wykres Swift-Charts z dwiema warstwami: 1. punkt na raport — pozycja z wybranych metryk X i Y, kolor według Strategy, symbol według statusu Mip. Rozmiar symbolu normalnie 80, podświetlonego 200 (gdy ID odpowiada aktualnie hoverowanemu raportowi). 2. linia dla frontu Pareto, tylko gdy przełącznik włączony.

Overlay wykresu: przezroczysty prostokąt rejestruje ruch myszy; na klatkę ustalana jest euklidesowo najbliższa pozycja punktu w ramce plot i hoverowany raport aktualizowany, jeśli dystans leży poniżej 24 px (inaczej resetowany). Tak otrzymujesz tooltip bez klikania — wystarczy hover.

PROSTYMI SŁOWAMI

Właściwy wykres scatter. Każdy punkt to przebieg treningu. Hover dla tooltipu szczegółów.

W22 Tooltip (szczegóły hover)

GDZIE

Pod wykresem, wyświetlany przy hover.

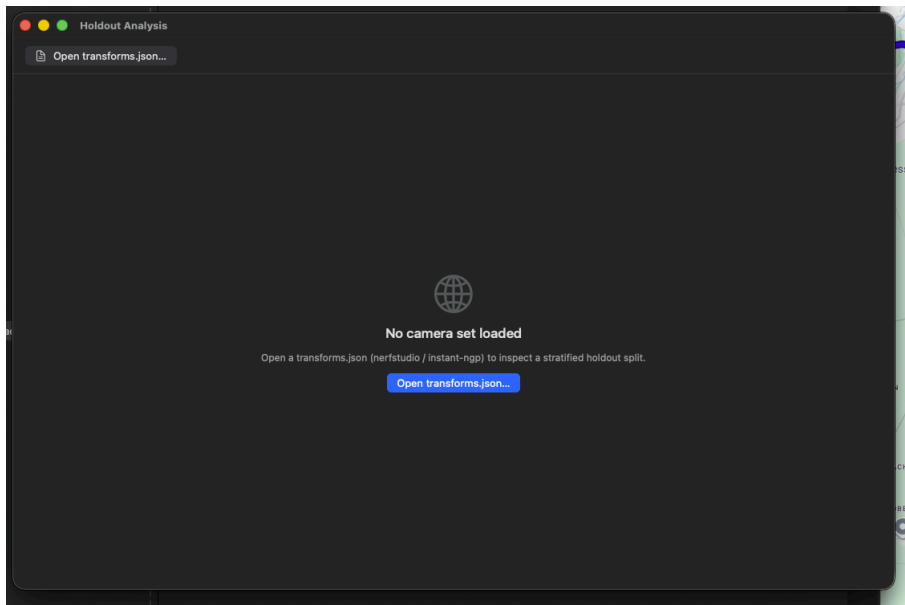
TECHNICZNI

Stos poziomy: nazwa sceny (Headline), tag Strategy (Caption), linia podziału, potem metryki PSNR/SSIM/LPIPS/Gs/Time w małych grupach pionowych (label + wartość monospaced). Gdy Mip był aktywowany, dodatkowo tag „Mip„ Capsule w kolorze akcentu. Tło półprzezroczyste rozmycie, zaokrąglony prostokąt z promieniem 8 pt. Wyświetlany tylko, gdy mysz faktycznie jest nad punktem. Znika automatycznie przy opuszczeniu.

PROSTYMI SŁOWAMI

Karta szczegółów na dole, gdy najedziesz myszą na punkt. Pokazuje wszystkie metryki jakości i konfigurację przebiegu jednocześnie.

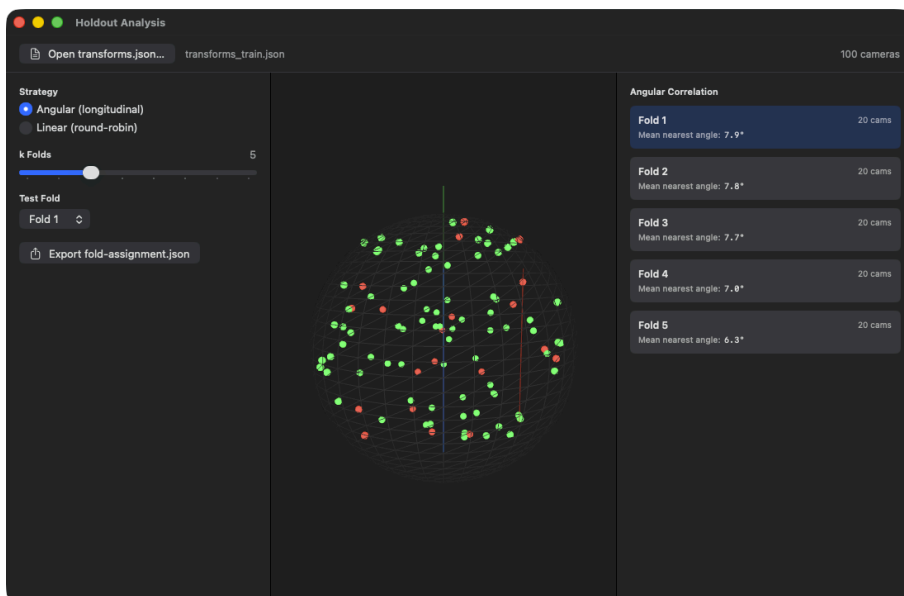
Holdout Analysis (W23–W29)



Rysunek 22: Holdout Analysis — pusty stan przed wczytaniem transforms.json

Pusty stan z empty-state i call-to-action „Open transforms.json...”. Akceptuje format NeRF-Studio i Instant-NGP.

Pusty stan (po pierwszym otwarciu) — znaczniki kamer pojawiają się, gdy tylko wczytany jest `transforms.json`, zobacz następne zdjęcie.



Rysunek 23: Globe Holdout z 100 kamerami NeRF-Blender-Mic, 5 foldów po 20 kamer, strategia Angular aktywna

CO WIDAĆ NA OBRAZIE Nagłówek pokazuje wczytany plik (`transforms_train.json`) i liczbę kamer („100 cameras”). Lewy pasek boczny: selektor strategii z dwiema opcjami — Angular (longitudinal) aktywne (wyrównuje foldy według sektorów długości/szerokości na

sferze, dzięki czemu każdy fold testowy jest geometrycznie gęsty) vs Linear (round-robin) (oparte na kolejności, każda k-ta klatka jako zestaw testowy). Suwak k-foldów stoi na 5, selektor folda testowego na fold 1. Przycisk Export generuje `fold-assignment.json` dla Nerfstudio/Instant-NGP. Środkowy panel: projekcja globusa 3D wszystkich 100 kamer — zielone punkty = Train, czerwone punkty = bieżący fold testowy (fold 1 z 20 kamerami). Prawy pasek boczny (Angular Correlation): na fold 20 kamer + Mean Nearest Angle (fold 1: 7.9°, fold 2: 7.8°, fold 3: 7.7°, fold 4: 7.0°, fold 5: 6.3°) — mniejsza wartość oznacza, że kamery w obrębie tego folda leżą ciasno koło siebie, czyli rozdział Holdout jest przestrzennie koherentny.

Co to jest: Wizualizator 3D twojego układu kamer z logiką cross-validation. Wczytujesz `transforms.json` (standardowy format Nerfstudio / Instant-NGP dla pól kamer), aplikacja czyta wszystkie kamery, rzutuje ich kierunki patrzenia na sferę jednostkową i pokazuje je jako małe kulkowe znaczniki na wirtualnym globusie. Następnie dzieli kamery na `k` foldów (według wybranej strategii: angular lub linear), zaznacza zielonym udział treningowy a czerwonym udział testowy (Holdout), i oblicza na fold score Angular Correlation, mówiący ci jak daleko fold testowy leży w przestrzeni kątów widzenia od folda treningowego.

📍 Gdy chcesz robić ocenę holdout — czyli: jak dobrze twój model generalizuje na niewidziane kąty widzenia? Standardem w treningu jest „every-8th view jako Holdout„ (konwencja Mip-NeRF360), ale to bardzo liniowy podział. Gdy twoje obrazy są np. zgrupowane czasowo (najpierw jedna strona obiektu, potem druga), to „every-8th” nie jest reprezentatywne — losowa pozycja sekwencji ląduje w teście, ale wszyscy jej sąsiedzi są w treningu, to za proste. Przy „angular„ stratyfikuje się zamiast tego po przestrzeni kątów widzenia: każdy fold zawiera kamery ze wszystkich obszarów orbity, dzięki czemu test naprawdę testuje luki w generalizacji.

JAK INTERPRETOWAĆ Angular vs Linear: - Angular (standard): dzieli kamery według kąta longitudinalnego (współrzędna ϕ wokół osi Y) na `k` równych sektorów. Fold 0 to kamery z $\phi \in [0^\circ, 360/k^\circ)$, fold 1 następnie i tak dalej. Zaleta: każdy fold pokrywa wycinek orbity; fold testowy jest przestrzennie kompaktowy, ale szeroko rozłożony po zestawie danych świata. Dobre dla klasycznych ujęć orbitalnych. - Linear (round-robin): indeks folda = $(\text{image_index} \bmod k)$. To prosty podział „every-k-th„. Działa, gdy kolejność obrazów NIE ma biasu przestrzennego (np. losowo posortowane ujęcia dronowe). Działa źle, gdy obrazy klastrują się czasowo.

W globusie 3D od razu widzisz: zielone punkty (trening) i czerwone punkty (test). Gdy czerwone punkty wszystkie klastrują w jednym rogu, holdout jest słaby (brak dobrego testu generalizacji). Gdy leżą równomiernie między zielonymi, jest dobry. Score Angular Correlation na fold (prawy pasek boczny, w stopniach) mówi dodatkowo: mniejsza wartość = test jest bliski treningu (każda kamera testowa ma bliską kamerę treningu, łatwy test); większa wartość = test jest daleko od treningu (trudniejsza generalizacja).

📍 Nagrałeś swoją scenę Truck z 251 obrazami, eksportujesz przez pozycję menu M33 (Export SfM transforms.json) plik nerfstudio. Otwórz okno Holdout (🔗⌘H), wczytaj JSON przez „Open transforms.json...“, spójrz na globus. `k=5` (domyślnie) daje ci 5 foldów. Kliknij na „Fold 3” — zobacz czy czerwone znaczniki są w miarę równomierne. Jeśli tak: „Export fold-assignment.json„, umieść wyeksportowany plik w folderze raportów,

a przy następnym przebiegu treningu z `--benchmark` (lub odpowiednimi ustawieniami Inspector) dokładnie ten podział foldów jest używany jako Holdout testowy — zamiast standardu „every-8th”.

W23 Przycisk „Open transforms.json...”

GDZIE

Pasek narzędzi u góry po lewej.

TECHNICZNIE

Otwiera dialog wyboru pliku ograniczony do plików JSON. Po potwierdzeniu moduł Holdout ładuje plik. Loader parsuje zarówno format nerfstudio (intrinsic kamery plus listę klatek ze ścieżką obrazu i macierzą transformacji), jak i format instant-ngp (taka sama budowa). Na klatkę kierunku patrzenia jest ekstrahowany z macierzy transformacji (oś z lokalnej bazy kamery) i zapisywany. Gdy parsowanie zawiedzie, wyświetlana jest wiadomość błędu w obszarze statusu.

Także przez CLI: `--holdout-file /sciezka/do/transforms.json` startuje okno bezpośrednio z wczytanym plikiem.

PROSTYMI SŁOWAMI

Ładuje twój JSON póź kamer. Standardem są eksporty Nerfstudio i Instant-NGP. RadianceKit sam może eksportować transforms.json przez menu → Export → SfM.

W24 Selektor „Strategy„ (angular/linear)

GDZIE

Lewy pasek boczny, u góry.

TECHNICZNIE

Selektor radiowy z dwiema opcjami: Angular i Linear. Zmiana strategii automatycznie wyzwala ponowne obliczenie foldów. Kierunki patrzenia to lista wektorów jednostkowych 3D na sferze; strategia Angular rzutuje je na kąt longitudinalny ϕ i sortuje, strategia Linear robi po prostu podział modulo po indeksie klatki.

PROSTYMI SŁOWAMI

Angular dla równomiernych ujęć orbitalnych (standard, bezpieczne), Linear tylko gdy twoje obrazy nie klastrują przestrzennie.

W25 Suwak „k Folds„ **GDZIE**

Lewy pasek boczny, pośrodku.

 **TECHNICZNIE**

Suwak od 3 do 10, krok 1. Przy zmianie obliczenie foldów jest automatycznie inicjowane na nowo, dzięki czemu lista foldów, indeksy Training/Test i score na fold są od razu nowo obliczane. Wybrana wartość jest wyświetlana jako tekst monospaced-digit po prawej obok etykiety.

Reguła kciuka: $k=5$ to standard (daje ci 20% test na fold, to zwykłe dla cross-validation). $k=10$ gdy masz bardzo dużo danych i potrzebujesz więcej foldów do siły statystycznej. $k=3$ gdy masz mało danych.

 **PROSTYMI SŁOWAMI**

Ile foldów w podziale. 5 to standard i pasuje prawie zawsze.

W26 Selektor „Test Fold„ **GDZIE**

Lewy pasek boczny, pod suwakiem k.

 **TECHNICZNIE**

Selektor menu. Opcje są dynamicznie $0..<k$, etykieta „Fold 1” do „Fold N„ (czyli 1-indeksowane w UI, 0-indeksowane wewnątrznie). Gdy wcześniej wybrany indeks $\geq k$ (np. ponieważ zredukowałeś k z 10 do 5), jest automatycznie resetowany na 0. Wybrany fold testowy jest na globusie prezentowany czerwono, wszystkie inne zielono.

 **PROSTYMI SŁOWAMI**

Który fold jest obecnie foldem testowym. Możesz przeklikać i widzisz, jak każdy pojedynczy fold wygląda na globusie.

W27 Przycisk „Export fold-assignment.json„

GDZIE

Lewy pasek boczny, na dole.



TECHNICZNIE

Otwiera dialog zapisu z domyślną nazwą pliku `fold-assignment.json`. Po potwierdzeniu moduł Holdout koduje bieżący podział w schemat JSON (zapisanie fold na klatkę plus meta-blok strategii). Plik ten można potem przy następnym treningu z `--benchmark` przekazać, dzięki czemu ten sam Holdout jest używany do finalnej oceny metryki. Błędy zapisu wyświetlane jako tekst błędu; sukces zielonym tekstem jako „Saved to (filename)„.

PROSTYMI SŁOWAMI

Zapisuje bieżący podział Train/Test jako JSON. Plik ten możesz potem przy treningu bezpośrednio przekazać, dzięki czemu ten sam zestaw testowy zostanie ponownie użyty.

W28 SCNView (globus kamer 3D)

GDZIE

Środkowy panel w oknie Holdout.



TECHNICZNIE

Widok globusa SceneKit. Scena składa się z: kuli Wireframe (promień 1.0, 36 segmentów, ciemnoszary), trzech kolorowych kikutów osi (czerwony/zielony/niebieski dla X/Y/Z, każdy 1.2 długości), i na kamerę małej kuli znacznika (promień 0.03) na odpowiedniej pozycji kierunku patrzenia na sferze jednostkowej (lekko na zewnątrz, aby nie zniknąć w kuli wireframe). Znaczniki przy każdej zmianie folda NIE są przebudowywane — przebudowa jest potrzebna tylko, gdy zmienia się lista klatek (czyli nowy JSON jest wczytywany). Zamiast tego na update działa in-place aktualizacja kolorów materiału: czerwony dla indeksów Test, zielony dla treningu, jasnoszary gdy ani jedno ani drugie. Tak ticki suwaka pozostają wydajne nawet przy $N > 1000$ kamer.

Sterowanie kamerą jest włączone — możesz myszą obracać globus, zoomować, panować. Oświetlenie sprawia, że znaczniki nie wyglądają płasko. Tło ciemnoszare.

PROSTYMI SŁOWAMI

Globus 3D z pozycjami kamer. Zielony = trening, czerwony = test, jasnoszary = nieprzypisany (nie występuje, wszystkie kamery gdzieś należą). Myszą możesz obracać globus i zoomować.

W29 FoldCard (Tap to Select Fold)**GDZIE**

Prawy pasek boczny, sekcja „Angular Correlation„.

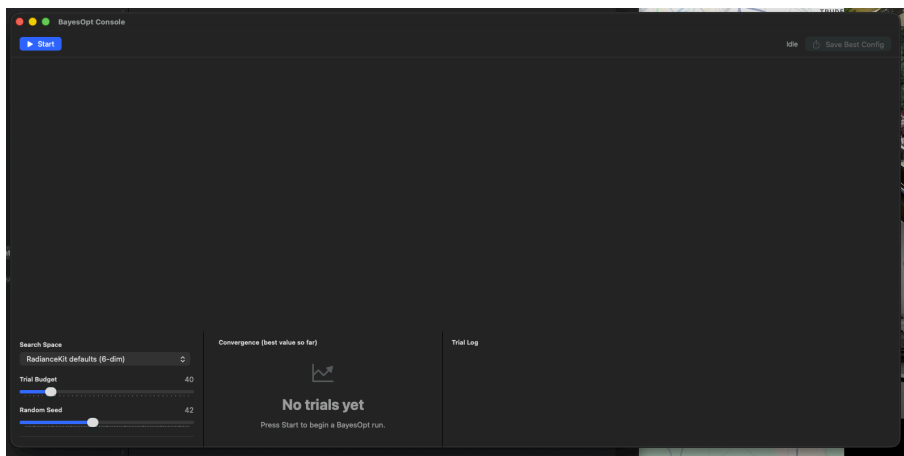
TECHNICZNIE

Na fold widok karty — zaokrąglony prostokąt z promieniem 6 pt, padding 10, pionowy układ z dwoma wierszami (u góry „Fold N„ + liczba kamer, na dole „Mean nearest angle:” + wartość w stopniach). Kolor tła warunkowo: aktywny fold = kolor akcentu półprzezroczysty, nieaktywne = neutralne standardowe Material. Stuknięcie wybiera fold i globus przebarwia się na żywo.

Score „Mean nearest angle„ to średni najmniejszy kąt na kamerę testową do najbliższej kamery treningu (wewnętrznie liczony w radianach, w UI wyświetlany w stopniach).

PROSTYMI SŁOWAMI

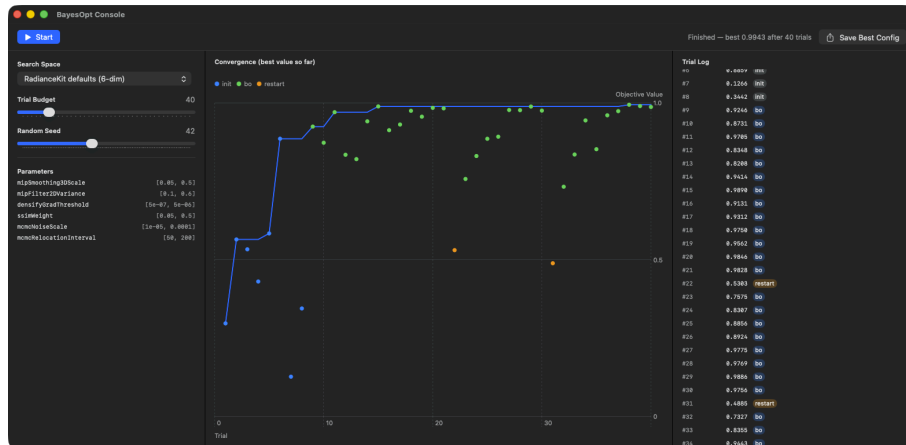
Na fold mała karta po prawej z liczbą kamer i średnim odstępem do najbliższej kamery treningu. Kliknięcie na nią wybiera ten fold jako test.

BayesOpt Console (W30–W39)

Rysunek 24: Konsola BayesOpt — pusty stan przed startem próby

Pusty stan z selektorem Search-Space (RadianceKit defaults (6-dim)), suwakiem Trial-Budget (domyślnie 40), Random-Seed (42) i trzema pustymi panelami dla wykresu Convergence, Trial Log i listy parametrów Search-Space.

Pusty stan (po pierwszym otwarciu) — wykres Convergence i tabela Trial wypełniają się, gdy tylko przebieg zostanie uruchomiony, zobacz następane zdjęcie.



Rysunek 25: Konsola BayesOpt po 40 próbach — wykres Convergence rośnie stromo do próby 15, Best Value 0.9943, Trial Log z tagami init/bo/restart

CO WIDAĆ NA OBRAZIE Status u góry po prawej „Finished — best 0.9943 after 40 trials”. Lewy pasek boczny: selektor Search-Space na RadianceKit defaults (6-dim), Trial-Budget 40, Random Seed 42. Lista parametrów pokazuje sześć hiperparametrów do strojenia z ich zakresami wartości: mipSmoothing3DScale [0.05, 0.5], mipFilter2DVariance [0.1, 0.6], densifyGradThreshold [5e-07, 5e-06], ssimWeight [0.05, 0.5], mcmcNoiseScale [1e-05, 0.0001], mcmcRelocationInterval [50, 200]. Pośrodku: wykres Convergence (X = indeks próby 1–40, Y = wartość Objective 0–1) — szare punkty = próbki początkowe (LHS), niebieskie punkty = akwizycja BayesOpt, pomarańczowe punkty = próby restartu (#22 i #31). Linia najlepszej wartości rośnie stromo do próby ~7, potem tylko marginalna poprawa do próby 15, od tego momentu płaski plateau przy 0.99+. Prawy pasek boczny: Trial-Log #1–#34 z score + tag (init/bo/restart). Przycisk Save-Best-Config u góry po prawej zapisuje `bayesopt-best.json`.


Co to jest: Konsola Bayes-optimizacji do poszukiwania hiperparametrów. Bayes-Opt to automatyczna procedura próbująca jak najmniej eksperymentów znaleźć optymalny punkt nieznannej funkcji — typowo: „która kombinacja z mcmcMaxGaussians, capMultiplier, ssimWeight i gradThreshold daje najlepszą PSNR dla mojej klasy sceny?”. Zamiast siatki $6^4 = 1296$ prób Bayes-Opt próbuje około 40–100 świadomych prób i dociera blisko optimum.

Ważne: Bieżąca dostarczona w aplikacji wersja nie przeprowadza optymalizacji wobec rzeczywistych przebiegów treningu (to trwałoby dni), lecz wobec syntetycznego demo-Objective — wielomodalnego krajobrazu z charakterem hill-climbingu plus lekkim noisem. To jest tak celowo: okno ma pokazywać ci zachowanie optymalizatora (przebieg zbieżności, punkty próbek, Best-So-Far) i pozwolić ci zrozumieć definicje Search-Space. Dla rzeczywistych przebiegów BayesOpt sterowanych treningiem (jak w Fazie Q7 dla ustawień Scene-Class) używany jest osobny offline workflow CLI; okno jest wariantem Live-UI.

Trzy przypadki użycia: 1. Chcesz zrozumieć, jak działa BayesOpt — wtedy uruchom demo-przebieg i obserwuj wykres Convergence. 2. Planujesz nową klasę sceny (np. „akwaria” lub „antyczne meble”), dla której wbudowane 10 ustawień nie pasuje idealnie. Zdefiniuj mentalnie przestrzeń poszukiwań, sprawdź ją tutaj z „Bowl demo”, lub ustawie-

niem „Densify”, eksportuj potem Best-Config jako JSON i użyj jako punktu startowego dla rzeczywistego przebiegu treningu. 3. Chcesz zainspekcionować zdefiniowane w paczce RKBayesOpt domyślne Search-Spaces (Mip-Subset, RadianceKit Defaults) — są wyszczególnione w panelu parametrów lewego paska bocznego.

JAK INTERPRETOWAĆ - **Wykres Convergence** (środkowa kolumna): Y = najlepsza dotychczas osiągnięta wartość funkcji Objective. X = indeks próby. Początkowo stromo rosnące (BayesOpt próbuje próbki początkowe losowo, niektóre z nich są szczęśliwe), potem coraz bardziej płaskie, ponieważ region bliski-optimum jest wyczerpany. Gdy linia pozostaje płaska przez 20+ prób, możesz przebieg zatrzymać — dalsze próby nic już nie dają. Poszczególne punkty na wykresie to indywidualne wartości prób (czyli nie „best so far”), pokolorowane według fazy: szary = próbka początkowa, niebieski = akwizycja BayesOpt, pomarańczowy = restart. - **Tabela prób** (prawa kolumna): #1, #2, #3, ... z wartością i tagiem fazy. Najlepsza dotychczas próba jest oznaczona żółtą gwiazdką. Z tabeli możesz zidentyfikować najlepszą próbę i jej wartości parametrów obejrzyć później przy eksporcie. - **Inspektor Search-Space** (lewy pasek boczny): pokazuje dla wybranego ustawienia wszystkie nazwy parametrów i ich zakresy poszukiwań `[lo, hi]`. Gdy stoisz przy ustawieniu „RadianceKit defaults (6-dim)”, widzisz np. „densifyGradThreshold `[5e-7, 5e-6]`”, — czyli log-uniform między tymi dwiema wartościami.

 Wybierz ustawienie „RadianceKit defaults (6-dim)”, Trial-Budget 40, Seed 42. Kliknij „Start”. Obserwuj: pierwsze 8 prób jest szarych (próbki początkowe, Latin-Hypercube), następne niebieskie (BayesOpt-akwirowane). Wykres Convergence robi się stromy do próby ~15, potem się spłaszcza. Przy próbie ~30–40 najlepsza wartość się stabilizuje. Kliknij „Save Best Config”, — zostanie zapisany `bayesopt-best.json` z nazwą ustawienia, indeksem próby, wartością i dekodowanymi wartościami parametrów. Ten JSON możesz potem ręcznie wprowadzić do definicji swojego ustawienia.

W30 Przycisk „Start„**GDZIE**

Pasek narzędzi po lewej, w stanie Idle/Finished.

**TECHNICZNI**

Resetuje listę prób, przełącza w stan Running, generuje nowe ID przebiegu (do wykrywania stale przy wielu kliknięciach Start) i tworzy świeżą bramkę Pause. Potem startuje zadanie w tle, wykonujące optimizer jako asynchroniczny strumień. Rozmiar próbek początkowych wynika z $\min(8, \text{budget} / 4 + 1)$ — czyli typowo 8 próbek Latin-Hypercube przy budżecie ≥ 28 , mniej przy małym budżecie. Aktualizacje prób są odbierane przyrostowo i dołączane do listy. Ochrona przed stale: gdy w międzyczasie drugie kliknięcie Start ustawi nowe ID przebiegu, aktualizacje ze starego przebiegu są odrzucane.

Primary-Action-Style dla prominentnego wyglądu przycisku.

**PROSTYMI SŁOWAMI**

Uruchamia świeży przebieg optymalizacji z bieżącą przestrzenią poszukiwań, budżetem i seedem.

W31 Przycisk „Pause„**GDZIE**

Pasek narzędzi po lewej, w stanie Running.

**TECHNICZNI**

Ustawia aktywną bramkę Pause i przełącza w stan Paused. Właściwy efekt: runner czeka w pętli pollingu 50 ms, zanim oceni następną funkcję Objective. To znaczy, że właśnie działająca próba jest doprowadzana do końca (jest syntetyczna i trwa tylko mikrosekundy), ale żadna kolejna próba nie jest inicjowana. Gdy Resume zadziała, idzie dalej tam, gdzie się zatrzymał.

**PROSTYMI SŁOWAMI**

Zatrzymuje przebieg. Bieżące obliczenie dobiega do końca, potem pauzuje.

W32 Przycisk „Stop„ **GDZIE**

Pasek narzędzi po lewej, w stanach Running i Paused.

 **TECHNICZNIE**

Anuluje zadanie runnora, zeruje referencję, zwalnia bramkę Pause (jeśli wciąż pauzowane), i przełącza w stan Finished (gdy istnieją próby) lub Idle (gdy żadne). Już obliczone próby pozostają widoczne w liście — Stop ich nie usuwa. Rola przycisku destrukcyjna pokazuje przycisk na czerwono, ponieważ anuluje przebieg.

 **PROSTYMI SŁOWAMI**

Definitywnie anuluje przebieg. Próby pozostają widoczne, mimo to możesz eksportować Best-Config.

W33 Przycisk „Resume„ **GDZIE**

Pasek narzędzi po lewej, w stanie Paused.

 **TECHNICZNIE**

Zwalnia bramkę Pause i przełącza z powrotem w stan Running. Zadanie runnora już działa (czeka w pętli pollingu); gdy pętla zauważy, że pauza jest zwolniona, działa dalej i startuje następną próbę.

 **PROSTYMI SŁOWAMI**

Kontynuuje pauzowany przebieg.

W34 Przycisk „Save Best Config„ **GDZIE**

Pasek narzędzi po prawej, zawsze widoczny (ale dezaktywowany, gdy nie ma bestTrial).

 **TECHNICZNIE**

Otwiera dialog zapisu z domyślną nazwą pliku `bayesopt-best.json`, ograniczone do JSON. Po potwierdzeniu budowany jest słownik payload: nazwa ustawienia, indeks próby, wartość (score Objective), parametry (słownik dekodowanych nazw parametrów → wartości). Dekodowanie projektuje znormalizowane współrzędne przestrzeni poszukiwań w $[0,1]^d$ z powrotem do oryginalnego zakresu wartości (ze skalami log-uniform / linear / integer odpowiednio). Wyjście JSON jest pretty-printed z posortowanymi kluczami. Przy błędzie zapisu (w bieżącej wersji demo) cicho ignorowane — brak UI błędu, ponieważ to ścieżka demo.

Przycisk pozostaje szary, dopóki żadna próba nie została przeprowadzona.

 **PROSTYMI SŁOWAMI**

Zapisuje wartości parametrów dotychczas najlepszej próby jako JSON. Możesz potem te wartości ręcznie przejąć do swojej konfiguracji ustawienia.

W35 Selektor ustawienia „Search Space„ GDZIE

Lewy pasek boczny, u góry.

 TECHNICZNIE

Selektor menu z czterema opcjami ustawienia: - „RadianceKit defaults (6-dim)„ — kompletna standardowa przestrzeń poszukiwań ze wszystkimi hiperparametrami Q7. - „Mip subset (2-dim)„ — tylko `mipSmoothing3DScale` [0.05, 0.5] log-uniform i `mipFilter2DVariance` [0.1, 0.6] linear. Użyteczne, gdy chcesz stroić Mip-Splatting dla klasy sceny. - „densify-until + ssim-weight + grad-thresh„ — trzy parametry istotne dla densyfikacji (`densifyGradThreshold` log-uniform, `ssimWeight` linear, `densifyUntilIter` integer). - „Bowl demo (1-dim)„ — pedagogiczna przestrzeń poszukiwań pojedynczego parametru do dem „jak działa BayesOpt„.

Gdy przebieg jest aktywny, przestrzeni poszukiwań nie można zmienić (zmyliłaby optimizer).

 PROSTYMI SŁOWAMI

Którą przestrzeń poszukiwań hiperparametrów BayesOpt przeszukuje. Standardem jest „RadianceKit defaults„. Do celowanych prób strojenia Mip „Mip subset„. Do zrozumienia, jak działa BayesOpt, „Bowl demo„.

W36 Suwak „Trial Budget„ GDZIE

Lewy pasek boczny, pod selektorem Search-Space.

 TECHNICZNIE

Suwak od 10 do 200, krok 5. Domyślnie 40. To znaczy: BayesOpt może zrobić maksymalnie N prób. Pierwsze są próbkami początkowymi (Latin-Hypercube), reszta to rzeczywiste próby BayesOpt. Reguły kciuka w praktyce: przestrzeń poszukiwań z d wymiarami potrzebuje około 10d do 20d prób dla dobrego optimum. Przy 6-dim Defaults więc 60–120, przy 2-dim Mip-Subset 20–40, przy 1-dim Bowl-Demo 10–20.

Podczas przebiegu suwak jest dezaktywowany.

 PROSTYMI SŁOWAMI

Ile prób optymalizacji maksymalnie. Więcej prób = lepsze rozwiązanie, ale kosztuje więcej czasu. 40 to dobry domyślny dla demo-Objective.

W37 Suwak „Random Seed„**GDZIE**

Lewy pasek boczny, pod suwakiem Budget.

TECHNICZNIE

Suwak od 1 do 100, krok 1. Domyślnie 42. Seed jest przekazywany zarówno do początkowych próbek Latin-Hypercube, jak i do komponentu Noise demo-Objective. Powtarzalność: ten sam seed + ta sama przestrzeń poszukiwań + ten sam budżet daje dokładnie identyczną sekwencję prób. Użyteczne dla „czy wszyscy twoi koledzy otrzymują ten sam przebieg, gdy odtwarzają demo?„. Podczas przebiegu dezaktywowany.

PROSTYMI SŁOWAMI

Steruje generatorem losowości. Ten sam seed = ten sam przebieg — do reprodukcji.

W38 Chart (Convergence)**GDZIE**

Środkowa kolumna okna.

TECHNICZNIE

Wykres Swift-Charts z dwiema warstwami: 1. linia „best-value-so-far„ na próbę — monotonicznie rosnąca lub pozostająca taka sama krzywa w kolorze akcentu. 2. punkt na próbę z indywidualną wartością Objective, pokolorowany według fazy. Rozmiar symbolu 40. Trzy etykiety fazy: „init“ (szary), „bo„ (niebieski), „restart“ (pomarańczowy).

Mała legenda pokazuje kolory faz u góry po lewej. Gdy lista prób jest pusta (przed pierwszym Startem), wyświetlany jest zamiast tego empty-state z ikoną wykresu i wskazówką „Press Start to begin a BayesOpt run.,„.

PROSTYMI SŁOWAMI

Wykres przebiegu. Ciągła linia to „dotychczas najlepsza znaleziona wartość„; punkty to poszczególne próby. Gdy linia pozostaje długo płaska, BayesOpt znalazł optimum.

W39 Tabela (Trial Log)

GDZIE

Prawa kolumna okna.

TECHNICZNIE

Obszar przewijania z lazy stackowanymi wierszami prób. Na wiersz horyzontalny stos: numer próby (3-cyfrowo monospaced, po lewej), wartość (monospaced, wyrównana do prawej, 70 pt szerokości), tag fazy (Capsule, wypełniony kolorem fazy przy 25% opacity), opcjonalnie żółta gwiazdka, gdy ta próba jest aktualnie najlepsza. Mechanizm auto-scroll automatycznie skacze na koniec, gdy tylko nowa próba dochodzi — dzięki czemu możesz przebieg na żywo czytać na dole ekranu, bez samodzielnego przewijania.

PROSTYMI SŁOWAMI

Tabela wszystkich prób. Wartość, faza, gwiazdka dla najlepszej. Przewija się automatycznie, nowe próby pojawiają się u dołu.

Okno główne: przebieg straty i liczba gaussianów (I39–I41, odsyłacz)

Trzy wyświetlenia Inspectora w oknie głównym zasługują na własne wyjaśnienie, ponieważ podczas działającego treningu są stale widoczne i istnieją ważne reguły kciuka, kiedy przebieg wygląda zdrowo. Wyświetlenia znajdują się w Inspectorze pod sekcją „Loss Chart„ (zobacz rozdział 2 — Inspector) i uzupełniają analizę Holdout z okna pomocniczego powyżej.

Kiedy krzywa Loss jest zdrowa? Zdrowa krzywa Loss pokazuje trzy fazy: (1) **Warmup** — pierwsze 200–500 iteracji Loss spada stromo z wysokich (typowo 0.15–0.25 dla L1+SSIM łączonego w zależności od sceny) na około połowę. Gdy Loss w tej fazie NIE spada, zwykle wejście jest błędne (obrazy zepsute, pozy SfM złe, liczba gaussianów początkowych zbyt mała). (2) **Densyfikacja** — między ~500 a densifyUntilIteration (klasycznie 15K, MCMC do 20K lub 25K) Loss spada dalej, często z małymi skokami w dół, gdy operacje densify wstawiają nowe gaussiany a optimizer je wykorzystuje. Liczba gaussianów rośnie w tej fazie. (3) **Refinement** — potem Loss biegnie w spłaszczający się tail. Typowe wartości końcowe: Truck Tanks-&- Temples z P4 Quality ląduje przy $L1 \approx 0.023$, Horse z Full Classic V546 przy $L1 \approx 0.0230$, sceny Outdoor-Mip-NeRF360 często gorzej (0.04–0.07).

Co oznacza plateau? Plateau (krzywa Loss biegnie horyzontalnie przez kilka tysięcy iteracji) ma dwie interpretacje: (a) model jest zbieżny, dalszy trening niczego więcej nie daje — to dobry przypadek. (b) model jest stuck (minimum lokalne, słaba informacja gradientu, cap na limicie bufora) — zły przypadek. Oba wyglądają na wykresie identycznie. Rozróżnienie: spójrz na liczbę gaussianów. Gdy też jest płaska i blisko cap MCMC (np. 150K z 150K przy `.fullMCMC`), jesteś przy limicie — albo cap zwiększyć, albo zaakceptować plateau. Gdy liczba gaussianów wciąż rośnie, ale Loss nie spada, to stuck.

Kiedy przerwać vs trenować dalej? Reguła kciuka: 10K iteracji bez poprawy Min-Loss → przerwać, dalsze iteracje są zmarnowane. Wcześniej: możesz przez Cmd+T (menu Training → Continue Training → +5K iterations) jeszcze podłączyć przedłużenie, gdy widzisz graniczną poprawę. Uwaga: przy MCMC plateau jest często prawdziwe — cap to naturalna granica.

Plateau liczby gaussianów to NIE sygnał „gotowe”. Oznacza tylko, że MCMC osiągnęło cap lub że densyfikacja Classic jest wyczerpana. Prawdziwe pytanie „gotowe” zadaje dopiero analiza Holdout — PSNR/SSIM/LPIPS na niezależnym zestawie testowym, oceniane w oknie Holdout (W23–W29) lub przez flagę `--benchmark`.

PSNR/Holdout to prawda, Loss tylko proxy. Loss to metryka względna: spada, gdy twój model dostosowuje się do widoków treningowych. Niski Loss nie oznacza jednak automatycznie dobrego modelu — gdy model wyuczył się obrazów treningowych na pamięć (overfitting), Loss byłby mały, ale PSNR na niewidzianych widokach (Holdout) byłaby zła. Dlatego: do finalnej oceny jakości zawsze patrz na metryki Holdout, nie na końcowy Loss sam w sobie.

Pudełko reguł kciuka

- User Guide i Keyboard Shortcuts to statyczna pomoc — przy pytaniach hasłowych szybko, do głębi używać tego podręcznika.
- Manage Storage otworzyć, gdy tylko dysk spadnie poniżej 10% wolnego miejsca. Logi i staging importów to typowi winowajcy.
- Pareto Dashboard sensowne dopiero po co najmniej trzech lub czterech raportach treningu. Oś X = koszty (Time / Gs), oś Y = jakość (PSNR / SSIM). Front Pareto pokazuje efektywne kombinacje.
- Holdout Analysis używać, zanim publikujesz z innymi benchmarki PSNR — zapewnia ci, że twój zestaw testowy jest naprawdę reprezentatywny.
- BayesOpt Console jest przede wszystkim narzędziem do nauki i inspekcji definicji przestrzeni poszukiwań. Do rzeczywistego strojenia hiperparametrów sterowanego treningiem używać offline workflow CLI.
- Plateau Loss i plateau liczby gaussianów są interpretowane oddzielnie. Limit cap to nie sygnał „gotowe”. Prawdziwą jakość mierzy tylko Holdout-PSNR.
- 10K iteracji bez poprawy Min-Loss → trening zatrzymać.

ROZDZIAŁ

Rozdział 6 — Konfiguracja treningu

```
preview-preset.json
{
  "id": "00000000-0000-0000-0000-000000000002",
  "name": "Preview",
  "category": "classic",
  "version": 1,
  "createdAt": "2026-05-27T22:54:00Z",
  "description": "Fast preview training - 5K iterations, 50% render scale, classic densification.",
  "trainingConfig": {
    "maxIterations": 5000,
    "densifyUntilIteration": 3500,
    "ssimWeight": 0.20,
    "renderScale": 0.50,
    "strategy": "classic",
    "cameraAlignment": "applePhotogrammetry",
    "densifyGradThreshold": 2.0e-06,
    "opacityResetInterval": 3000,
    "minOpacity": 0.005,
    "postCompactification": true,
    "perceptualLoss": 0.0,
    "metalFXUpscaling": false,
    "mpsLanczosScaling": false,
    "skyMasking": false,
    "midTrainingFloaterCleanup": true,
    "scaleRegularization": false
  }
}
```

Rysunek 26: Preview-Preset als JSON exportiert + in TextEdit angezeigt — Felder id/name/category/version/createdAt/description, trainingConfig mit allen relevanten Parametern (maxIterations 5000, densifyUntilIteration 3500, ssimWeight 0.20, renderScale 0.50, strategy classic, cameraAlignment applePhotogrammetry, densifyGradThreshold 2.0e-06, opacityResetInterval 3000, minOpacity 0.005, sechs Bool-Toggles)

WAS IM BILD ZU SEHEN IST Ein typischer Preset-JSON-Export. Top-Level-Felder: `id` (UUID), `name`, (`classic` | `mcmc` | `sceneClass` | `custom`), (Schema-Version), (Timestamp), (Free-Text). Verschachteltes `-Objekt` enthält die für Reproduzierbarkeit kritischen Parameter — beim Import wird der gesamte Block in die `TrainingConfig`-Struktur deserialisiert, und Defaults aus der App-Version füllen die Felder, die in der JSON fehlen (z. B. nach App-Update). Wer ein Preset an einen anderen Mac übergibt, schickt einfach diese JSON-Datei rüber.

Die `TrainingConfig`-Struktur ist das Herzstück jedes Trainingslaufs in RadianceKit. Sie versammelt jeden Parameter, der das Training beeinflusst — von der maximalen Iterationszahl über die acht Lernraten bis hin zu den Spezialfeldern für MCMC, Mip-Splatting, das Curriculum und die scene-aware Cap-Logik. Du bearbeitest sie in der Sidebar im Bereich Trainings-Konfigurations-Sektion (Expert View), speicherst sie als

Preset oder reichst sie als JSON-Export an einen anderen Mac weiter. Beim Training wird genau dieses Objekt eingefroren und ans GPU-Backend gegeben.

Dieses Kapitel ist Referenz-Material für Power-User und Skript-Autoren. Es listet alle 81 öffentlichen Felder, die 9 statischen Presets und die eine öffentliche Methode. Quelldatei ist `TrainingConfig.swift` — bei Zweifeln gilt der dort hinterlegte Doc-Comment und der Initializer-Default als Source-of-Truth.

HINWEIS · UI VS. PRESET/CLI

Nur 12 der 81 Felder haben einen direkten Schieberegler, Toggle oder Picker im Inspector (sandboxed App-Store-Build): **T1, T2, T17, T20, T22, T38, T56–T58, T60, T61, T73**. Die restlichen 69 Felder werden über das gewählte **Preset** (Kapitel 7) gesetzt und lassen sich nur per **CLI-Flag** (siehe Kapitel 5) direkt überschreiben. Diese Trennung ist Absicht: Defaults bleiben stabil und produktionserprobt, Power-User haben dennoch eine Escape-Luke. Wenn ein Feld dich besonders interessiert: schau zuerst in Kapitel 2 (Inspector) und Kapitel 5 (CLI) nach, ob du es ohne JSON-Basterei erreichst.

Inhaltsverzeichnis:

1. Iteration (T1–T2)
2. Learning Rates (T3–T10)
3. Densification — Classic (T11–T16)
4. Loss (T17–T20)
5. SH-Degree-Progression (T21)
6. Performance (T22–T25)
7. Diagnose und Punktwolken-Vorbereitung (T26–T30)
8. Regularisierung (T31–T37)
9. Refinement (T38–T44)
10. Sky-Dome (T45–T48)
11. Adam + LR-Schedule (T49–T55)
12. Post-Processing + Apple AI (T56–T60)
13. MCMC-Densification (T61–T73)
14. Mip-Splatting (Q1.5) (T74–T76)
15. Adaptive Densification (Q5) (T77–T79)
16. Curriculum (Q6) (T80–T81)
17. Statische Presets (TP1–TP9)
18. Methode:
19. Welches Feld wofür? (Cheat-Sheet)
20. Gefährliche Felder

Iteration (T1–T2)

T1 maxiterations

SZCZEGÓŁY

Default: 30 000 (Initializer), 35 000 (`.full`), 200 000 (`.fullMCMC`) **Range:** 1 000 – 500 000 (UI-Slider), keine harte obere Grenze in der Logik **Defined in:**

TECHNICZNI

Gesamtzahl der Trainings-Iterationen, die das Backend durchläuft. Eine Iteration bezeichnet ein Forward-Render einer einzelnen Trainingskamera, einen Backward-Pass über alle Loss-Komponenten (L1 + SSIM + optionale Regularisierungen + Sky-Mask) und einen Adam-Optimizer-Schritt. Diese Zahl wirkt direkt auf die anderen Schedules: Position-Lernrate folgt einer Cosine-Annealing-Kurve von 0 bis entweder `T1` selbst oder bis `T49 positionLRScheduleEndIteration`; Densification stoppt bei `T2 densifyUntilIteration`; MCMC-Noise-Decay endet bei `T69 mcmcNoiseDecayEnd`; SH-Degree-Upgrades passieren an den drei in `T21` definierten Marken. Bei klassischer Densification liegt der empirisch ermittelte sweet spot bei 20 000–35 000 Iterationen (Sessions 1–32, V546-Tests), bei MCMC bei 60 000–200 000 (V534). Eine drastische Erhöhung über die in Preset hinterlegten Werte hinaus bringt selten zusätzliche Qualität — Adam-Momentum sättigt, und ohne LR-Decay-Ende stagniert der Loss. Umgekehrt führt Unterschreitung von ~5 000 zu unvollständig konvergierten Geometrien (Density-Control hat zu wenig Zeit zum Klonen/Splitten).

PROSTYMI SŁOWAMI

Wie lange die App rechnet. Mehr Iterationen = besseres Ergebnis, aber irgendwann nicht mehr spürbar besser, dafür viel länger. Die Presets sind so gewählt, dass du ohne nachzudenken einen guten Wert hast: Quick 1 000, Preview 5 000, Balanced 20 000, Quality 35 000, MCMC Quality 200 000. Wenn du selbst dran drehst, gilt: bei MCMC darfst du gerne hoch (100 000–200 000), bei Classic nicht über 40 000 — bringt dann nichts mehr.

T2 densifyUntilIteration

SZCZEGÓŁY

Default: 15 000 (Initializer), 5 000 (`.full`), 160 000 (`.fullMCMC`) **Range:** 0 – **Defined in:**

TECHNICZNI

Iteration, ab der die Densification aufhört. Bis hierhin werden Gaussians über die in `T11–T16` (Classic) oder `T67–T70` (MCMC) parametrisierten Regeln geklont, gesplittet und geprunt; danach bleibt die Gaussian-Anzahl konstant und nur noch Positionen, Rotationen, Skalen, Opazitäten und SH-Koeffizienten werden optimiert (Refinement-Phase). Im 3DGS-Originalpaper liegt der Wert bei 50 % von `T1`, in RadiancKits `.full`-Preset bei nur ~14 % (5 000 von 35 000) — Folge der V310/V338-Experimente, die zeigten, dass nach 5 000 Iterationen weitere Densifizierung das Resultat eher verschlechtert (mehr Floaters, mehr Speicherbedarf, kein Qualitätsgewinn). MCMC dagegen läuft die Relocation bis 80 % von `T1` (V504b), weil MCMC keine schädlichen Floaters produziert. Wird `T2` zu klein gewählt (< 1 000), entstehen zu wenige Gaussians; zu groß bei Classic (> 50 % von `T1`) führt zu Overgrowth und RGB-Saturation-Outliers (siehe Outdoor-Overtraining-Findings).

PROSTYMI SŁOWAMI

Bis wann die App neue Gaussians erzeugen darf. Danach wird nur noch verfeinert, was schon da ist. Bei klassischem Training mit 35 000 Iterationen ist hier 5 000 der richtige Wert — alles darüber macht die Szene matschiger. Bei MCMC ist es 80 % der Gesamtiterationen (also 160 000 bei 200 000-Lauf). Wenn du Quality-Preset änderst, lass dieses Feld lieber in Ruhe.

Learning Rates (T3–T10)

T3 positionLearningRate

📄 SZCZEGÓŁY

Default: 0.00016 **Range:** 1e-7 – 1e-3 (empfohlen)

Defined in:

🔧 TECHNICZNI

Adam-Lernrate für die XYZ-Position jeder Gaussian zu Beginn des Trainings (Iteration 0). Folgt einer Cosine-Annealing-Kurve und sinkt im Laufe des Trainings auf T4 `positionLearningRateFinal`. Der Default 0.00016 stammt aus dem 3DGS-Originalpaper (Kerbl et al. 2023) und ist in RadianceKit auch bei Erhöhung der Bildauflösung nicht zu skalieren — die Position bewegt sich im Welt-Koordinatensystem, nicht im Pixelraum. Eine deutliche Erhöhung (> 0.0005) bewirkt, dass Gaussians über lange Distanzen springen und der Loss instabil wird; Werte deutlich darunter (< 0.00005) führen dazu, dass falsch initialisierte Punktwolken nie ihren Platz finden. V414 testete eine Verdopplung des Init-Werts → 16.8 % schlechterer L1-Loss; die V544a-Tunings bestätigten den Paper-Default als optimal. Beachte: bei `.fullMCMC` lassen wir diesen Wert bewusst beim Default — MCMC braucht konstante Lernraten für seine Relocation-Logik, daher bringt das Tunen hier nichts.

💬 PROSTYMI SŁOWAMI

Wie schnell sich die Splatt-Punkte im Raum bewegen dürfen. Der Standardwert ist sehr gut justiert und braucht eigentlich keine Änderung. Nur wenn dir Splats im Bild „herumeiern“, oder eine ganze Ecke fehlt, weil sich nichts hinbewegt, wäre die Lernrate ein Punkt zum Drehen — aber dann passt typischerweise vorher schon etwas anderes nicht (Kameraposen, Initialpunktwolke).

T4 positionLearningRateFinal

SZCZEGÓŁY

Default: 0.0000016 (Initializer + Paper), 0.000016 (`.full`, `.fullMCMC` — 10× höher) **Range:** 0 – **Defined in:**

TECHNICZNE

Endwert der Position-LR-Cosine-Annealing-Kurve. Erreicht wird er entweder bei `T1 maxIterations` oder, falls gesetzt, bei `T49 positionLRScheduleEndIteration`. Der RadianceKit-`.full`-Preset verwendet 0.000016 — also 10× höher als der Paper-Default 0.0000016. V420-Experimente zeigten, dass 0.5× des Final-Werts (0.000008) den Loss um 6.4 % verschlechtert; V414 zeigte, dass 2× Init-Wert ihn um 16.8 % verschlechtert. Der hohe Final-Wert ist nicht Trade-off, sondern bewusste Wahl: bei zu starkem Decay verlieren die Gaussians während der Refinement-Phase ihre Fähigkeit, sich auf neu hinzugekommene Densification-Kandidaten einzustellen. Über die V431/V433-Erweiterung kann die Schedule-Phase verkürzt werden (`T49 < T1`), sodass `T4` bereits vor Trainingsende erreicht wird und der Rest des Trainings bei konstanter Mini-LR läuft — typische Konfiguration: `T49 = 20 000`, `T1 = 35 000`, Refinement also bei 0.000016 für 15 000 Iterationen.

PROSTYMI SŁOWAMI

Wie langsam die Position-Lernrate am Ende des Trainings wird. Wir haben das bewusst weniger aggressiv eingestellt als das Originalpaper — Splats können bis zum Schluss noch ein bisschen wackeln, das macht sie schärfer. Wenn du daran drehst: höher = unruhigere Splats am Ende, niedriger = Splats können sich nicht mehr anpassen, wenn neue auftauchen.

T5 shDCLearningRate SZCZEGÓŁY

Default: 0.0025 (Initializer + Paper), 0.005 (`.full` und alle MCMC-Presets — 2x) **Range:** 0.0001 – 0.05 **Defined in:**

 TECHNICZNI

Adam-Lernrate für den DC-Anteil (degree 0, also konstantes Albedo) der spherical-harmonic-Farbe. SH-DC entspricht dem richtungsunabhängigen Grundton einer Gaussian, gewissermaßen die „Basisfarbe“. Die V176- und V188-Experimente fanden 2x höher als der Paper-Default optimal — schnellere Farb-Konvergenz, gerade weil bei kurzem Training (, 5 000 Iterationen) die SH-DC sonst nicht in Form kommt. Anders als die geometrischen LRs hat SH-DC keinen Decay; die Lernrate bleibt über alle Iterationen konstant (oder folgt nur dem optionalen extended-phase-Decay aus `T51`). V416 testete eine Vervierfachung auf 0.01 → 6.4 % schlechterer Loss bei beta2=0.99-Adam.

 PROSTYMI SŁOWAMI

Wie schnell sich die Grundfarbe jedes Splats anpasst. Den Wert ändert man fast nie selbst — die Presets haben den richtigen Wert. Höher ginge schneller, kann aber zu instabilen Farben führen.

T6 shRestLearningRate SZCZEGÓŁY

Default: 0.000125 (Initializer + Paper), 0.00025 (`.full` und MCMC — 2×) **Range:** 0.000001 – 0.005 **Defined in:**

 TECHNICZNI

Adam-Lernrate für die SH-Koeffizienten höherer Ordnung (Degree 1, 2, 3 — also die view-direction-abhängigen Farbanteile, die für Glanzlichter, Spiegelungen und sanfte Schattierung sorgen). 20× kleiner als `T5` per Paper-Konvention, weil diese Koeffizienten quadratisch in Anzahl wachsen (3 für Degree 1, 5 für Degree 2, 7 für Degree 3 → insgesamt 15 Floats pro Gaussian) und ohne kleinere Lernrate das Bild übersättigen würden. Wird in zwei Schritten freigeschaltet — bis zur ersten Marke in `T21 shDegreeUpgradeIterations` ist nur Degree 0 aktiv (also nur `T5`), danach 1, später 2, schließlich 3. Niedrige Werte hier sind besonders wichtig auf Szenen mit viel diffuser Beleuchtung; bei sehr glänzenden Oberflächen (Auto-Lack, Wasser) lohnt sich kein Drehen — die SH-Repräsentation an sich ist begrenzt.

 PROSTYMI SŁOWAMI

Wie schnell die richtungsabhängigen Farb-Effekte (Spiegelungen, Glanz) lernen. Standardmäßig sehr klein, weil sonst alles glänzt. Den Wert lässt man besser stehen — wer Glanzlichter besser hinbekommen will, ist eher bei MCMC und längerer Trainingszeit besser bedient als bei dieser LR.

T7 **opacityLearningRate** **SZCZEGÓŁY**

Default: 0.05 (Initializer + Paper), 0.1 (`.full` , MCMC — 2x) **Range:** 0.001 – 1.0 **Defined in:**

 **TECHNICZNI**

Adam-Lernrate für die logit-Opazität jeder Gaussian. Die App speichert Opazität als unbeschränkten Float-Wert und transformiert ihn mit Sigmoid in $[0, 1]$; die LR wirkt im Logit-Space. Der Paper-Default 0.05 ist nach V50-Tests (Best Single-Run L1 0.1664) wiederhergestellt, V71 revertete V67s 0.025. Die V188-Verdopplung auf 0.1 macht das Pruning effizienter — tote Gaussians fallen schneller unter den T14 `pruneOpacityThreshold` ab. V418 zeigte: 0.05 mit `beta2=0.99`-Adam ist 7.1 % schlechter als 0.1 — die Wechselwirkung mit der Adam-Konfiguration ist nicht trivial. Niedrige Werte (< 0.01) führen dazu, dass „dead“ Gaussians ewig herumliegen und Speicher verbrauchen; zu hohe Werte (> 0.5) können zu Opacity-Explosion führen, daher wird der Logit-Wert im Optimizer auf $[-15, 3]$ geclamped (siehe Notiz „Opacity Explosion Prevention“ in CLAUDE.md).

 **PROSTYMI SŁOWAMI**

Wie schnell Splots durchsichtig oder undurchsichtig werden. Wichtig fürs Aufräumen — Splots, die nichts beitragen, müssen schnell verschwinden, damit kein Schleier entsteht. Standardwert passt, nur Profis ändern ihn.

T8 `opacityLearningRateFinal` SZCZEGÓŁY

Default: 0.0 (= „kein Decay„) **Range:** 0 oder 0.001 – **Defined in:**

 TECHNICZNIE

Optionaler Cosine-Decay-Endwert für die Opacity-LR (V427). Wenn 0.0, ist Decay deaktiviert und die Opacity-LR bleibt über das gesamte Training konstant bei `T7`. V427 testete einen Decay 0.1 → 0.01 — Ergebnis 11.5 % schlechterer Loss; reverted, daher der Default „aus“. Die Hypothese hinter dem Feld: in der Refinement-Phase könnte konstante Opacity-LR zu Oszillation führen, sodass Splats, die schon das richtige Maß an Transparenz erreicht haben, durch zufällige Gradient-Schwankungen wieder verschoben werden. Empirisch bestätigt sich das nicht — die Logit-Clamping-Logik fängt das ohnehin ab. Das Feld bleibt verfügbar für zukünftige Experimente; auch sehr lange MCMC-Läufe (> 500K Iterationen) könnten davon profitieren.

 PROSTYMI SŁOWAMI

Ob die Opacity-Lernrate gegen Ende kleiner werden soll. Standard: nein. Wir haben es probiert, war schlechter, lassen es deaktiviert. Bleib bei 0.

T9 `scaleLearningRate` SZCZEGÓŁY

Default: 0.005 (Initializer + Paper), 0.01 (`.full`, MCMC — 2x) **Range:** 0.0001 – 0.1 **Defined in:**

 TECHNICZNIE

Adam-Lernrate für die drei Skalen-Komponenten jeder Gaussian im log-Space (RadiancKit speichert $\log(\text{scale})$, damit Skalen positiv bleiben). Der Paper-Default 0.005, in RadiancKit verdoppelt auf 0.01 für besseren Scale-Konvergenz bei den optimierten Lernraten-Konfigurationen. V423-Experiment: 0.005 mit $\beta_2=0.99$ -Adam → 18.7 % schlechterer Loss und sichtbar zu wenige Gaussians (Density-Control konnte nicht klonen, weil die Skalen-Updates zu lahm waren). Skala kontrolliert die Ausdehnung jeder Gaussian — zu schnelles Lernen führt zu „needle“-Gaussians (extrem lange dünne Splats, siehe T34 `scaleRatioPruneThreshold`), zu langsames Lernen lässt Splats zu kompakt bleiben und Density-Control muss zu oft splitten.

 PROSTYMI SŁOWAMI

Wie schnell die Form der Splats sich anpasst. Standard ist gut. Wenn du das hochdrehst, bekommst du gerne „Nadel“-Splats — extreme lange dünne Tropfen, die das Bild floaten lassen.

T10 rotationLearningRate SZCZEGÓŁY

Default: 0.001 (Initializer + Paper), 0.002 (`.full` , MCMC — 2×) **Range:** 0.0001 – 0.05 **Defined in:**

 TECHNICZNI

Adam-Lernrate für die vier Quaternion-Komponenten jeder Gaussian. Die Quaternion wird in jedem Optimizer-Schritt nach der Adam-Update wieder normalisiert (L2-Norm = 1) — andernfalls würde die Kovarianzmatrix entartet. RadianceKit verdoppelt den Paper-Default in den Quality-Presets, weil Rotation gegenüber Skala / Position kleinere absolute Gradient-Magnituden hat (auf der Einheitskugel bleibt jeder Schritt kurz) und ohne 2× wäre die Rotation im 35 000-Iterations-Fenster deutlich unterkonvergiert. V188 dokumentiert. Auf NeRF-Blender-Szenen (Lego, Chair) wirkt sich Rotation besonders aus — die Kanten der Objekte richten sich erst nach 5 000–10 000 Iterationen richtig aus.

 PROSTYMI SŁOWAMI

Wie schnell sich die Splats drehen lernen — also auf der Oberfläche eines Objekts in die richtige Ausrichtung kommen. Standard passt. Anders ausgedrückt: wenn Splats wie schief liegende Klötze aussehen statt sich an die Oberfläche anzuschmiegen, ist eher die Trainingszeit zu kurz, nicht diese Lernrate zu niedrig.

Densification — Classic (T11–T16)

T11 densifyGradThreshold

SZCZEGÓŁY

Default: 0.000002 (Initializer, kalibriert für 0.5× Auflösung), 0.0000011 (`.full` , kalibriert für 1.0×), 0.000004 (`.quickTest` , kalibriert für 0.25×), $2e-7$ (`.fullClassicPaper`) **Range:** $1e-8$ – $1e-3$ (auflösungsabhängig) **Defined in:**

TECHNICZNIE

Schwellwert für die L2-Norm des bildschirmraum-projizierten Gradienten `dMean2D` , oberhalb dessen eine Gaussian für Klonen oder Splitten markiert wird. Der absolute Wert hängt direkt von der Trainingsauflösung ab — `dMean2D` skaliert ungefähr wie $1/\text{Auflösung}^2$ (mehr Pixel = kleinere Per-Pixel-Gradienten). Daher braucht jede T22 `trainingRenderScale`-Stufe einen kalibrierten Schwellwert: 0.25× → $4e-6$, 0.5× → $2e-6$, 1.0× → $5e-8$... $1.1e-6$ (`.full`). Der Paper-Default 0.0002 ist NDC-normalisiert und in RadiancesKits Welt-Raum-Pipeline nicht direkt vergleichbar. Mit dem in V440 zugeschalteten T52 `adaptiveDensifyThreshold`-Flag lässt sich der Wert in Laufzeit aus dem p98 der aktuellen Gradienten-Verteilung berechnen — aber V440 testete das auf realen Szenen und produzierte 63 K Gaussians (katastrophaler Pruning-Verlust); das Flag bleibt aus. Q5 (T77–T79) liefert eine alternative Adaptive-Logik via rolling median. **Gefahrlos ist dieses Feld nicht** — Halbierung erzeugt 2–4× mehr Gaussians (Speicher-Druck, OOM-Risiko); Verdopplung kann die Szene unter-densifizieren.

PROSTYMI SŁOWAMI

Wie empfindlich die App ist, wenn sie entscheiden soll, ob ein Splat zu wenig dargestellt wird und vervielfältigt werden muss. Niedriger Wert = empfindlicher = mehr Splats. Höher = weniger Splats. Das ist einer der gefährlichsten Werte überhaupt: zu niedrig und der Mac läuft mit Millionen Splats voll Speicher und stürzt eventuell ab. Lass das Feld in Ruhe, oder ändere es nur in Schritten von 10 %.

T12 densifyFromIteration**SZCZEGÓŁY****Default:** 500 **Range:** 100 – 5 000 **Defined in:****TECHNICZNI**

Erste Iteration, ab der Densification aktiv wird. Vorher passiert nur „nacktes„ Lernen auf der initialen SfM-Punktewolke, ohne dass neue Gaussians erzeugt werden. Der Default 500 stammt aus dem 3DGS-Paper und gibt der Initialisierung Zeit, sich zu stabilisieren — wenn schon ab Iteration 0 densifiziert wird, klonen sich falsch positionierte SfM-Punkte vielfach, bevor sie überhaupt ihren richtigen Platz finden. V349 testete 1000 → leicht schlechter Loss; der Default ist optimal.

PROSTYMI SŁOWAMI

Wann die App das erste Mal anfängt, Splats zu klonen. Vorher lernt sie nur die schon vorhandenen Punkte. 500 ist der Standardwert — gibt der App genug Zeit, sich erst mal zu orientieren, bevor sie vervielfacht.

T13 densifyInterval**SZCZEGÓŁY****Default:** 100 (Initializer, MCMC), 200 (`.full`) **Range:** 50 – 1 000 **Defined in:****TECHNICZNI**

Wie viele Iterationen zwischen zwei Densification-Schritten liegen. Im Paper-Default 100 — alle 100 Iterationen wird die Liste der densify-Kandidaten ausgewertet, geklont/gesplittet und gleichzeitig die Liste der prune-Kandidaten ($\text{sigmoid}(\text{opacity}) < T14 \text{ pruneOpacityThreshold}$) entfernt. V112-Tests fanden 200 als optimal für `.full` — das entlastet die GPU, weil weniger Reorganisations-Passes laufen, und gibt jeder Gaussian mehr Zeit, sich nach einer Klone-Aktion einzupendeln. V417 testete 100 mit $\text{beta2}=0.99 \rightarrow 5.8\%$ schlechter (957 K Gaussians, Überdensifizierung). Bei MCMC wird dasselbe Feld als Relocation-Interval interpretiert; siehe T67 `mcmcRelocationInterval` für die MCMC-spezifische Logik.

PROSTYMI SŁOWAMI

Wie oft die App nach neuen Splats schaut. 100 = oft, 200 = mittel. Höher heißt: jeder Splat hat länger Zeit sich einzurichten, bevor wieder vervielfältigt wird. Das ist gut. Auf 50 zu senken kann den GPU dauerhaft beschäftigen, ohne dass es nennenswert besser wird.

T14 `pruneOpacityThreshold` SZCZEGÓŁY

Default: 0.005 (Initializer, Paper, MCMC), 0.001 (`.full`) **Range:** 0.0001 – 0.1 **Defined in:**

 TECHNICZNI

Sigmoid-Opazitäts-Schwelle, unter der eine Gaussian beim nächsten Densification-Step gelöscht wird. Wirkt zusammen mit `T7 opacityLearningRate` und der Logit-Clamp-Logik im Optimizer. V393 senkte den Default von 0.005 auf 0.001 in `.full` — Folge: Splats die nur unter exotischen Blickwinkeln eine Rolle spielen, bleiben länger erhalten und tragen zur SH-Detail bei. V394 testete 0.0001 → leicht schlechter (zu wenig gepruntet, Speicher verschwendet). Wichtig: Density-Control muss IMMER prunen, auch wenn die Buffer-Kapazität durch andere Maßnahmen schon voll ist (siehe „Density Control Must Always Prune“, in CLAUDE.md) — sonst akkumulieren tote Gaussians und der Count friert ein.

 PROSTYMI SŁOWAMI

Wann ein Splat als „durchsichtig genug“, gilt, um gelöscht zu werden. 0.005 ist der Paper-Standard, wir haben in Quality 0.001 — also wir geben Splats länger eine Chance. Das macht weiches Licht und schwache Schatten besser darstellbar. Höher zu setzen (über 0.01) lässt die Splat-Anzahl schnell sinken — kann sinnvoll sein bei Speicherknappheit, kostet aber Detail.

T15 `opacityResetInterval` SZCZEGÓŁY

Default: 3 000 (Initializer + Paper), 100 000 (`.full` = effektiv deaktiviert), 200 000 (`.fullMCMC` = deaktiviert) **Range:** 1 000 – 100 000+ **Defined in:**

 TECHNICZNIE

Alle wie viele Iterationen wird die Opazität aller Gaussians auf einen niedrigen Wert (~0.01) zurückgesetzt — eine Maßnahme aus dem 3DGS-Paper, um „eingefrorene„ Splats neu zu beurteilen. V194 zeigte, dass mit RadianceKits Warmup + Stochastic-Trainings-Setup + 2x Lernraten der Opacity-Reset 5.5 % Qualität kostet und der Logit-Clamp die Reset-Funktion bereits abdeckt. Daher in `.full` praktisch deaktiviert (100 000 > 35 000 = nie ausgelöst). V421 testete Reset alle 3 000 mit $\beta_2=0.99 \rightarrow 4.9\%$ schlechter; reverted. Bei `.fullClassicPaper` (Q1.5-A, Paper-treuer Test) ist es bewusst wieder auf 3 000 gesetzt — das war eines der Lever, mit denen die Paper-Magnitude-Gaussian-Budgets erreicht werden sollten.

 PROSTYMI SŁOWAMI

Alle wie viele Iterationen die App die Sichtbarkeit aller Splats auf „fast unsichtbar„ zurücksetzt — eine Art Reset-Knopf für die Opacity. Bei uns deaktiviert (Wert so hoch, dass es nie passiert), weil andere Mechanismen das überflüssig machen. Nur bei Paper-getreuen Experimenten anschalten.

T16 maxScreenSize SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 (off) oder > 0
Defined in:

 TECHNICZNI

Maximale Bildschirmraum-Größe (in projizierten Pixeln), die eine Gaussian erreichen darf, bevor sie zwangsweise gesplittet wird. Der Wert ist auf 0 gesetzt (V48 testete und reverted) — RadianceKits Density-Control verwendet stattdessen den Welt-Raum-Skala-Schwellwert aus der `dMean2D`-Logik. Bleibt im Feldkatalog enthalten, weil künftige Experimente mit Mip-Splatting (T74–T76) oder szenenspezifischen Splatting-Strategien davon profitieren könnten. Aktivierung (Wert > 0, z.B. 20) würde sehr groß gewordene Splats im Bildschirm zwingen, sich aufzuteilen — relevant bei großen, glatten Wandflächen, wo ein einzelnes Riesensplat zu wenig Detail bietet.

 PROSTYMI SŁOWAMI

Begrenzung, wie groß ein einzelner Splat auf dem Bildschirm werden darf. Bei uns aus. Eingeschaltet würde es bewirken, dass riesige flache Splats (z.B. auf einer Wand) zwangsweise in mehrere kleine zerlegt werden. Lass es aus, falls nicht ausdrücklich Experimente damit.

Loss (T17–T20)

T17 ssimWeight

SZCZEGÓŁY

Default: 0.2 (Initializer + Paper + `.full`), 0.05 (alle MCMC-Presets) **Range:** 0.0 – 1.0 **Defined in:**

TECHNICZNI

Gewicht des D-SSIM-Anteils in der kombinierten Loss-Funktion $loss = (1 - \lambda) * L1 + \lambda * D-SSIM$, wobei $\lambda = T17$. Der 3DGS-Paper-Default 0.2 ist für Classic-Densification optimal — V383 testete 0.3 → 28.9 % schlechter, V373b bestätigte 0.2 als sweet spot. Für MCMC wurde in V521b/V534 unabhängig festgestellt: 0.05 ist optimal, weil MCMC durch seine stochastische Exploration einen stärkeren L1-Signal-Anteil braucht — höhere SSIM-Gewichte würden die Relocation-Entscheidungen verwässern. SSIM ist deutlich teurer zu rechnen als L1 (lokale 11×11-Fenster über das ganze Bild); RadianceKit nutzt eine MPS-beschleunigte Implementierung, die unter 1 ms pro 1080p-Bild bleibt. Q7-BayesOpt-Sweeps fanden szenenspezifische Optima zwischen 0.05 (`.outdoorPreset : 0.082`) und 0.171 (`.indoorPreset`).

PROSTYMI SŁOWAMI

Wie wichtig die App neben „jeder Pixel stimmt“, auch „Strukturen sind ähnlich“ findet. 0.2 ist der Standard und liefert ein gutes Bild. Niedriger = pixel-genauer, aber kann weichere Übergänge bekommen. Höher = strukturähnlicher, aber Details werden weicher. Lass die Presets entscheiden.

T18 **ssimWeightRefinement** **SZCZEGÓŁY**

Default: 0.0 (= „kein Wechsel, behalte ssimWeight„)

Range: 0 oder 0 – 1.0 **Defined in:**

 **TECHNICZNIE**

Optionaler SSIM-Wert für die Refinement-Phase nach T2 `densifyUntilIteration`. V428 testete 0.2 → 0.3 im Refinement → 16 % schlechterer Loss (sowohl L1 als auch SSIM verschlechterten sich); revertet, daher Default 0.0. Die Hypothese hinter dem Feld war, dass nach der Densification — wenn keine neuen Gaussians mehr entstehen — ein stärkerer SSIM-Anteil die strukturelle Schärfe maximieren würde. Empirisch falsch: SSIM-Gewicht zu erhöhen heißt indirekt L1-Gewicht zu senken, und L1 ist das deutlich aussagekräftigere Signal in der Final-Refinement-Phase. Das Feld bleibt verfügbar für künftige Experimente mit perceptueller Loss (T60) oder Edge-Loss (T19), wo eine Refinement-spezifische Loss-Komposition sinnvoll sein könnte.

 **PROSTYMI SŁOWAMI**

Spezial-Einstellung für die zweite Trainingsphase (Verfeinerung nach Splat-Vervielfältigung). Bei 0.0: dieselbe SSIM-Gewichtung wie davor. Drehen bringt empirisch nichts, daher aus.

T19 edgeLossWeight SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.001 – 1.0 **Defined in:**

 TECHNICZNIE

V437-Experimental-Loss: Gewicht eines Sobel-Gradient-Domain-L1-Loss, der die Bildkanten direkt vergleicht (Ground-Truth-Sobel vs Render-Sobel) zusätzlich zu L1+SSIM. Hypothese: Kanten-Information ist ein perceptueller Eckpfeiler von Bildqualität und ein expliziter Term sollte Gaussians ermutigen, Kanten besser zu treffen. Test-Ergebnisse: Gewicht 0.1 → 11 % schlechterer Loss, 0.01 → quality-neutral aber 10 % langsamer. Der Sobel-Pass kostet einen zusätzlichen MPS-Forward auf Ground-Truth und Render. Daher dauerhaft deaktiviert. Künftiger Use-Case: Szenen mit harten künstlichen Kanten (Architektur, Möbel, Renderings) könnten profitieren — Q7-Scene-Class-Presets haben das aber nicht gepickt, sondern stattdessen das SSIM-Gewicht skaliert.

 PROSTYMI SŁOWAMI

Experimenteller Zusatz, der Kanten extra wichtig nimmt. Bringt empirisch nichts. Bleibt aus.

T20 skyMaskingEnabled**SZCZEGÓŁY**

Default: false (Initializer und alle Presets) **Range:** boolean **Defined in:**

TECHNICZNE

Schaltet Sky Masking ein. Dabei wird in jedem Bild via Apple Vision Framework (VNGenerateForegroundInstanceMaskRequest) die Sky-Region ausmaskiert, und der Loss in diesem Bereich auf null gesetzt. Sinn: Outdoor-Szenen leiden oft daran, dass blaue/graue/weiße Sky-Pixel die App dazu bringen, Gaussians genau dort zu plazieren — was als „floater“, wahrgenommen wird. Ohne Sky-Mask wäre der Loss in diesem Bereich nie null, weil der Himmel im Bild leicht variiert und die App ewig versucht, das mit Splats nachzubauen. Die Vision-Maske wird einmal pro Kamera vor dem Training berechnet und im RAM gehalten. Wird typischerweise zusammen mit `T45 skyDomeEnabled` aktiviert (UI-Logik in der Settings-View). Bei Innen-Szenen oder synthetischen Renderings deaktiviert lassen — die Maske würde dort fälschlicherweise Decken oder Wände als „Sky“ erkennen.

PROSTYMI SŁOWAMI

Schaltet einen Spezialmodus für Außen-Aufnahmen an: der Himmel wird beim Training ignoriert, damit nicht versucht wird, ihn mit Splats nachzubauen. Empfohlen für jede Außen-Szene. Bei Innen oder bei 3D-Renderings aus Blender aus lassen.

SH-Degree-Progression (T21)

T21 shDegreeUpgraderIterations

SZCZEGÓŁY

Default: [1_000, 2_000, 3_000] (Initializer), [2_000, 5_000, 8_000] (.full , MCMC), [1_000, 2_000] (.preview — Degree 3 übersprungen) **Range:** [Int] , jeder Wert in [0, maxIterations] , monoton steigend **Defined in:**

TECHNICZNI

Iterationen, an denen der aktive SH-Degree von 0→1, 1→2, 2→3 hochgeschaltet wird. Vor der ersten Marke sind nur die DC-Komponenten aktiv (also T5 shDCLearningRate), nach der ersten Marke die DC + 3 Degree-1-Koeffizienten, nach der zweiten Marke + 5 Degree-2-Koeffizienten, nach der dritten Marke alle 15 Koeffizienten. Der Speicherbedarf pro Gaussian wächst dabei in Stufen — 4 Floats → 16 Floats → 36 Floats → 64 Floats. Die Quality-Presets verzögern die Aufstufungen gegenüber Initializer Defaults (V228), weil die Geometrie zuerst stabilisieren soll, bevor die Farbdetails mit ihrer höheren Frequenz draufkommen. V384 testete [1K, 2K, 3K] für .full → 9.3 % schlechter — bestätigt das Delay. .preview kappt bei Degree 2, weil Degree 3 in 5 000 Iterationen nicht konvergiert und nur Optimizer-Kapazität verbraucht. Q6 (T80–T81) bietet eine alternative Curriculum-Logik, die diese Liste dynamisch überschreibt.

PROSTYMI SŁOWAMI

An welchen Punkten im Training die App lernt, dass Farben aus verschiedenen Blickwinkeln unterschiedlich aussehen können (Glanzlichter, Spiegelungen). Erst spät — damit erst die Form stimmt, dann die Farbe. Die Werte in den Presets sind so eingestellt, dass das gut funktioniert. Nichts dran ändern, außer du weißt genau warum.

Performance (T22–T25)

T22 trainingRenderScale

SZCZEGÓŁY

Default: 1.0 (Initializer, `.full`, MCMC, Scene-Class), 0.5 (`.preview`), 0.25 (`.quickTest`) **Range:** 0.05 – 2.0 (typisch 0.25, 0.5, 1.0) **Defined in:**

TECHNICZNI

Render-Auflösung beim Training relativ zur Originalauflösung der Trainingsbilder. Bei 0.5 wird jedes Bild auf 50 % Breite × 50 % Höhe heruntergerechnet (also 25 % der Pixel) und das Gaussian-Rendern erfolgt in dieser kleineren Auflösung. Reduziert sowohl Speicher- als auch Rechenaufwand quadratisch. Wichtig: T11 `densifyGradThreshold` muss zur gewählten Auflösung passen — die Gradient-Magnituden skalieren mit $1/\text{Auflösung}^2$, daher hat `.quickTest` (0.25x) einen viel höheren Threshold ($4e-6$) als `.full` (1.0x, $1.1e-6$). RadianceKit warnt bei sehr großen Bildern und passt automatisch an — 3-MP-Ziel-Auflösung. Bei extremen 4K-Eingangsbildern wäre 0.5 oder sogar 0.25 sinnvoll, sonst läuft jeder Mac auch nur in CPU-Compaction.

PROSTYMI SŁOWAMI

Wie groß die Bilder beim Training sind. 1.0 = original, 0.5 = halb so groß. Halbe Größe = viermal schneller, aber feinste Details fehlen. Die Presets wählen den richtigen Wert; bei extrem großen Eingangsbildern (über 12 Megapixel) schaltet die App automatisch runter.

T23 resolutionWarmupScale

SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.1 – **Defined in:**

TECHNICZNI

V133-Optimierung: Trainiere die Densification-Phase (Iter 0 bis T2) in einer niedrigeren Auflösung als die Refinement-Phase. V308 hat sie für `.full` wieder ausgeschaltet, weil bei T22 = 1.0 und Cosine-Annealing der Time-Win marginal war und Qualität minimal litt. Bleibt im Feldkatalog, weil sie bei 4K-Eingaben und langen Trainingsläufen wieder sinnvoll werden könnte — Q6 Curriculum (T80) hat eine ähnliche Logik aufgegriffen, dort ist sie aber an die LR-Schedule gekoppelt. Wenn aktiviert und T80 `curriculumResolutionRamp` ebenfalls true, gewinnt Q6 und überschreibt diesen Wert.

PROSTYMI SŁOWAMI

Spezial-Feature: in der ersten Trainingshälfte mit kleineren Bildern lernen, in der zweiten mit großen. Spart Zeit. Aus, weil die neuere Q6-Variante das besser löst.

T24 tileSize SZCZEGÓŁY**Default:** 16 **Range:** 8, 16, 32 **Defined in:** TECHNICZNIE

Größe der Rasterisierungs-Tiles in Pixeln. Das Gaussian-Splatting-Rendering ist tile-basiert: das Bild wird in 16x16-Pixel-Kacheln zerlegt, jede Kachel sammelt die für sie relevanten Gaussians, sortiert sie nach Tiefe und blendet sie ein. 16 ist der von praktisch allen 3DGS-Implementierungen verwendete Standard und in den RadianceKit-Metal-Kernels hartkodiert; eine Änderung dieses Werts würde Re-Compilation der Shader bedingen und ist im aktuellen Stand nicht effektiv. Bleibt als Feld, falls eine künftige Engine-Version Tile-Size dynamisch unterstützt.

 PROSTYMI SŁOWAMI

Interner Render-Parameter. Standard 16, nicht ändern.

T25 throttleDelayMs SZCZEGÓŁY**Default:** 0 (Initializer, `.full`, MCMC, Scene-Class), 0 (`.preview`) **Range:** 0 – 100 **Defined in:** TECHNICZNIE

Künstliche Verzögerung zwischen Trainings-Iterationen in Millisekunden. 0 = volle Geschwindigkeit (Standard). Höhere Werte machen den Mac während des Trainings „benutzbarer“, indem GPU/CPU regelmäßig Atempausen bekommen — die Bedienfreundlichkeit anderer Apps steigt, die Trainingszeit aber linear mit dem Delay. Typische Werte: 1–2 ms („leichtes“ Throttling, +5 % Trainingszeit, Mac fühlt sich responsiver an), 5 ms („Mittelschwer“, +15 % Trainingszeit), 10+ ms („Eco“, potentiell doppelte Trainingszeit). Wird im Inspector unter „Performance“ geboten, ist aber nicht in der Standardansicht — siehe Backlog `dev_ux-backlog.md`, der vorschlägt, ihn aus dem Expert View zu entfernen, weil falsch verstanden er die Trainingszeit dramatisch verlängert.

 PROSTYMI SŁOWAMI

Wie viele Millisekunden Pause die App zwischen Trainingsschritten macht. 0 = keine Pause, schnellstmöglich. Höhere Werte machen den Mac während des Trainings besser benutzbar — aber das Training dauert dann auch länger. Auf einem M3 Ultra oder Mac Studio kannst du das auf 0 lassen; auf einem MacBook Air wäre 2 oder 5 ein guter Wert.

Diagnose und Punktwolken-Vorbereitung (T26–T30)

T26 depthDistortionWeight

SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.0001 – 0.05 **Defined in:**

TECHNICZNIE

V366-Experimental: Gewicht eines Depth-Distortion-Regularisierungs-Loss. Bestraft Gaussians, die entlang eines Render-Strahls zwar tief gestaffelt sind, aber konzeptuell zur selben Oberfläche gehören — das encouraged konzentrierte Tiefenverteilungen und reduziert Floaters. Tests: 0.01 → 4.5 % schlechter, 0.001 → 8.1 % schlechter. Der theoretische Vorteil — Multi-View-Konsistenz verbessern — schlägt sich nicht im L1-Loss nieder, weil die Hypothese implizit annimmt, dass die SfM-Geometrie korrekt ist und die Gaussians nur „gestapelt“ werden müssen. In der Praxis ist die SfM-Punktwolke meist die schwächste Komponente, nicht die Stapelung. Bleibt verfügbar für Multi-View-Datensätze mit besonders sauberen Posen (Synthetic, Mip-NeRF 360 mit Ground Truth).

PROSTYMI SŁOWAMI

Experimentelles Feature zur Vermeidung mehrerer Splits hintereinander an derselben Stelle. Nicht aktiviert, weil die Tests nichts gebracht haben.

T27 singleViewOverfit

SZCZEGÓŁY

Default: false **Range:** boolean **Defined in:**

TECHNICZNIE

Diagnose-Flag: wenn true, wird in jeder Trainingsiteration zwingend Kamera-Index 0 verwendet statt einer zufällig aus dem Camera-Pool. Sinn: Wenn das Modell nicht mal eine einzige View überfitten kann (sprich, der Loss auf View 0 auch nach 10 000 Iterationen nicht gegen Null geht), ist im Forward/Backward-Pass ein fundamentaler Bug. Dieser Schalter wurde während der Entwicklung der Metal-Shader und der Differentiable-Rasterizer-Kernels intensiv genutzt — V42–V47 Phase. Heute nur noch als Sanity-Check verfügbar, wenn jemand Backend-Code modifiziert hat und ein Regression Test machen will. Per CLI mit `--single-view`.

PROSTYMI SŁOWAMI

Test-Modus für Entwickler. Sie können damit prüfen, ob die App überhaupt aus EINEM Bild lernen kann. Für normale Benutzer irrelevant, immer aus lassen.

T28 maxCameras**SZCZEGÓŁY**

Default: 0 (= „alle Kameras verwenden,“) **Range:** 0 oder 1 – N **Defined in:**

TECHNICZNI

Diagnose-Limit aus V43: trainiere nur mit den ersten N Kameras, ignoriere alle weiteren. Sinn ursprünglich: Hypothese testen, dass zu viele Kameras Gradient-Konflikte erzeugen (zu viele widersprüchliche Loss-Signale für dieselbe Gaussian). Test-Ergebnis: kein systematischer Vorteil bei künstlicher Begrenzung — Mehr-Frames bringen praktisch immer mehr Qualität. Bleibt als CLI-Flag (`--max-cameras N`) für gezielte Experimente, z.B. „funktioniert das Training auf den ersten 100 Bildern eines 1 500-Bild-Drohnenflugs?“, Im UI nicht exponiert.

PROSTYMI SŁOWAMI

Diagnose-Feld für Entwickler — nur die ersten N Bilder verwenden, Rest ignorieren. Normaler Benutzer braucht nicht, Wert auf 0 = alle Bilder. Mehr Bilder = besseres Ergebnis (siehe `feedback_more-frames-better.md`).

T29 maxInitialPoints**SZCZEGÓŁY**

Default: 0 (= „alle SfM-Punkte verwenden,“) **Range:** 0 oder 1 000 – 200 000+ **Defined in:**

TECHNICZNI

V54-Sicherung: limitiert die Anzahl der initialen SfM-Punkte, mit denen das Training startet. Dichte COLMAP-Rekonstruktionen können > 60 000 Punkte produzieren, was bei großen Initial-Skalen zu 200–300 Gaussians pro Pixel-Überlapp führt — das macht ein „Nebelfeld“, aus, in dem das Training nicht konvergiert. Subsampling auf ~16 000 Punkte (Hard-cap-Logik in der Trainings-Engine) bringt die Initial-Dichte auf das Niveau, das das Referenz-3DGS verwendet, und reduziert Overlap dramatisch. Wird bei sehr dichten SfMs automatisch gesetzt; per CLI mit `--max-points N`.

PROSTYMI SŁOWAMI

Wie viele Anfangs-Punkte aus der Kamera-Rekonstruktion verwendet werden. Bei sehr dichten Rekonstruktionen (mehr als 60 000) limitiert die App automatisch auf 16 000 — sonst gibt es zu viel Nebel am Anfang. Du brauchst das nicht zu setzen; die App regelt es.

T30 cameraClusterOutlierMultiplier**SZCZEGÓŁY**

Default: 10.0 (alle Presets — niemals überschrieben) **Range:** 1.0 – 100.0 **Defined in:**

TECHNICZNE

Multiplikator für den Camera-Cluster-Outlier-Filter, eingeführt in Phase 3.10 A.1. Vor dem Training berechnet die Trainings-Engine das Centroid aller Kamera-Positionen und die maximale Distanz einer Kamera vom Centroid. SfM-Punkte, deren Distanz vom Centroid $\text{multiplier} \times \text{maxCameraDistance}$ überschreitet, werden als Outlier verworfen. Default 10x bewahrt das Verhalten vor Phase 3.10. Ein subtiler Bug: Tighter SfM (Kameras enger zusammen) → kleinerer → kleinerer Schwellwert → mehr Punkte werden als Outlier verworfen. Looser SfM → größerer Schwellwert → weniger Punkte verworfen. Dies ist eine der Ursachen für die Phase-3.9-Funnel-vs-Training-Anti-Korrelation: bessere SfM kann downstream zu schlechterem Training führen, weil zu viele Initialpunkte gekillt werden. Das Feld liegt als CLI-Override (`--camera-cluster-outlier-multiplier`) für die A.3-Sweeps; im UI nicht exponiert. Werte unter 5 sind in der Regel zu restriktiv, über 20 wirkungslos.

PROSTYMI SŁOWAMI

Spezial-Filter, der Punkte aus der Rekonstruktion verwirft, die weit weg von der Kamera-Wolke liegen. 10 = die App ist großzügig, behält fast alles. Erhöhen kann sinnvoll sein, wenn weit entfernte Punkte (Berge in der Ferne) im Bild aussehen wie schwebende Klümpchen. Niedriger setzen nur in Notfall — verlierst dabei Detail in der Ferne.

Regularisierung (T31–T37)

T31 coarseToFineBlurRadius

SZCZEGÓŁY

Default: 0 (= deaktiviert) **Range:** 0 oder 1 – 10 **Defined in:**

TECHNICZNI

V369-Experimental: Box-Blur-Radius, der zu Beginn der Densification-Phase auf das Ground-Truth-Bild angewendet und linear bis zum Ende der Densification (T2) auf 0 reduziert wird. Hypothese: Coarse-to-Fine-Training — erst grobe Strukturen lernen, dann Details — sollte stabilere Geometrie liefern. Tests: $r=3 \rightarrow 9.6\%$ schlechter, $r=1 \rightarrow 5.1\%$ schlechter. Der Grund für das Fehlschlagen: die Densification entscheidet basierend auf bilddomänen-Gradienten, und Bluren reduziert genau die Signale, die für „hier muss klonen werden“, wichtig sind. Bleibt im Feld-Katalog für künftige Tests mit anderem Density-Control-Schema.

PROSTYMI SŁOWAMI

Experimenteller „Erst-grob-dann-detailliert“-Modus. Hat nichts gebracht, bleibt aus.

T32 scaleRegWeight

SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.0001 – 0.05 **Defined in:**

TECHNICZNI

V370-Experimental: L1-Regularisierung auf welt-räumliche Skala. Bestraft Gaussians, die zu groß werden — verhindert „Mega-Splatts“, die ganze Wandflächen mit einer Gaussian abdecken. Tests: $0.01 \rightarrow 200\%$ schlechterer Loss (2 M Gaussians, totale Explosion), $0.001 \rightarrow 214\%$ schlechter. Der Grund: Skala-Regularisierung kommt mit Density-Control in Konflikt — kleinere Skalen heißen, mehr Gaussians werden gebraucht, also splittet Density-Control häufiger, was wiederum mehr Gradient-Aufwand bedeutet. Disabled, aber dokumentiert für Mip-Splatting-Experimente (T74): in diesem Kontext könnte eine Skalen-Untergrenze sinnvoll sein.

PROSTYMI SŁOWAMI

Regularisierung, die Splats zwingt, klein zu bleiben. Hat in Tests Splat-Explosionen ausgelöst (Millionen Splats). Nicht aktivieren.

T33 anisotropyRegWeight SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.0001 – 0.05 **Defined in:**

 TECHNICZNIE

V445-Experimental: Penalty auf das $\max(\text{scale})/\min(\text{scale})$ -Verhältnis, soll extrem langgezogene „Needle“-Gaussians verhindern, die als floater wahrgenommen werden. Tests: 0.01 → 69 % schlechter, 0.001 → 15 % schlechter. Der Grund: Regularisierung zwingt Splats in Richtung „runde“ Form, was auf einer flachen Oberfläche (Wand, Tisch, Boden) genau falsch ist — dort ist eine flache, breite Gaussian effizienter als eine kugelförmige. Disabled. V549f bot mit T34 `scaleRatioPruneThreshold` einen alternativen, gezielteren Ansatz, der ebenfalls revertet wurde.

 PROSTYMI SŁOWAMI

Regularisierung, die zu lange dünne Splats bestraft. Klingt sinnvoll, war in Tests aber schlechter. Aus.

T34 scaleRatioPruneThreshold SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 5.0 – 100.0 (typisch 10.0 – 30.0) **Defined in:**

 TECHNICZNIE

Experimentelles post-Training-Pruning, das jede Gaussian löscht, deren $\max(\text{scale})/\min(\text{scale})$ -Verhältnis den hier gesetzten linearen Schwellwert überschreitet. Zielt auf extrem langgezogene „Needle/Disc“-Floaters ab, die durch Regularisierung allein nicht eliminiert werden können. Im Test entfernte das Pruning Floaters wie erhofft, aber gleichzeitig auch sinnvolle flache Splats auf Wänden und Böden — das Bild wurde löchriger. Daher per Default aus, das CLI-Flag (`--scale-ratio-prune N`) bleibt für gezielte Experimente verfügbar. Empfohlene Werte falls man trotzdem testen will: 30 (sehr konservativ, entfernt nur extreme Outlier), 10 (aggressiv, kostet Detail).

 PROSTYMI SŁOWAMI

Versuch, ganz langgezogene Splats nach dem Training rauszufiltern. War in Tests netto-negativ — Floater weg, aber auch Detail weg. Aus.

T35 **opacityRegWeight** SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.0001 – 0.05 **Defined in:**

 TECHNICZNIE

V446-Experimental: Binary-Cross-Entropy-Penalty, der Opazität gegen 0 oder 1 zieht (also weg von „halb-transparent“). Hypothese: scharfere Opazitätsverteilung würde Bildklarheit verbessern. Test mit T33 kombiniert → Regularisierung kostet Qualität, beide deaktiviert. Disabled. Achtung: in 1.4.3-Beta tauchte ein Bug auf, der genau dieses Feld in einer Default-Wert-Veränderung (Initializer = 0.01) hatte, was zu Mass-Extinction des Gaussian-Counts (460 K → 5 in einer Iteration) führte. Seit 1.4.4 fest auf 0.0 als Default verankert.

 PROSTYMI SŁOWAMI

Regularisierung, die Splits entweder ganz transparent oder ganz solid macht. Bringt nichts, kann sogar gefährlich werden (1.4.3-Bug Mass-Extinction). Lass auf 0.

T36 **opacityDecayFactor** SZCZEGÓŁY

Default: 0.0 (Initializer = deaktiviert), 0.9995 (`.full`, `.classicBalanced` — HTGS-Standard) **Range:** 0 (off) oder 0.95 – 1.0 **Defined in:**

 TECHNICZNIE

V546-Implementation des HTGS-Schemas (Hierarchical Time-Gating, Eurographics 2025): alle T37 `opacityDecayInterval` Iterationen wird die sigmoid-Opazität jeder Gaussian mit diesem Faktor multipliziert. 0.9995×100 Anwendungen ergibt ~95 %-Verbleib pro Densification-Phase — ein leichter aber stetiger Abwärts-Druck auf alle Opacities, der schwach beigetragene Gaussians verlässlich gegen den T14 `pruneOpacityThreshold` sinken lässt. Das Resultat: 14 % besserer L1-Loss auf Horse Full (3-Trial-Avg V546) gegenüber V438 ohne Decay. Nur während Densification-Phase aktiv (bis T2), danach läuft das Training ohne Decay weiter, damit die im Refinement etablierten Opacities stabil bleiben. Bei MCMC nicht verwendet (MCMC hat eigene Mechanismen via T67 `mcmcRelocationInterval` + T68 `mcmcDeadOpacityThreshold`).

 PROSTYMI SŁOWAMI

„Sanftes Verblässen“ aller Splits über die Trainingszeit. Macht inaktiv gewordene Splits schneller transparent, sodass sie beim Aufräumen weggehen. War der wichtigste Quality-Hebel des V546-Updates: 14 % besser. In Quality-Preset eingebaut. Selbst zum Drehen nicht empfohlen, weil exakt austariert.

T37 opacityDecayInterval SZCZEGÓŁY**Default:** 50 **Range:** 10 – 500 **Defined in:** TECHNICZNIE

Iterations-Intervall, in dem T36 opacityDecayFactor angewendet wird. HTGS-Paper-Default 50, in `.full` belassen. Lange Intervalle (>200) heben den Effekt teilweise auf, weil zwischen zwei Anwendungen genug Gradient-Updates passieren, dass Opacity wieder steigt. Kürzere Intervalle (<20) machen Decay zu aggressiv. Nur in Densification-Phase aktiv.

 PROSTYMI SŁOWAMI

Wie oft das „Verblassen“ angewendet wird. 50 = alle 50 Iterationen ein bisschen verblasse-Schritt. Passt.

Refinement (T38–T44)**T38** gradientAccumulationSteps SZCZEGÓŁY**Default:** 1 (= „eine View pro Adam-Schritt,“) **Range:** 1 – 8 **Defined in:** TECHNICZNIE

V424-Feature: Anzahl der Views, deren Gradienten akkumuliert werden, bevor ein Adam-Update ausgeführt wird. Bei `> 1` läuft die App auf einem separaten, „unfused“, Backward-Project-Pfad, der die Gradienten in einem separaten Buffer summiert; die finale Anwendung skaliert mit $1/N$, um die Magnitude konstant zu halten. V424 testete 2-View → quality-neutral, aber 10 % langsamer (weil unfused-Pfad teurer ist als fused-Pfad). Reverted für `.full`, aber für MCMC bewusst verwendet — `.fullMCMC` läuft mit, aber V544a-Tests zeigten, dass mit der Quality-Gap zu Classic auf 5 % schrumpft (statt 11 %). Im Initializer-Default 1, im aktuellen Preset 1, bleibt CLI-Flag (`--accum-steps N`).

 PROSTYMI SŁOWAMI

Wie viele Bilder die App betrachtet, bevor sie die Splits anpasst. 1 = jedes Bild einzeln. Höher = mehrere Bilder gleichzeitig anschauen und dann einen Mittelwert anwenden. Bringt im Standardfall nichts; bei MCMC kann 2 ein bisschen helfen.

T39 testViewIndices SZCZEGÓŁY

Default: `[]` (= leer, alle Views werden zum Training verwendet) **Range:** `Set<Int>`, beliebige Untermenge der Camera-Indices **Defined in:**

 TECHNICZNIE

V546-Feature: Set von Camera-Indices, die NICHT zum Training verwendet, sondern als Holdout für PSNR/SSIM/LPIPS-Auswertung gespart werden. Wird automatisch gesetzt, wenn der `--benchmark`-CLI-Flag aktiv ist: dann jede achte View, beginnend bei Index 0 (LLFF-Standard, identisch mit den Mip-NeRF-360- und 3DGS-Paper-Konventionen). Ohne benchmark leer — das Training nutzt alle Views. **Vorsicht:** das manuelle Setzen dieses Feldes ohne Verständnis der Indizes kann den Benchmark unbrauchbar machen (z.B. wenn alle Indices über N gesetzt werden, während es nur N-50 Views gibt → keine Holdouts → keine Auswertung). Beim eigenen Preset-Export wird testViewIndices nicht persistiert, weil es szenenabhängig ist und sonst zwischen verschiedenen Datensätzen sinnlose Werte hinterlassen würde.

 PROSTYMI SŁOWAMI

Welche Bilder beim Training „ausgespart“, werden, um sie später für Qualitätsmessung zu nutzen. Du setzt das nicht selbst; das `--benchmark`-Flag macht das automatisch (jedes achte Bild ist Test). Wenn du eigene Indices setzt: gefährlich, kann den Benchmark verfälschen.

T40 refinementPruneInterval SZCZEGÓŁY

Default: 0 (= deaktiviert) **Range:** 0 oder 100 – 5 000 **Defined in:**

 TECHNICZNIE

V425-Feature: alle N Iterationen während der Refinement-Phase (nach T2) wird ein zusätzlicher Prune-Pass laufen gelassen, der Gaussians mit $\text{sigmoid}(\text{opacity}) < T_{41}$ `refinementPruneOpacityThreshold` entfernt. Sinn: während Densification gibt es regelmäßige Density-Control-Calls, danach nicht mehr — Gaussians, deren Opacity weiter sinkt, bleiben aber im Buffer. V425 testete und reverted: das zusätzliche Pruning korrelierte mit V426 (Two-Phase Densification, ebenfalls in 0 Gaussians cascade failure abgebrochen). Disabled. CLI-Flag verfügbar für Experimente; falls aktiviert, sind 1 000 oder 2 000 sinnvolle Werte.

 PROSTYMI SŁOWAMI

Zusätzliches Aufräumen während der Verfeinerungsphase. Bringt nichts, bleibt aus.

T41 refinementPruneOpacityThreshold SZCZEGÓŁY

Default: 0.0 (= „verwende T14 „) **Range:** 0 oder 0.001 – 0.1 **Defined in:**

 TECHNICZNIE

V425b: separater Opacity-Schwellwert für Refinement-Pruning. Nach Densification haben die meisten Gaussians eine deutlich höhere Opacity erreicht (> 0.001), sodass der Standard-T14 `pruneOpacityThreshold` zu lasch wäre. Wenn T40 aktiv, bestimmt dieses Feld den eigenen Schwellwert. Bei 0.0 wird T14 weiterverwendet. Nur relevant wenn $T_{40} > 0$.

 PROSTYMI SŁOWAMI

Schwellwert für das zusätzliche Refinement-Aufräumen (siehe T40). Beide Felder nicht aktiv, also irrelevant.

T42 midTrainingCompactificationIterations SZCZEGÓŁY

Default: `[]` (= deaktiviert) **Range:** `[Int]`, Werte in `(densifyUntilIteration, maxIterations)` **Defined in:**

 TECHNICZNI

V549-Feature: explizite Iteration-Punkte während der Refinement-Phase, an denen ein Compactification-Pass läuft (entfernt $\text{sigmoid}(\text{opacity}) < 0.01$ + Outlier-Scale-Gaussians, dieselbe Logik wie T56 `postTrainingCompactification`). Sinn: lange Refinement-Phasen können Confetti-/Floater-Akkumulation zeigen, deren SH dann auf view-spezifische Artefakte überfitten. Typische Konfiguration falls aktiviert: `[10000, 20000, 30000]` für 40K Classic. **ABER:** V549-A/B-Tests auf Family-Dataset zeigten in allen Konfigurationen schlechteres L1: `[10K, 20K, 30K]@0.01` → -48 % Count aber +36 % L1; `[20K, 30K]@0.005` → -44 % Count aber +45 % L1; `[20K, 30K]@0.001` → -17 % Count aber +87 % L1. Daher disabled. CLI-Flag `--mid-compact "10000,20000"` verfügbar, falls man den visuellen Floater-Tradeoff (weniger Confetti im Viewport) gegenüber dem Loss-Regression bevorzugt.

 PROSTYMI SŁOWAMI

Zwischendurch-Aufräumaktionen während des Trainings. In Tests hat das Aufräumen das Endergebnis schlechter gemacht (zwar weniger Floater, aber auch weniger Detail). Aus, kann via CLI eingeschaltet werden, falls Floater dich mehr stören als ein etwas matschigeres Bild.

T43 frustumCullEnabled **SZCZEGÓŁY**

Default: false **Range:** boolean **Defined in:**

 **TECHNICZNIE**

V549b-Feature: nach Training werden alle Gaussians entfernt, die außerhalb der Vereinigung aller Trainings-Kamera-Frusta liegen. Solche Gaussians wurden nie vom Loss-Signal eingeschränkt und sind immer Floater. Besonders effektiv für Szenen, in denen die Novel-View hinter oder neben dem Kamerapfad liegt (z.B. Rückseite eines linearen Drohnenflugs) — die Floater dort werden in der Trainingssphase nie sichtbar, beim späteren Bewegen im 3D-Viewer aber sehr wohl. V549b A/B auf Drohnenflügen positive Ergebnisse, daher als Opt-In verfügbar. Default false, weil bei Object-Captures mit voller Orbit-Coverage die Frustum-Union die ganze Szene umfasst und das Feature nichts entfernt — wird in Settings unter „Floater Reduction“ angeboten und auch in Q9 Outdoor-Preset implizit über T44 `frustumCullExpansion` getestet (Q7-BayesOpt hat es aber nicht aktiviert, weil Outdoor-Sky-Dome dasselbe Problem besser löst).

 **PROSTYMI SŁOWAMI**

Spezial-Filter für Drohnenflüge oder lineare Aufnahmen: nach dem Training werden Splats gelöscht, die in keiner Kamera „gesehen“ wurden. Optional einschaltbar in den Settings. Bei einfachen Objekt-Aufnahmen unnötig.

T44 frustumCullExpansion **SZCZEGÓŁY**

Default: 1.1 **Range:** 1.0 – 2.0 **Defined in:**

 **TECHNICZNIE**

NDC-Margin für T43 `frustumCullEnabled`. 1.0 würde exakt am Bildrand schneiden, was wackelige Splats am Bildrand zu sehr kürzen würde. 1.1 = 10 % Padding über die exakte Kamera-Framing hinaus — gibt etwas Toleranz für Randpixel, die in einer leicht versetzten Novel-View doch sichtbar werden könnten. Werte > 1.2 machen den Cull praktisch unwirksam, weil das erweiterte Frustum sehr viel mehr Raum umfasst.

 **PROSTYMI SŁOWAMI**

Wie streng der oben beschriebene Filter zuschneidet. 1.1 = ein bisschen Sicherheitsabstand zum Bildrand. Lass den Wert.

Sky-Dome (T45–T48)

T45 skyDomeEnabled

SZCZEGÓŁY

Default: false (Initializer + alle Presets außer P9 Outdoor) **Range:** boolean **Defined in:**

TECHNICZNIE

V549e-Feature: vor Training-Start wird eine kugelförmige Punktwolke generiert (Fibonacci-sphere mit T46 Sample-Points), in einem Radius von T47 $\text{skyDomeRadiusMultiplier} \times \text{scene_extent}$ um den Szenen-Mittelpunkt platziert und mit den Farben aus den sky-maskierten Pixeln aller Trainings-Kameras (siehe T20 `skyMaskingEnabled`) initialisiert. Diese Sky-Dome-Gaussians werden am Anfang des Gaussian-Buffers eingefügt und während Training „eingefroren“, (Position/Skala/Rotation-Gradienten = 0, nur SH und Opacity bleiben optimierbar). Effekt: statt schwarzer „Confetti“-Bereiche in der Ferne sieht der User in Novel-Views einen echten Himmel. V549e-MVP funktioniert auf Drohnen- und Landschaftsszenen sehr gut; in P9 Outdoor-Preset Default-on. Bei Innen-Szenen aus lassen — die Sphere würde sinnlos außerhalb des Raums hängen.

PROSTYMI SŁOWAMI

Schaltet eine künstliche „Himmelskuppel“, um die Szene ein. Macht Außenaufnahmen viel schöner: statt schwarzer Klümpchen am Bildrand zeigt die App den echten Himmel. Pflicht für Drohnenflüge und Landschaften, sinnlos für Innenräume.

T46 skyDomeSampleCount SZCZEGÓŁY

Default: 5 000 **Range:** 1 000 – 50 000 (typisch 2 000 – 10 000) **Defined in:**

 TECHNICZNI

Anzahl der Fibonacci-Sphere-Sample-Punkte auf der Sky-Dome-Sphere. Höhere Werte → dichteres Sky-Dome (besser bei großen Auflösungen und viel sichtbarem Himmel), aber mehr Speicherbedarf. 5 000 ist sweet spot für 4K-Renderings; bei niedrigeren Auflösungen reicht 2 000–3 000. Die Punkte werden nach Cosine-Distance zu jedem Trainings-Kamera-View-Vektor mit den entsprechenden sky-maskierten Pixeln initialisiert — Sample-Points, deren View-Cone keine Kamera sieht, bleiben mit niedrigem Opazitäts-Initialwert hinten, werden aber im Training nicht verändert (eingefroren).

 PROSTYMI SŁOWAMI

Wie dicht der künstliche Himmel ist. 5 000 Punkte reichen normalerweise. Mehr = besserer Übergang aus der Ferne, aber kostet etwas Speicher.

T47 skyDomeRadiusMultipller SZCZEGÓŁY

Default: 30.0 (Initializer + meisten Presets), 59.0 (P9 Outdoor, Q7-BayesOpt-Optimum) **Range:** 5.0 – 200.0 **Defined in:**

 TECHNICZNI

Radius der Sky-Dome-Sphere relativ zur Szenen-Ausdehnung (= mittlere Distanz zwischen Kamera-Positionen). 30 = die Kugel hat den 30-fachen Durchmesser der Kamera-Wolke. Zu klein (< 5) → Sky-Dome interferiert mit der Szene selbst (z.B. ein Sky-Dome-Splatt landet im Vordergrund); zu groß (> 100) → float32-Präzisionsverlust an den Sky-Dome-Positionen, was Render-Glitches in der Ferne auslöst. Q7-BayesOpt auf Bicycle (Mip-NeRF 360) fand 59.0 als szenenspezifisches Optimum für outdoor — das deutet darauf hin, dass die Standard-30.0 für tiefe Landschaften zu klein ist und die Sky-Dome-Pixel in den Bildrand-Bereichen sichtbar als „Wand“ rendern.

 PROSTYMI SŁOWAMI

Wie weit weg die künstliche Hemisphäre sein soll. 30 = ziemlich weit. Bei großen Landschaften ist 50–60 besser (Outdoor-Preset macht das automatisch). Zu klein wäre, als hätte man Klümpchen direkt vor der Linse.

T48 frozenGaussianCount SZCZEGÓŁY

Default: 0 (= keine eingefrorenen Gaussians) **Range:** 0 oder 1 – T46 **Defined in:**

 TECHNICZNIE

Anzahl der Gaussians am Anfang des Buffers, deren Position/Skala/Rotation-Gradienten im Optimizer auf null gesetzt werden — sie bleiben über das gesamte Training räumlich starr. Density-Control darf sie nicht klonen, splitten oder prunen. Genutzt für Sky-Dome-Injection (siehe T45): wenn Sky-Dome an ist, wird dieses Feld automatisch auf T46 skyDomeSampleCount gesetzt. Manuelles Setzen ist möglich (z.B. um eine vor-platzierte Punktwolke aus einem LiDAR-Scan einzufrieren), aber im UI nicht direkt zugänglich. Wichtig: die ersten N Gaussians im Buffer sind immer die frozen — die Reihenfolge im Buffer entscheidet, nicht ein expliziter Index.

 PROSTYMI SŁOWAMI

Wie viele Splats am Anfang fest sind und sich nicht bewegen dürfen. Wird automatisch auf die Sky-Dome-Anzahl gesetzt, wenn Sky-Dome an ist. Selbst dran drehen brauchst du nicht.

Adam + LR-Schedule (T49–T55)**T49** adamResetIteration SZCZEGÓŁY

Default: 0 (= deaktiviert) **Range:** 0 oder 100 – **Defined in:**

 TECHNICZNIE

V430-Feature: Iteration, an der die Adam-Optimizer-Momentum-Akkumulatoren (m1, m2) auf null zurückgesetzt werden. Bias-Korrektur danach läuft mit (iter - adamResetIteration) statt mit iter . V430 testete Reset bei 5 000 (nach Densification-Ende) → 12.8 % schlechter Loss. Grund: das Adam-Momentum, das sich während Densification aufgebaut hat, trägt Information über die typischen Gradient-Magnituden bei und beschleunigt die Refinement-Phase. Es wegzuwerfen kostet die ersten ~500 Iterationen Refinement an Konvergenz. Disabled. Bleibt CLI-Flag für Forschungsexperimente.

 PROSTYMI SŁOWAMI

Reset-Knopf für das interne Adam-Optimierer-„Gedächtnis“. Hat in Tests geschadet, bleibt aus.

T50 positionLRScheduleEndIteration SZCZEGÓŁY

Default: 0 (Initializer = „verwende maxIterations“), 20 000 (`.full` — Cosine endet bei 20K trotz maxIter=35K), 30 000 (`.fullClassicPaper`) **Range:** 0 oder 1 000 – **Defined in:**

 TECHNICZNIE

V431-Feature: Iteration, an der die Cosine-Annealing-Kurve für Position-LR ihr Minimum erreicht. Wenn 0, ist das identisch mit `T1 maxIterations`. Wenn > 0 , läuft die Schedule bis zu diesem Wert und bleibt danach bei `T4 positionLearningRateFinal` konstant. Das erlaubt eine „extended refinement phase“, mit minimaler aber konstanter Lernrate — verfeinert Positionen langsam ohne erneuten Decay. `.full` macht das (Schedule-Ende bei 20K, Training läuft bis 35K), V434c/V434d bestätigten: 15K und 25K beide etwa gleich, 20K minimal optimal. Wird in Verbindung mit `T51` weiterverwendet, um auch die Nicht-Position-LRs in der extended phase zu modifizieren.

 PROSTYMI SŁOWAMI

Wann die App aufhört, die Position-Lernrate weiter zu senken. Wenn niedriger als die maximale Iteration, läuft danach mit konstanter Mini-Rate — das verfeinert sehr langsam aber sehr stabil. In Quality-Preset eingebaut, brauchst nicht zu drehen.

T51 extendedPhaseLRDecay SZCZEGÓŁY

Default: 0.0 (= deaktiviert, konstante LRs) **Range:** 0 oder 0.01 – 1.0 **Defined in:**

 TECHNICZNIE

V433-Feature: minimaler Multiplikator für die Nicht-Position-LRs (Skala, Rotation, Opacity, SH) in der „extended phase“, — sprich: nachdem `T50` erreicht ist und Position-LR bereits bei `T4` ist. Wenn 0.1, werden Skala/Rotation/Opacity/SH ihrerseits cosine-decayed von 1.0 (= ihr Standard-LR) auf 0.1× ihres Standards. Wenn 0.0 (Default), bleiben sie konstant. V457 testete vollen Decay (0.0 = decay-bis-null) gegen kein-Decay und fand: avg 0.0400 (2 Runs) = gleicher Loss wie V438 ohne Decay. Behavior cleaner mit Decay, aber nicht messbar besser. Daher disabled. Bleibt im CLI als `--nonpos-lr-scale F`.

 PROSTYMI SŁOWAMI

In der späten Verfeinerungsphase auch Farb- und Form-Lernraten kleiner machen. Macht das Training „stabiler“, aber empirisch nicht besser. Aus.

T52 adaptiveDensifyThreshold**SZCZEGÓŁY****Default:** false **Range:** boolean **Defined in:****TECHNICZNI**

V440-Experimental: wenn true, berechnet die App in jedem Densification-Schritt das p98 der aktuellen Gradient-Verteilung und nutzt es als dynamischen Schwellwert (geclamped auf mindestens 0.5x des konfigurierten Werts aus T11, damit es nicht zu sehr ausreißt). Hypothese: Automatische Anpassung an aktuelle Szenen-Phase würde Density-Control robuster machen — z.B. zu Anfang strenger pruning, später lasser, oder umgekehrt. V440 testete und revertete: katastrophaler Drop auf 63 K Gaussians (Mass-Pruning, weil das p98 in den ersten Iterationen extrem hoch ist und dann fast nichts den Threshold überschreitet). Der fixe Threshold ist bereits gut kalibriert, dynamische Anpassung schadet mehr als sie nutzt. Q5 (T77) bietet eine alternative Adaptive-Logik via rolling median, die das Problem umgeht.

PROSTYMI SŁOWAMI

Adaptive Versions des Densify-Schwellwerts. In Tests katastrophal (Splat-Anzahl auf 63K abgestürzt). Aus. Q5 hat eine bessere Variante davon.

T53 mergeAfterDensification**SZCZEGÓŁY****Default:** false (Initializer), true (`.full`, `.classicBalanced`, `.fullClassicPaper`) **Range:** boolean **Defined in:****TECHNICZNI**

V438-Feature: am Ende der Densification-Phase (Iter T2) wird ein einmaliger Merge-Pass durchgeführt, der nahe beieinanderliegende Gaussians mit ähnlicher Skala und Farbe zusammenfasst. Reduziert die Gaussian-Anzahl um typisch 5–15 % ohne sichtbaren Qualitätsverlust. Sinn: nach intensivem Klonen entstehen Cluster von quasi-identischen Gaussians, die nichts Neues beitragen — das Merging gibt Optimizer-Kapazität für andere Bereiche frei. Standard in Classic-Quality-Presets. Bei MCMC nicht verwendet, weil MCMC durch seine Relocation-Logik solche Cluster gar nicht erst entstehen lässt.

PROSTYMI SŁOWAMI

Am Ende der Splat-Vervielfältigungs-Phase Klone, die fast identisch sind, zusammenfassen. Reduziert Datenmenge ohne sichtbaren Effekt. Standardmäßig in Quality-Preset an.

T54 densifyPhase2FromIteration SZCZEGÓŁY**Default:** 0 (= deaktiviert) **Range:** 0 oder T2 – T1**Defined in:** TECHNICZNIE

V426-Experimental: ermöglicht eine zweite Densification-Phase, die nach der Refinement-Pause bei dieser Iteration startet und bis T55 läuft. Hypothese: nach einer Refinement-Phase haben die Gradient-Akkumulatoren stabilere Magnitudes und können präziser sagen, welche Bereiche noch zusätzliche Gaussians brauchen. V426 testete und revertete: Two-Phase Densification fiel in 0-Gaussians-Cascade-Failure (mit V425 Refinement-Pruning kombiniert zerstörte es den Buffer). Disabled. CLI-Flag verfügbar für Experimente.

 PROSTYMI SŁOWAMI

Zweite Vervielfältigungsrunde nach Pause. Hat in Tests den Splat-Bestand vernichtet. Aus.

T55 densifyPhase2Untilliteration SZCZEGÓŁY**Default:** 0 **Range:** 0 oder T54 – T1 **Defined in:** TECHNICZNIE

Ende der V426-Two-Phase-Densification. Nur relevant wenn T54 > 0. Beide Felder zusammen disabled.

 PROSTYMI SŁOWAMI

Ende der zweiten Vervielfältigungsrunde (siehe T54). Beide aus.

Post-Processing + Apple AI (T56–T60)

T56 postTrainingCompactification

☰ SZCZEGÓŁY

Default: true (in allen Production-Presets), false (`.quickTest`, `.preview`) **Range:** boolean **Defined in:**

☰ TECHNICZNI

V443-Feature: nach Trainingsende werden Gaussians mit $\text{sigmoid}(\text{opacity}) < 0.01$ hart entfernt (sie tragen praktisch nicht mehr zum Bild bei). Reduziert Gaussian-Count um typisch 58 % und Export-Dateigröße um 55 % ohne sichtbaren Qualitätsverlust. Standardmäßig in Production-Presets aktiv — das Endresultat soll möglichst kompakt ausgeliefert werden können. In `.quickTest` aus, weil ein Diagnose-Lauf sowieso nicht exportiert wird. Anders als T42 `midTrainingCompactificationIterations` (V549) findet die Compactification erst am Ende statt — Refinement kann bis dahin alle Gaussians benutzen.

💬 PROSTYMI SŁOWAMI

Aufräumen nach dem Training: nahezu unsichtbare Splats werden entfernt. Macht die Export-Datei rund halb so groß ohne Qualitätsverlust. Pflicht-Feature, aus lassen nur in Diagnose-Läufen.

T57 metalFXUpscaling

☰ SZCZEGÓŁY

Default: false **Range:** boolean **Defined in:**

☰ TECHNICZNI

V444-Feature: aktiviert Apples MetalFX Spatial Upscaler statt bilineare Interpolation im 3D-Viewer-Output. Wenn Trainingsauflösung $<$ Viewport-Größe (z.B. Training auf 0.5x, Viewport-Anzeige in voller Auflösung), kann MetalFX ein deutlich schärferes Bild liefern. Ändert sich live im Viewport, kein Re-Training nötig. Schließt sich mit T58 `mpsLanczosScaling` aus — MetalFX hat Vorrang. Empfehlung: einschalten, wenn das Bild im Viewer „verwaschen“, wirkt im Vergleich zum erwarteten Detail.

💬 PROSTYMI SŁOWAMI

Apple-ML-basierte Bild-Schärfung im 3D-Viewer. Hilft, wenn du in einer niedrigeren Auflösung trainiert hast und das Ergebnis im Vollbild zeigst. Live-Toggle, probier aus.

T58 mpsLanczosScaling SZCZEGÓŁY

Default: false **Range:** boolean **Defined in:**

 TECHNICZNIE

V444-Feature: MPSImageLanczosScale für Viewport-Skalierung statt bilineare Interpolation. Lanczos ist ein klassisches Sinc-basiertes Resampling-Verfahren, das deutlich schärfere Ergebnisse als Bilinear liefert mit minimalem Overhead. Live-Toggle. Wird von T57 überschrieben, wenn beide an.

 PROSTYMI SŁOWAMI

Klassisches Schärfungsverfahren für den 3D-Viewer (Lanczos). MetalFX (T57) ist ML-basiert und meist besser; Lanczos ist eine weniger aggressive Alternative.

T59 livePreviewInterval SZCZEGÓŁY

Default: 50 (Initializer und meisten Presets) **Range:** 0 (off) oder 10 – 5 000 **Defined in:**

 TECHNICZNIE

Wie oft während des Trainings der 3D-Viewer mit den aktuellen Gaussians aktualisiert wird. 50 = alle 50 Iterationen ein neues Render im Viewer — gut genug, um den Fortschritt zu beobachten, ohne das Training zu verlangsamen. 0 = Viewer wird gar nicht aktualisiert (Hintergrund-Training, max Geschwindigkeit). Typische Anpassung: bei `.quickTest` runter auf 10 (man will jeden Schritt sehen), bei langen MCMC-Läufen hoch auf 500–2000 (Update-Overhead in Summe spürbar).

 PROSTYMI SŁOWAMI

Wie oft die 3D-Vorschau während des Trainings aktualisiert wird. 50 = alle 50 Iterationen. Höher = weniger oft = etwas schneller, aber du siehst seltener Fortschritt. 0 = keine Vorschau (für maximales Tempo).

T60 perceptualLossWeight SZCZEGÓŁY

Default: 0.0 (= deaktiviert) **Range:** 0 oder 0.001 – 0.5 **Defined in:**

 TECHNICZNI

V444-Future-Feature: Gewicht eines perceptuellen Loss-Terms via MPSGraph (VGG-ähnliches kleines Netzwerk). Würde strukturelle und textuelle Ähnlichkeit auf einer höheren semantischen Ebene als L1+SSIM erfassen — typisch in Forschungs-Pipelines, wo „pixel-perfect“, weniger wichtig ist als „sieht realistisch aus“. Implementation noch ausstehend (Code-Stub vorhanden, aber Forward-Pass nicht implementiert). Default 0.0. Bleibt im Feldkatalog für künftige Aktivierung; CLI-Flag `--percep-weight F` reserviert.

 PROSTYMI SŁOWAMI

Geplantes Feature, das mit KI-Hilfe „natürlich aussehend“, anstrebt statt „pixelgenau“. Noch nicht fertig implementiert.

MCMC-Densification (T61–T73)**T61** densificationStrategy SZCZEGÓŁY

Default: `.classic` (Initializer + Classic-Presets), `.mcmc` (alle MCMC-Presets + Scene-Class) **Range:** `.classic` oder `.mcmc` **Defined in:**

 TECHNICZNI

Wählt zwischen Classic-Densification (Klon/Split/Prune, Kerbl et al. 2023) und MCMC-Densification (Stochastic Gradient Langevin Dynamics mit Relocation, Kheradmand et al. NeurIPS 2024). Bei `.classic` werden T11–T16 ausgewertet, bei `.mcmc` die T62–T73. Achtung beim Wechsel: Classic Defaults und MCMC Defaults sind völlig anders kalibriert — wer den Picker im Expert View flippt, ohne ein passendes Preset zu laden, riskiert 1.4.3-Bug-Style Mass-Extinction (460 K → 5 in einer Iteration, weil MCMC-OpacityReg auf 0.01 die Classic-Opacities killt). Daher die MCMC-Init Defaults absichtlich „weichgespült“, (alle Reg-Werte 0.0).

 PROSTYMI SŁOWAMI

Welcher Algorithmus zum Vermehren der Splats verwendet wird. Classic = ursprüngliche Methode (schnell, viele Splats). MCMC = neuere Methode (langsamer, viel weniger Splats, dafür kompakter). Die Presets wählen das richtige. Selbst nur umschalten, wenn du auch das passende Preset (P5–P7 oder P8–P10) lädst.

T62 mcmcMaxGaussians**SZCZEGÓŁY**

Default: 150 000 (Initializer + `.fullMCMC` + `.mcmcBalanced`), 100 000 (`.mcmcPreview`), 1 500 000 (`.fullMCMCMip` — Mip-Splatting-Variante mit 10x Budget), 1.19 M (`.renderPreset`), 1.25 M (`.outdoorPreset`), 670 K (`.indoorPreset`) **Range:** 0 (= „verwende Buffer-Kapazität,“) oder 10 000 – 5 000 000 **Defined in:**

TECHNICZNI

Harte Obergrenze für die Anzahl der Gaussians bei MCMC-Strategie. Die Anzahl wächst graduell um `T70 mcmcGrowthRate` (typisch 5 %) pro Relocation-Step bis zu diesem Cap. V473/V531 fanden 150 K als sweet spot — über 200 K dilutet die Splat-Qualität (zu viele kleine, redundante Gaussians), unter 100 K bleibt die Szene unter-densifiziert. Bei sehr großen Szenen (z.B. 1 545-Foto-Drohnenflug mit 158 K SfM-init) ist 150 K zu niedrig — daher die 1.4.5-Erweiterung `T72 mcmcCapMultiplier` + `T73 mcmcAutoScaleByScene`. Q7-BayesOpt fand szenenspezifische Optima zwischen 670 K (Indoor) und 1.25 M (Outdoor). Bei Wert 0 verwendet die Engine die volle Buffer-Kapazität als Cap.

PROSTYMI SŁOWAMI

Maximale Anzahl Splats bei MCMC. 150 000 ist der Standard und reicht für die meisten Szenen. Outdoor- und Render-Presets (P8, P9) gehen auf 1+ Million für Detail-reichere Szenen. Hochsetzen kann Detail bringen, kostet Speicher; runter ist eher eine Notbremse.

T63 **mcmcNoiseScale** **SZCZEGÓŁY**

Default: 0.00005 (5e-5 = Paper-Default) **Range:** 1e-6 – 1e-3 **Defined in:**

 **TECHNICZNIE**

Multiplikator für das Gauss'sche Rauschen, das in jeder MCMC-Iteration zur Position jeder Gaussian addiert wird (SGLD-Logik). Höher = mehr Exploration (Gaussians wandern mehr, finden potenziell bessere Plätze), niedriger = mehr Exploitation (Gaussians bleiben da, wo sie schon gut sind). V467 und V536 bestätigten 5e-5 als optimal — 1e-5/2e-5 zu wenig Exploration, 1e-4 zu viel (Splats zerlaufen). Wird über die Trainingszeit cosine-decayed bis T69 `mcmcNoiseDecayEnd` — am Ende des Decay-Bereichs ist Noise effektiv 0 und die Gaussians konvergieren.

 **PROSTYMI SŁOWAMI**

Wie viel zufälliges „Wackeln“, die App den Splats erlaubt, damit sie selber den besten Platz finden. Standardwert ist optimal getestet. Wenn du das hochdrehst, werden die Splats unruhig.

T64 **mcmcOpacityRegWeight** **SZCZEGÓŁY**

Default: 0.0 (= deaktiviert in den RadianceKit Defaults, Paper: 0.01) **Range:** 0 oder 0.001 – 0.05 **Defined in:**

 **TECHNICZNIE**

MCMC-spezifische L1-Penalty auf Opacity. Paper-Default 0.01 (drückt unbenutzte Gaussians gegen Null, macht sie für Relocation verfügbar). V464b zeigte aber: ohne Reg ist es in RadianceKit messbar besser (Session 28 bestätigt). Grund: Das mit T68 `mcmcDeadOpacityThreshold` definierte Pruning-Kriterium reicht alleine — eine zusätzliche L1-Penalty zwingt auch wertvolle, niedrig-Opacity-Gaussians zum Sterben. Daher Default 0. **Achtung:** in 1.4.3-Beta-Build war der Initializer-Default fälschlich 0.01, was im Mass-Extinction-Bug resultierte (siehe T61-Erläuterung); seit 1.4.4 auf 0.0 fixiert.

 **PROSTYMI SŁOWAMI**

MCMC-Spezialregularisierung. Aus, weil der andere MCMC-Mechanismus (Schwellwert in T68) das schon abdeckt. Auf 0 lassen.

T65 **mcmcScaleRegWeight** SZCZEGÓŁY

Default: 0.0 (= deaktiviert, Paper: 0.01) **Range:** 0 oder 0.001 – 0.05 **Defined in:**

 TECHNICZNIE

MCMC-spezifische L1-Penalty auf die Skala-Eigenwerte. Paper-Default 0.01. V464b: ohne Reg besser, gleiche Begründung wie T64. Disabled in allen RadianceKit-MCMC-Presets. Achtung wie bei T64: 1.4.3-Bug.

 PROSTYMI SŁOWAMI

Wie T64, aber für Splat-Größe. Aus.

T66 **mcmcRelocationInterval** SZCZEGÓŁY

Default: 100 (Initializer + alle MCMC-Presets, Paper-Standard), 155 (P9 Outdoor — Q7-BayesOpt-Optimum) **Range:** 50 – 500 **Defined in:**

 TECHNICZNIE

Iterations-Intervall, in dem MCMC tote Gaussians ($\text{sigmoid}(\text{opacity}) < T68 \text{ mcmcDeadOpacityThreshold}$) zu neuen Positionen relociert. V537 testete 50 (zu disruptiv, Loss schwankt) und 200 (marginal schlechter, MCMC verliert Reaktionsfähigkeit). 100 ist optimal. Q7-BayesOpt auf Bicycle fand 155 als szenenspezifisches Optimum für outdoor — die etwas längeren Intervalle geben Adam mehr Zeit, neu platzierte Gaussians zu integrieren, bevor das nächste Reloc-Event sie unter Druck setzt.

 PROSTYMI SŁOWAMI

Alle wie viele Iterationen MCMC die toten Splats irgendwo anders hin verschiebt. 100 ist Standard. Selbst dran drehen brauchst du nicht — Outdoor-Preset hat schon den optimalen Wert.

T67 **mcmcWarmupIterations** **SZCZEGÓŁY****Default:** 500 **Range:** 100 – 5 000 **Defined in:** **TECHNICZNIE**

Anzahl der Initial-Iterationen, in denen noch keine MCMC-Relocation passiert. Erst nach diesem Warmup beginnt die Reloc-Logik. Sinn: in den ersten Iterationen sind die Opacity-Werte noch nicht eingeschwungen — würde direkt mit Reloc gestartet, würden Gaussians an den falschen Stellen platziert und müssten gleich wieder bewegt werden, was Adam-Momentum zerstört. Paper-Default 500. RadianceKit übernimmt diesen Wert, weil V464b zeigte dass es robust ist.

 **PROSTYMI SŁOWAMI**

Wie viele Iterationen MCMC erst „ankommen“, darf, bevor es anfängt, Splats umzustellen. 500 ist Standard und passt.

T68 **mcmcDeadOpacityThreshold** **SZCZEGÓŁY****Default:** 0.005 (Initializer, Paper-Standard), 0.01 (`.fullMCMC` und alle MCMC-Presets — V535-Optimum) **Range:** 0.001 – 0.05 **Defined in:** **TECHNICZNIE**

sigmoid(Opacity)-Schwellwert, unter dem eine Gaussian als „tot“, gilt und für Relocation in Frage kommt. V535 fand 0.01 als optimal (0.005 marginal, 0.02 schlechter). Höher = aggressiveres Reloc (mehr Gaussians werden bewegt), niedriger = vorsichtiger. 0.01 entspricht ungefähr „0.5 % visuelle Sichtbarkeit“. P10 Indoor verwendet via Q7-BayesOpt 0.0142 als Optimum.

 **PROSTYMI SŁOWAMI**

Ab welcher Durchsichtigkeit ein Splat als „tot“, gilt, sodass MCMC ihn woanders hinschiebt. 0.01 ist optimal in unseren Tests. Selbst dran drehen brauchst du nicht.

T69 **mcmcNoiseDecayEnd** **SZCZEGÓŁY**

Default: 0 (Initializer = „kein Decay“), 160 000 (`.fullMCMC` = 80 % von 200K), 96 000 (`.mcmcBalanced` = 80 % von 120K), 40 000 (`.mcmcPreview`) **Range:** 0 oder 1 000 – **Defined in:**

 **TECHNICZNIE**

Iteration, an der das `T63 mcmcNoiseScale`-Rauschen vollständig auf null gedämpft wird (Cosine-Decay von Iter 0 bis hier). V497c/V502 fanden 80 % der `maxIterations optimal` — gibt MCMC genug Exploration-Zeit, lässt aber die letzten 20 % zur Konvergenz ohne Rauschen. 0 = konstantes Rauschen über alle Iterationen (selten sinnvoll, MCMC kann dann nicht konvergieren).

 **PROSTYMI SŁOWAMI**

Wann das zufällige „Wackeln“ der Splats aufhört. In den MCMC-Presets bei 80 % der Gesamtiterationen — erst Exploration, dann Konvergenz. Lass den Wert.

T70 **mcmcGrowthRate** **SZCZEGÓŁY**

Default: 0.05 (Paper-Standard = 5 %) **Range:** 0.01 – 0.2 **Defined in:**

 **TECHNICZNIE**

Wachstumsrate des MCMC-Populations-Targets pro Relocation-Step. Die Logik: bei jedem Reloc-Event wird das Ziel-Populations-Größe um $(1 + \text{growthRate})$ erhöht, bis `T62 mcmcMaxGaussians` (oder die per `T72/T73` skalierte Variante) erreicht ist. V512/V522 fanden 0.05 als optimal — höhere Werte führen zu zu schnellem Wachstum (Gaussians werden eingefügt, bevor das Adam-Momentum sie integrieren kann), niedrigere zu unter-densifizierten Szenen am Ende.

 **PROSTYMI SŁOWAMI**

Wie schnell die Splat-Anzahl bei MCMC wächst. 5 % pro Schritt ist optimal. Lass den Wert.

T71 mcmcSigmoidK**SZCZEGÓŁY****Default:** 100.0 **Range:** 10.0 – 500.0 **Defined in:****TECHNICZNI**

Sigmoid-Sharpness-Parameter für die MCMC-Noise-Attenuation. Im SGLD-Schritt wird das pro-Gaussian-Rauschen durch gedämpft — hoch-opake Gaussians (deren Logit positiv ist) bekommen exponentiell weniger Rauschen als niedrig-opake. $K = 100$ ist scharf, sprich der Übergang von „voll-Noise“ zu „kein-Noise“ passiert sehr schnell um Opacity 0.5. V484–V487 fanden $K = 100$ optimal — kleinere Werte (10–50) lassen auch hoch-opake Gaussians mit-Wackeln (zerstört konvergierte Gaussians), größere (> 500) machen den Übergang künstlich hart und tote Gaussians werden gar nicht mehr bewegt.

PROSTYMI SŁOWAMI

Spezialparameter, der bestimmt, wie scharf MCMC zwischen „durchsichtig genug zum Verschieben“ und „solid, nicht anrühren“ unterscheidet. Standardwert ist optimal. Nicht dran drehen.

T72 mcmcCapMultiplier**SZCZEGÓŁY****Default:** 3.0 (Initializer + `.fullMCMC`), 2.0 (`.mcmcPreview`), 2.5 (`.mcmcBalanced`), 2.98 (P8 Render), 5.32 (P9 Outdoor), 1.76 (P10 Indoor) **Range:** 0 (= deaktiviert) oder 1.0 – 10.0 **Defined in:****TECHNICZNI**

1.4.5-Feature: szenen-adaptive Cap-Skalierung. Wenn T73 `mcmcAutoScaleByScene` true ist, wird der effektive Cap als berechnet (geclamped an Buffer-Kapazität). Hintergrund: bei großen Szenen (z.B. 1 545-Foto-Drohnenflug → 158 K SfM-init) ist `T62 = 150 000` zu niedrig — Density-Control würde gar nicht wachsen können. Mit Multiplier 3.0 wird der Cap bei diesem Beispiel auf 474 K skaliert ($158\text{ K} \times 3.0$). Q7-BayesOpt fand szenenspezifische Optima: Outdoor profitiert von hohem Multiplier (5.32 → ~830 K Cap bei 156 K bicycle-init), Indoor begnügt sich mit 1.76 (Wände sättigen schneller). Komplette Auflösung des Caps siehe -Methode.

PROSTYMI SŁOWAMI

Multiplikator, der den Splat-Cap automatisch an die Szenen-Größe anpasst. Große Szene = mehr Anfangspunkte = höherer Cap. Standard 3× passt für die meisten Szenen; Outdoor-Preset geht auf 5× (große Tiefenbereiche), Indoor auf 1.76× (Wände begrenzen ohnehin).

T73 **mcmcAutoScaleByScene** **SZCZEGÓŁY**

Default: true (Initializer + alle MCMC-Presets) **Range:** boolean **Defined in:**

 **TECHNICZNI**

1.4.5-Feature: Master-Switch für die scene-aware Cap-Logik (siehe T72 +). Wenn false, wird ausschließlich T62 `mcmcMaxGaussians` als Cap verwendet (zurück zu 1.4.4-Verhalten). Standardmäßig an, weil die Mass-Extinction-Probleme bei großen Szenen aus 1.4.3 sonst wiederkehren. Manuell deaktivieren nur, wenn du explizit ein hartes Cap setzen willst — z.B. um eine 150 K-Variante zu trainieren, deren Endgröße planbar ist.

 **PROSTYMI SŁOWAMI**

Schaltet die automatische Anpassung des Splat-Caps an die Szenen-Größe an. Standardmäßig an. Aus lassen nur, wenn du selbst genau eine bestimmte Splat-Anzahl haben willst.

Mip-Splatting (Q1.5) (T74–T76)

Status: Q1.5 wurde am 2026-05-25 nach 14 autonomen Iterationen + Overnight-1.5M-Confidence-Check als „closed no-win“, verworfen (max $\Delta@2\times = +0.27$ dB, Original-Gate verlangte $\geq +1.5$ dB Mittelwert über $0.5\times/2\times$, FAILT auf 0/11 Pair-Scenes). Die Felder bleiben **opt-in** für Forschungs-Experimente; alle Production-Presets haben. Siehe Verdict: `docs/plans/2026-05-25-phase-q1.5-final-verdict.md`.

T74 useMipSplatting**SZCZEGÓŁY**

Default: false (alle Production-Presets), true (`.fullMCMCMip` — Forschungs-Sibling) **Range:** boolean **Defined in:**

TECHNICZNI

Aktiviert Mip-Splatting (Yu et al. CVPR 2024): 3D-Smoothing-Filter + 2D-Filter + α -Kompensation, der die per-Gaussian-Frequenz auf die Nyquist-Grenze der dichtesten Trainingskamera-Sampling-Rate begrenzt. Theoretisches Ziel: Eliminierung von Aliasing bei Rendering in off-training-Skalen (0.5x oder 2x der Trainingsauflösung). In den Preprocess- und Backward-Projection-Shadern aktiviert, funktional korrekt verifiziert in Q1.5-D-Test. Aber: der Original-Akzeptanz-Gate ($\Delta@1x \geq +0.3$ dB UND $\text{avg}(\Delta@0.5x, \Delta@2x) \geq +1.5$ dB) wurde auf keinem von 11 Pair-Scenes erreicht. Maximal beobachtet: family 750K classic $\Delta@2x = +0.270$ dB. Outdoor-Szenen (Truck, Flowers) zeigten sogar Verschlechterung 1x und 0.5x. Hypothese: 3D-Smoothing konkurriert mit MCMC-Relocation bei high-Gs. Feld bleibt für künftige Multi-Scale-Re-Eval mit korrekter Mip-NeRF-360-Methodology (siehe O3-Backlog im Benchmark-Pfad).

PROSTYMI SŁOWAMI

Aliasing-Filter aus einem 2024er-Paper. Theoretisch toll, praktisch hat er in unseren Tests nichts gebracht und manchmal sogar geschadet. Bleibt verfügbar für Experimentierer, aber wir empfehlen ihn nicht. Aus lassen.

T75 mipSmoothing3DScale**SZCZEGÓŁY**

Default: 0.2 (Paper-Default) **Range:** 0.05 – 1.0 **Defined in:**

TECHNICZNI

3D-Smoothing-Skala-Parameter (Yu et al. §3.3, Paper-Default 0.2). Größer = mehr Welt-Raum-Glättung pro Gaussian (= mehr Anti-Aliasing, aber auch mehr blur in der Default-Skala), kleiner = schärfer aber anfälliger für Aliasing. Wird nur konsultiert, wenn `T74 useMipSplatting = true`. In Q1.5-Tests nicht weiter optimiert — die A/B-Gate hat schon mit Paper-Default 0.2 verloren, weitere Sweeps wären zwecklos.

PROSTYMI SŁOWAMI

. Wenn du Mip nicht eingeschaltet hast, irrelevant.

T76 mipFilter2DVariance**SZCZEGÓŁY****Default:** 0.3 (= exakt das V242-Legacy-Verhalten)**Range:** 0.1 – 1.0 **Defined in:****TECHNICZNI**

2D-Mip-Filter-Varianz, die zur Σ_{2D} -Diagonale addiert wird (Varianz direkt, nicht quadriert). 0.3 ist exakt der V242-Legacy-Wert, der vor Mip-Splatting hardcoded im Kernel war. Wenn `T74 useMipSplatting = false`, ignoriert der Kernel diesen Wert komplett und schreibt das hartkodierte 0.3 — sodass die Baseline nicht regredieren kann (Cortex-Round-1-S3-1-Garantie). Wenn, wird der hier gesetzte Wert verwendet. Bleibt im Feld-Katalog für Mip-Sweeps.

PROSTYMI SŁOWAMI

Weiterer Mip-Splatting-Parameter. Bei Mip-aus: irrelevant.

Adaptive Densification (Q5) (T77–T79)**T77 adaptiveDensification****SZCZEGÓŁY****Default:** false **Range:** boolean **Defined in:****TECHNICZNI**

Q5-Feature: rolling-median-Tracker als Alternative zum fixen `T11 densifyGradThreshold`. Wenn true, wird in jedem Densify-Step der aktuelle Schwellwert mit $\text{median}(\text{letzte } N \text{ avgGrad-Samples}) \times \text{T79 adaptiveDensifyMultiplier}$ überschrieben. $N = \text{T78 adaptiveWindow}$. Strikter als V440 p98 (die katastrophale 63 K-Pruning-Falle), median + 2× sitzt etwa beim p70–p80 der Gradient-Verteilung in steady state. Q5-Tests: alleinstehend FAIL 0/3 Szenen, aber gemeinsam mit Q6 (siehe T80/T81) PASS 1/3 Szenen — das Bundle Q5+Q6 wurde 2026-05-25 als opt-in gepasst und ist via CLI `--adaptive-densify` aktivierbar. Q6 ist dabei der „Carrier“, des Quality-Gewinns, Q5 trägt eher zur Stabilität bei.

PROSTYMI SŁOWAMI

Selbstlernender Densify-Schwellwert. Statt einer fest eingestellten Empfindlichkeit passt die App sich der Szene an. Alleine getestet nicht besser, mit dem Curriculum aus Q6 zusammen aber ja. Beide gemeinsam einschalten oder beide aus.

T78 adaptiveWindow SZCZEGÓŁY**Default:** 1 000 **Range:** 100 – 10 000 **Defined in:** TECHNICZNI

Rolling-Median-Window in Densification-Events (NICHT Iterationen — jeder T13 `densifyInterval-Step` liefert ein Sample). Default 1 000 — bei heißt das die letzten 100 000 Trainings-Iterationen tragen zum Median bei, also typisch die gesamte Trainingshistorie bis hierhin. Frühe Phase (vor T78 Samples): Tracker returns nil → Fallback auf fixen Threshold T11. Nur relevant wenn.

 PROSTYMI SŁOWAMI

Wie viele alte Densify-Schritte in den Median für T77 einfließen. Standard 1000 ist gut. Nur relevant wenn Q5-Adaptive an.

T79 adaptiveDensifyMultiplier SZCZEGÓŁY**Default:** 2.0 **Range:** 1.0 – 4.0 **Defined in:** TECHNICZNI

Multiplikator auf den Rolling-Median für den adaptiven Schwellwert. Default 2.0 entspricht ungefähr p70–p80 der typischen Gradient-Verteilung. Niedriger = aggressiveres Wachstum (mehr Klone), höher = strenger (weniger Klone). Q5-Tests in Range 1.5–3.0 — 2.0 bestes Default. Nur relevant wenn.

 PROSTYMI SŁOWAMI

Faktor für T77/T78. Standard 2.0 = strenger als der typische Median. Nicht dran drehen.

Curriculum (Q6) (T80–T81)

T80 curriculumResolutionRamp

SZCZEGÓŁY

Default: false **Range:** boolean **Defined in:**

TECHNICZNI

Q6-Feature: Trainings-Auflösung startet bei 0.5x und wechselt bei T50 $\text{positionLRScheduleEndIteration} / 2$ (oder $\text{T1} \text{maxIterations} / 2$, falls T50 nicht gesetzt) auf T22 $\text{trainingRenderScale}$. Verwendet die in Q1.5.1 entwickelte `resize/restoreImageBuffers`-Infrastruktur. Überschreibt T23 `resolutionWarmupScale`, wenn aktiviert. Q6 ist als „Carrier des Quality-Gewinns“, im Q5+Q6-Bundle gepasst (siehe T77) — die schrittweise Auflösungserhöhung gibt der App Zeit, grobe Geometrie auf der niedrigeren Auflösung zu finden, bevor sie zur feinen Detailarbeit übergeht. Über CLI: `--curriculum-resolution`.

PROSTYMI SŁOWAMI

„Erst grob, dann fein,“ für die Trainingsauflösung. Halbe Auflösung in der ersten Hälfte, dann volle Auflösung. Hilft in bestimmten Szenen, in anderen nicht — am besten mit T81 zusammen einschalten.

T81 curriculumSHProgression

SZCZEGÓŁY

Default: false **Range:** boolean **Defined in:**

TECHNICZNI

Q6-Feature: überschreibt T21 `shDegreeUpgradeIterations` mit $[\text{maxIter}/4, \text{maxIter}/2, \text{maxIter} * 3/4]$, verteilt also die SH-Aufstufungen gleichmäßig über die Trainingszeit statt sie front-zu-loaden. Hypothese: stabile Geometrie wird vor Color-Detail-Explosion etabliert, was die View-Direction-abhängigen Glanz-Effekte präziser positioniert. Q5+Q6 zusammen PASS 1/3 Szenen, Q6 als Carrier des Gewinns (Q5 alone FAIL). Via CLI: `--curriculum-sh`.

PROSTYMI SŁOWAMI

„Erst Form, dann Farbe,“ — die Glanz-Effekte werden erst spät im Training freigeschaltet, so dass die Splats zuerst ihre Position und Größe finden. Mit T80 zusammen einschaltbar; alleine bringt es nicht ganz so viel.

Statische Presets (TP1–TP9)

Hier nur die strukturellen Unterschiede zum Initializer-Default. Die volle Marketing-Beschreibung der elf UI-Presets P1–P11 findest du in Kapitel 7.

TP1 `.preview`

SZCZEGÓŁY

Diagnose-/Vorschau-Preset für Systeme ≥ 10 GB RAM. Overrides gegenüber Initializer: - 30 000 \rightarrow 5 000 - 15 000 \rightarrow 3 500 (70 % von maxlter) - 1.6e-6 \rightarrow 1.6e-5 (10x höher, weniger aggressiver Decay) - „,,“, jeweils 2x (V176) - 3 000 \rightarrow 100 000 (effektiv aus, V172: Reset zerstört kurze Trainings) - [1K, 2K, 3K] \rightarrow [1K, 2K] (V182: Degree 3 konvergiert in 2K lter nicht) - 1.0 \rightarrow 0.5

PROSTYMI SŁOWAMI

jede beliebige Initial-Begutachtung einer neu importierten Bildreihe — 2–3 min Wartezeit, anschließend reicht das Ergebnis für eine binäre Frage „lohnt sich Quality-Lauf?“,,

TP2 `.full`

SZCZEGÓŁY

Production-Quality Classic. Overrides: - 30 000 \rightarrow 35 000 (V550: 40K-Tests Truck-Overtraining +10.7 % Gs bei -1.3 % L1) - 15 000 \rightarrow 5 000 (V310 sweet spot, V338 7K worse) - Alle LRs 2x (V188) - 1.6e-6 \rightarrow 1.6e-5 (V45 10x) - 2e-6 \rightarrow 1.1e-6 (V335) - 100 \rightarrow 200 (V112) - 0.005 \rightarrow 0.001 (V393) - 3 000 \rightarrow 100 000 (V194 disabled, V421 confirmed) - [1K, 2K, 3K] \rightarrow [2K, 5K, 8K] (V228 delayed) - 0.0 \rightarrow 0.9995 (V546 HTGS, 14 % Verbesserung) - 50 (unverändert, V546) - false \rightarrow true (V438) - 0 \rightarrow 20 000 (V431) - true (V443, schon Initializer-Default für `.full`)

PROSTYMI SŁOWAMI

jede Standard-Foto-Aufnahme (Objekt, kleiner Raum, Skulptur) mit < 500 Bildern. Die in V546 angekündigte 14 % Loss-Verbesserung gegen V438 wurde 3-Trial-gemittelt auf Horse Full bestätigt.

TP3 `.fullClassicPaper`

SZCZEGÓŁY

Q1.5-A-Test-Sibling von TP2, paper-treuer Classic. Overrides gegenüber TP2: - 35 000 \rightarrow 30 000 (Paper-Standard) - 5 000 \rightarrow 15 000 (Paper: 50 % von maxlter) - 1.6e-5 \rightarrow 1.6e-6 (Paper-Default) - „,,“, zurück auf Paper Defaults (0.05, 0.005, 0.001) - 1.1e-6 \rightarrow 2e-7 (kalibriert für ~ 1 -2M Gs auf Bicycle) - 200 \rightarrow 100 (Paper) - 0.001 \rightarrow 0.005 (Paper-Default) - 100 000 \rightarrow 3 000 (Paper §5.2, riskant — kann V194-Regression auslösen) - 0.9995 \rightarrow 0.0 (Paper hat keinen Decay) - 20 000 \rightarrow 30 000 (cosine läuft auf 100 % von maxlter)

PROSTYMI SŁOWAMI

Q1.5-Forschungs-Experimente, die Paper-Magnitude-Gaussian-Budgets (1–2 M) für Mip-Splating-Tests brauchen. Nach Q1.5-„closed no-win“-Verdict bleibt das Preset für advanced users zugänglich, ist aber nicht Production-empfohlen.

TP4 `.fullMCMC`**SZCZEGÓŁY**

Production-Quality MCMC. Overrides gegenüber Initializer: - 30 000 → 200 000 (V534, MCMC braucht 5x mehr Iter als Classic) - 15 000 → 160 000 (V504b 80 % von maxIter) - 1.6e-6 → 1.6e-5 - LR-Schedule wie TP2 (alle 2x) - 0.2 → 0.05 (V521b/V534: MCMC braucht stärkeres L1-Signal) - [1K, 2K, 3K] → [2K, 5K, 8K] - `.classic` → `.mcmc` - 150 000 (im Initializer schon, im Preset bestätigt) - 5e-5 (V467/V536 optimal) - 0.005 → 0.01 (V535 optimal) - 0 → 160 000 (80 % von maxIter, V497c/V502) - 3.0 (im Initializer schon) - true (im Initializer schon) - 3 000 → 200 000 (effektiv aus, MCMC nutzt Reloc statt Reset)

PROSTYMI SŁOWAMI

Web-Auslieferung, Object-Captures mit Detail-Anspruch, Drohenflüge (auch wenn dann P9 Outdoor noch besser ist). 71 % weniger Gaussians als Classic bei vergleichbarer L1.

TP5 `.fullMCMCMip`**SZCZEGÓŁY**

Q1.5-D-Test-Sibling von TP4, mit Mip-Splatting + Paper-Magnitude-MCMC-Budget. Overrides gegenüber TP4:

- `mcmcMaxGaussians` 150 000 → 1 500 000 (10x, Paper-Magnitude)
- `useMipSplatting` false → true (Mip-on)

PROSTYMI SŁOWAMI

Alle anderen Felder identisch zu TP4. Q1.5 D-PASS auf Bicycle 2026-05-24 (bricht 12-Iter-Multi-Scale-FAIL-Streak). Q1.5-Endurteil 2026-05-25 trotzdem closed-no-win — Mip-Splatting-Gewinn nicht reproduzierbar über 11 Pair-Scenes. Preset bleibt opt-in.

TP6 `.classicBalanced`**SZCZEGÓŁY**

Mid-Tier Classic. Overrides gegenüber TP2: - 35 000 → 20 000 (V149: 20K = 30K bei 33 % weniger Zeit) - 20 000 → 0 (Cosine läuft auf maxIter = 20K, kein extended phase)

PROSTYMI SŁOWAMI

Standardfälle mit kürzerer Wartezeit. V149 als sweet spot identifiziert.

TP7 `.mcmcPreview` **SZCZEGÓŁY**

MCMC-Diagnose. Overrides gegenüber TP4: - 200 000 → 60 000 (V494b) - 160 000 → 48 000 (80 %) - 150 000 → 100 000 (V473b) - 160 000 → 40 000 (V494b) - 3.0 → 2.0 (1.4.5: Preview = lighter scaling)

 **PROSTYMI SŁOWAMI**

schnell ein MCMC-Resultat sehen, um zu beurteilen, ob TP4 oder ein Scene-Class-Preset lohnt.

TP8 `.mcmcBalanced` **SZCZEGÓŁY**

Mid-Tier MCMC. Overrides gegenüber TP4: - 200 000 → 120 000 (V518) - 160 000 → 96 000 (80 %) - 160 000 → 96 000 (80 %) - 3.0 → 2.5 (zwischen Preview 2.0 und Full 3.0)

 **PROSTYMI SŁOWAMI**

MCMC ohne den vollen 200K-Lauf. ~120 K Iterationen sind ein guter Kompromiss aus Qualität und Wartezeit.

TP9 `.quickTest` **SZCZEGÓŁY**


Reiner Funktionstest. Overrides gegenüber Initializer: - 30 000 → 1 000 - 15 000 → 500 - 2e-6 → 4e-6 (kalibriert für 0.25× Auflösung) - 100 → 50 - 3 000 → 100 000 (aus, da viel zu kurz) - 1.0 → 0.25

 **PROSTYMI SŁOWAMI**

Sanity-Check „startet das Training überhaupt sinnvoll?“. Dauer < 30 s auf M3 Ultra. Sieht garantiert matschig aus.

Methode:


Signature: `public func resolveMcmcMaxGaussians(initialPointCount: Int, bufferCapacity: Int) -> Int` **Defined in:**

 Einzige Source-of-Truth für die Frage „wie viele Gaussians darf MCMC maximal wachsen lassen?“. Berechnet sich aus drei Eingaben: dem konfigurierten `T62 mcmcMaxGaussians` (mit Mass-Extinction-Floor 150 000, falls 0), dem (Anzahl der SfM-Init-Punkte) und der (vorallozierte Gaussian-Buffer-Größe). Logik:

1. `base = T62 > 0 ? T62: 150_000` (der Mass-Extinction-Floor schützt gegen Initializer-Default-Bugs wie den 1.4.3-Mass-Extinction-Vorfall)
2. Falls `T73 mcmcAutoScaleByScene && initialPointCount > 0 && T72 mcmcCapMultiplier > 0`:
 - `scaled = max(base, ceil(initialPointCount × T72))` sonst
3. Falls `bufferCapacity > 0`: `return min(scaled, bufferCapacity)`

4. Sonst `return scaled`

Beispiel: Bicycle (Mip-NeRF 360, 194 Foto-Frames) → SfM-init ~156 K Punkte, `T62 = 150 000`, `T72 = 5.32` „ Buffer-Kapazität 8 M. Resolved Cap = $\min(8M, \max(150K, \text{ceil}(156K \times 5.32))) = \min(8M, 830K) = 830 K$. Das ist der effektive Wachstums-Cap, an den die MCMC-Relocation-Logik sich hält.

 Berechnet die wirkliche maximale Splat-Anzahl bei MCMC. Nimmt deine Einstellung, schaut sich an, wie viele Punkte deine Szene anfangs hat, und skaliert mit dem `Multipliiert`, falls die automatische Anpassung an ist. So passt sich der Cap an die Szene an, statt für eine kleine und eine riesige Szene den gleichen Wert zu erzwingen. Du musst die Methode nicht selbst aufrufen — das Training nutzt sie intern.

Welches Feld wofür? (Cheat-Sheet)

Ziel	Felder zum Drehen
Mehr Detail in der Ferne	<code>T62 mcmcMaxGaussians</code> hoch, <code>T72 mcmcCapMultipliiert</code> 5+
Mehr Detail allgemein (Classic)	<code>T1 maxIterations</code> hoch ($\leq 40K$), <code>T2 densifyUntilIteration</code> $\leq 14\%$ von T1
Floater in Drohnenflügen reduzieren	<code>T43 frustumCullEnabled</code> an, <code>T20 skyMaskingEnabled</code> an, <code>T45 skyDomeEnabled</code> an
Schöner Himmel in Außen-Szenen	<code>T45 skyDomeEnabled</code> an, <code>T47 skyDomeRadiusMultipliiert</code> 30–60
Kleinere Export-Datei	Strategie <code>.mcmc</code> (T61), <code>T56 postTrainingCompactification</code> an, <code>T62 mcmcMaxGaussians</code> $\leq 200K$
Schnelleres Training	<code>T22 trainingRenderScale</code> 0.5, <code>T1 maxIterations</code> halbieren — aber nicht beides!
Bessere Glanzlichter	<code>T21 shDegreeUpgradeIterations</code> mit [2K, 5K, 8K] (kein early-front-load), MCMC + 200K iter
Mac responsiv halten	<code>T25 throttleDelayMs</code> 5–10 (kostet ~15 % Trainingszeit)
Live-Vorschau häufiger	<code>T59 livePreviewInterval</code> runter auf 10–20
Sanftere Übergänge an Schatten	<code>T17 ssimWeight</code> etwas hoch (0.15–0.25), aber nicht über 0.3
Innenräume kompakt halten	<code>P10 Indoor-Preset</code> (, <code>T72 = 1.76</code>)

Gefährliche Felder

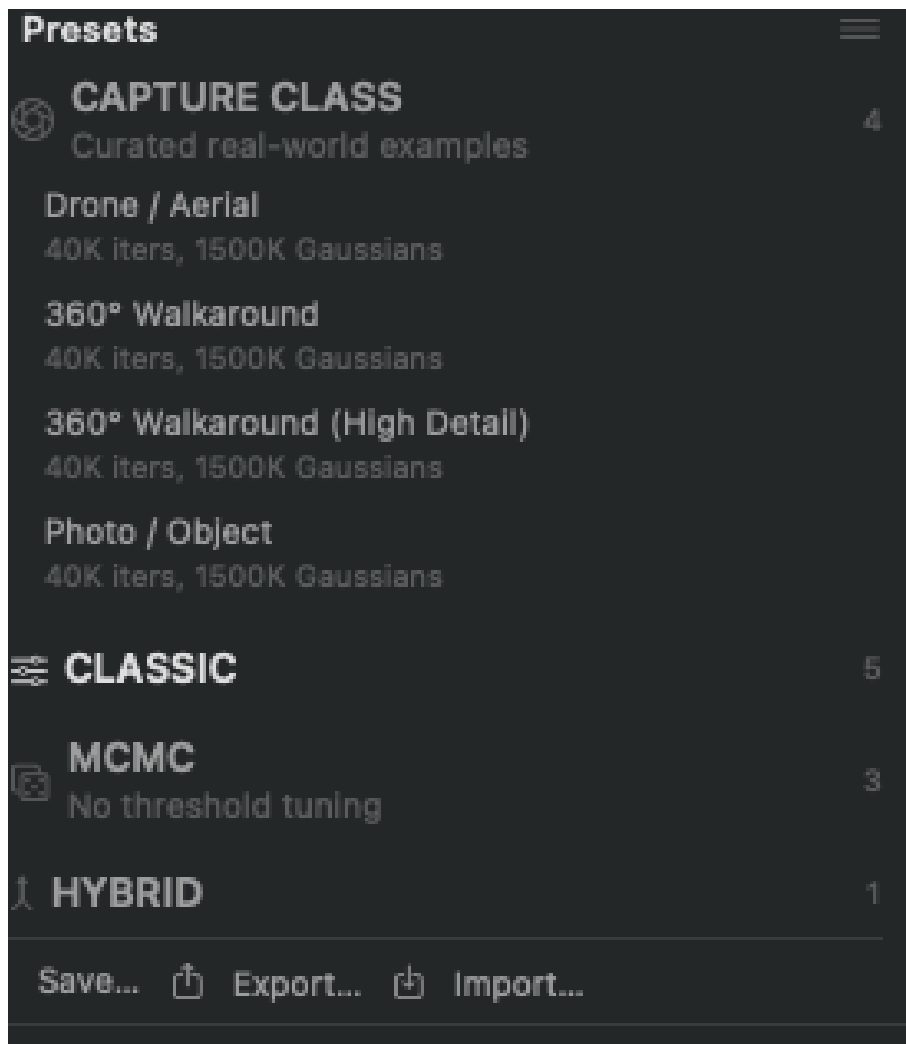
Diese Felder können bei Falsch-Konfiguration zu OOM, App-Crash, Mass-Extinction der Gaussians oder unbrauchbaren Benchmark-Daten führen. Mit Vorsicht zu behandeln:

- T11 `densifyGradThreshold` — eine Halbierung kann 2–4× so viele Gaussians erzeugen, was schnell den GPU-Speicher sprengt. Auch zu beachten: muss zur T22 `trainingRenderScale` passen (1.0× → 1e-6, 0.5× → 2e-6, 0.25× → 4e-6).
- T72 `mcmcCapMultiplier` — bei großen Szenen mit > 200 K SfM-init-Punkten und Multiplier > 5 entsteht ein Resolved-Cap von Millionen Gaussians. Auf 36-GB-RAM-Macs OOM möglich. Outdoor-Preset 5.32 funktioniert nur, weil Mip-NeRF-360-Bicycle 156 K init-Punkte hat → 830 K Cap.
- T39 `testViewIndices` — manuelles Setzen kann den Benchmark unbrauchbar machen (alle Indices > N → keine Holdouts). Lass das `--benchmark`-Flag das setzen.
- T64 `mcmcOpacityRegWeight` **und** T65 `mcmcScaleRegWeight` — In 1.4.3-Beta auf 0.01 gesetzt, was zu Mass-Extinction führte (460 K → 5 Gaussians in einer Iteration). Seit 1.4.4 auf 0.0 fixiert, aber manuelles Erhöhen kann das Problem reproduzieren.
- T15 `opacityResetInterval` — wenn nicht 100 000+ (effektiv aus) und das Training kürzer als 10 000 Iterationen ist, zerstört der Reset die Konvergenz. `.preview` hat es deshalb auf 100 000 trotz `maxIterations = 5 000`.
- T54/T55 `densifyPhase2*` — Two-Phase Densification ist in Tests in 0-Gaussians-Cascade abgebrochen. Lass beide auf 0.
- T74 `useMipSplatting` — Q1.5 closed-no-win 2026-05-25, kann auf manchen Outdoor-Szenen sogar PSNR verschlechtern. Default off, opt-in nur für Forschung.

Wenn ein Feld auf dieser Liste steht und du es ändern willst, mache vorher eine Sicherung deines aktuellen Presets (Export als JSON) und überlege, ob du das Resultat reproduzierbar messen kannst — sonst weißt du hinterher nicht, ob du eine Verbesserung oder Verschlechterung herbeigeführt hast.

ROZDZIAŁ


Rozdział 7 — Wbudowane ustawienia wstępne jakości



Rysunek 27: Sekcja Ustawienia wstępne ze wszystkimi czterema rozwiniętymi grupami — CAPTURE CLASS (4 ustawienia: Drone/Aerial, 360° Walkaround, 360° Walkaround (High Detail), Photo/Object), CLASSIC (5 ustawień: Quick/Preview/Balanced/Quality/Ultra Detail), MCMC (3 ustawienia, wskazówka „No threshold tuning”), HYBRID (1 ustawienie: Balanced (Hybrid))

CO WIDAĆ NA OBRAZIE Sekcja Ustawienia wstępne w Inspectorze, wszystkie cztery grupy rozwinięte. CAPTURE CLASS z czterema kuratorowanymi ustawieniami dla rzeczywistych ujęć (Drone / Aerial, 360° Walkaround, 360° Walkaround (High Detail), Photo /

Object) — to grupa podstawowa i w Trybie początkującym jedyna widoczna. CLASSIC z Quick (1K iter.), Preview (5K iter., aktywny wybór z niebieskim ptaszkiem), Balanced (20K iter.), Quality (35K iter.) i Ultra Detail (35K iter.). MCMC z podtytułem „No threshold tuning„ — MCMC nie potrzebuje progu Densify-Until: Preview (60K iter., 100K gaussianów), Balanced (120K, 150K), Quality (200K, 150K). HYBRID z ustawieniem Balanced (Hybrid) (20K iter., 150K gaussianów). Wiersz akcji w stopce: Save..., Export..., Import...

Ustawienie wstępne to przygotowana konfiguracja treningu. RadianceKit dostarcza trzynaście wbudowanych ustawień wstępnych w czterech grupach: cztery ustawienia **Capture-Class** (P9–P12) — kuratorowane, wzrokowo zwalidowane na prawdziwym materiale społeczności receptury dla rzeczywistych rodzajów ujęć (dron, obejście 360°, foto-obiekt), i oś podstawowa od v1.6 —, pięć ustawień Classic (P1–P5: Quick/Preview/Balanced/Quality/Ultra Detail), trzy ustawienia MCMC (P6–P8) oraz jedno ustawienie hybrydowe (P13), które łączy strategie Classic i MCMC. Wcześniejsze ustawienia „Scene-Class„ (Render/3D, Outdoor, Indoor, w Fazie Q7 akademicko dostrojone względem scen Mip-NeRF-360 i NeRF-Blender) zostały w v1.6 wycofane jako widoczna grupa — wzrokowo zwalidowana na prawdziwym materiale Capture-Class jest teraz osią podstawową; konfiguracje dostrojone w Q7 pozostają zachowane tylko wewnętrznie. Wybierasz ustawienia na pasku bocznym w obszarze **Presets** lub w Trybie początkującym przy imporcie. Przyciski  otwierają okna dialogowe, aby dodać własne ustawienia obok — trzynastu wbudowanych nie da się usunąć, ale można je zduplikować.

W Expert View ustawienia wstępne pojawiają się pogrupowane według rodzaju ujęcia i strategii (Capture Class / Classic / MCMC / Hybrydowy). Kliknięcie wpisu zapisuje przypisaną do niego konfigurację treningu do bieżącego stanu. To nie jest zrzut — gdy potem przekręcisz suwakami, stan się zmieni, ale samo ustawienie pozostanie niezmiennione; kolorowa wskazówka pokazuje wtedy „modified„.

Które ustawienie wstępne kiedy jest właściwe, zależy przede wszystkim od typu sceny i sprzętu. Trzy zestawienia tabelaryczne na końcu rozdziału to podsumowują.

| P1 — Quick

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Classic„ → pozycja „Quick“. Końcówka UUID ...001 .

TECHNICZNIE

Ustawienie diagnostyczne z 1 000 iteracjami, klasyczną (adaptacyjną) strategią densyfikacji i skalowaniem rozdzielczości treningu 0.25× (obraz wejściowy jest przed treningiem zmniejszany do 25%). Nie służy do dostarczania finalnej sceny, lecz do szybkiego stwierdzenia, czy konfiguracja (pozy kamer, chmura punktów, seria zdjęć) w ogóle wykazuje sensowny ruch w wartościach loss. Na M3 Ultra typowo poniżej 30 sekund na 50–200 obrazach. Mała rozdzielczość zaciemnia rzeczywistą jakość obrazu, ale utrzymuje zapotrzebowanie na pamięć i koszty renderingu na bardzo niskim poziomie. Wybierane także automatycznie jako domyślne przy pierwszym uruchomieniu, gdy system ma mniej niż 10 GB RAM.

PROSTYMI SŁOWAMI

Szybki test funkcjonalny. Wrzucasz obrazy, czekasz ledwie pół minuty, sprawdzasz czy pojawia się surowy zarys sceny. Jeśli obraz w podglądzie wygląda jak rozmyta plama — pasuje, tak ma być. Jeśli natomiast widzisz tylko ciemne punkty albo totalnie zniekształconą formę, prawdopodobnie pozy kamer są błędne (patrz rozdział 9). Aby uzyskać prezentowalny wynik, potrzebujesz potem co najmniej P2 lub P3.

| P2 — Preview (Classic)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Classic„ → pozycja „Preview“. Końcówka UUID ...002 .

TECHNICZNIE

5 000 iteracji densyfikacji Classic, skalowanie rozdzielczości 0.5×, podwójne współczynniki uczenia względem standardu. Densyfikacja (klonowanie + dzielenie) jest aktywna przez pierwsze 2 500 iteracji, później tylko pruning. Domyślne ustawienie dla systemów z ≥ 10 GB RAM. Na M3 Ultra typowo 90 sekund do 3 minut dla sceny z 200 obrazami. Daje użyteczne wrażenie geometrii i pozy kamery, ale tekstury są wyraźnie rozmyte — rozdzielczości renderingu 0.5× nie da się potem ominąć przez ponowny trening z P3 lub P4, ponieważ współczynniki uczenia są skalibrowane pod połowiczną rozdzielczość.

PROSTYMI SŁOWAMI

Standard dla „rzutu okiem„. Jeśli właśnie zaimportowałeś nowe obrazy i chcesz zobaczyć, czy scena w ogóle daje się zrekonstruować, to jest właściwy poziom. Około 2–3 minut czekania, potem możesz obracać scenę w widoku 3D i ocenić, czy inwestycja w kolejne treningi ma sens. Dopiero gdy wynik podglądu wygląda już dobrze, optaca się przejść do Balanced lub Quality.

| P3 — Balanced (Classic)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Classic„ → pozycja „Balanced“. Końcówka UUID `...005`.

TECHNICZNI

20 000 iteracji densyfikacji Classic przy pełnej rozdzielczości obrazu. Densyfikacja działa przez pierwsze 15 000 iteracji, od iter 3 000 z interwałem densyfikacji 100. Empirycznie „sweet spot„ z udokumentowanych sesji treningowych: przy klasycznej densyfikacji na Horse Full i Truck loss L1 stabilizuje się między iter 18 000 a 22 000, dłuższy trening nie przynosi istotnej poprawy poniżej Quality (P4). Na M3 Ultra typowo 30–60 sekund na 200 obrazach, 5–8 minut na 1 000+ obrazach.

PROSTYMI SŁOWAMI

„Dobry kompromis„. Większość scen wygląda już z tym dobrze, bez konieczności czekania godzin. Jeśli chcesz pokazać wynik końcowy gdziekolwiek (social media, strona internetowa, demo dla klienta), to często wystarczy. Dopiero gdy chcesz wczytać się w model splot lub potrzebujesz szczegółów tekstury powierzchni, opłaca się skok do P4 Quality lub P7 MCMC.

| P4 — Quality (Classic)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Classic„ → pozycja „Quality“. Końcówka UUID `...003`.

TECHNICZNI

35 000 iteracji densyfikacji Classic z V546- „Opacity Decay„ (HTGS, Eurographics 2025): po każdym cyklu densyfikacji opacyty wszystkich istniejących gaussianów mnoży się przez współczynnik < 1.0 , co niezawodnie usuwa nieaktywne gaussiany przy pruningu i tym samym na identycznej liczbie iter osiąga 14% lepszy loss L1 niż klasyczny przebieg 35 000. Loss SSIM jest wyłączony (`ssimWeight=0.05`). Na M3 Ultra typowo 2–4 minuty na 200 obrazach. Dostarcza na NeRF-Blender (Lego, Chair, Drums) finalne L1 ≈ 0.023 — najlepszy wariant Classic w 560+ udokumentowanych eksperymentach. Uwaga: potrzebuje ~3–5 GB pamięci GPU; na systemach 8-GB bezpieczniejszym wyborem jest P3.

PROSTYMI SŁOWAMI

Najlepszy wariant klasyczny. Daje ostrą teksturę i drobną geometrię, zwłaszcza w ujęciach obiektów (rzeźba, krzesło, waza). Przy dużych scenach plenerowych lub pomieszczeniach prawie nie widzisz różnicy względem Balanced — tam zmiana na ustawienie MCMC (P6–P8) lub Capture-Class (P9–P12) opłaca się bardziej niż skok z P3 na P4. Kto chce absolutnego maksimum rodziny Classic, bierze P5 Ultra Detail.

I P5 — Ultra Detail (Classic)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Classic„ → pozycja „Ultra Detail“. Końcówka UUID `...008`.

TECHNICZNIE

Około 35 000 iteracji densyfikacji Classic — zwycięzca przebiegu held-out matrycy jakości (2026-06-10). Na wszystkich trzech testowanych scenach Mip-NeRF-360 Ultra Detail bije wbudowane ustawienie MCMC „Quality„ (P8) przy porównywalnym czasie zegarowym o około +0.94 dB PSNR. Tym samym jest najmocniejszym ustawieniem jakościowym grupy Classic i najostrzejszym wariantem Classic, jaki dostarcza RadianceKit. Na M3 Ultra typowo w tych samych ramach czasowych co P4 Quality (2–5 minut na 200 obrazach), potrzebuje jednak nieco więcej pamięci GPU; na systemach 8-GB bezpieczniejszym wyborem pozostaje P3.

PROSTYMI SŁOWAMI

Najostrzejszy poziom Classic i zwycięzca held-out naszych testów jakości: na rzeczywistych scenach około jeden decybel lepiej niż wariant MCMC „Quality„ — przy podobnym czasie oczekiwania. Jeśli chcesz maksymalnej wierności detalu przy sprawdzonej klasycznej densyfikacji i masz wystarczająco pamięci GPU, to pierwszy wybór. Jeśli pamięci nie starcza lub potrzebujesz możliwie małego pliku eksportowego, zostań przy P4 Quality lub ustawieniu MCMC.

I P6 — Preview (MCMC)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „MCMC„ → pozycja „Preview“. Końcówka UUID `...006`.

TECHNICZNIE

60 000 iteracji densyfikacji MCMC (3DGS-MCMC, NeurIPS 2024) przy limicie 100 000 gaussianów. MCMC zastępuje heurystyczną logikę klonowania/dzielenia relokacją Markov Chain Monte Carlo: martwe gaussiany są na nowo rozmieszczane przez głębokości próbkowania ważone sigmoidą, co daje kontrolowaną i powtarzalną liczbę gaussianów. Limit twardo ogranicza maksymalną liczbę przy 100K — to oszczędza pamięć i czas renderingu, ale kosztuje detal. Na M3 Ultra typowo 5–8 minut na 200 obrazach. Nadaje się jako „test funkcjonalny MCMC„ — pomaga ocenić, czy zmiana z Classic na MCMC ma sens, zanim zainwestujesz więcej czasu w P7 lub P8.

PROSTYMI SŁOWAMI

Jak P2 Preview, ale z nowszą procedurą MCMC. Często daje nieco bardziej zwarte, równomiernie rozłożone chmury splat niż wariant Classic. Do pierwszej oceny sceny wystarczają zwykle 5–8 minut. Jeśli wynik podglądu ci się podoba, następnym krokiem jest P7 (Balanced) lub od razu P8 (Quality MCMC).

I P7 — Balanced (MCMC)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „MCMC„ → pozycja „Balanced“. Końcówka UUID ...007 .

TECHNICZNI

120 000 iteracji MCMC przy limicie 150 000 gaussianów. Średni poziom MCMC — prawie finalna liczba gaussianów z P8 Quality, ale tylko 60% iteracji. Empirycznie loss L1 w udokumentowanych sesjach treningowych wynosi 0.026–0.028 na Horse Full, w porównaniu do P8 z 0.0246 — czyli około 7% wyżej, za to połowiczny czas oczekiwania. Na M3 Ultra typowo 8–15 minut na 200 obrazach. Używa procedury, która skaluje efektywny limit gaussianów do gęstości punktów wejściowej chmury SfM (zobacz T75 w rozdziale 6).

PROSTYMI SŁOWAMI

MCMC z porządną głębią szczegółu, ale bez długiego pełnego przebiegu P8. Dla większości scen to wystarcza, zwłaszcza jeśli chcesz wcisnąć przebieg MCMC w ramy czasowe przerwy obiadowej. Jeśli pamięci jest mało (np. na procesorach M tylko 16 GB), zostań tutaj — P8 potrzebuje więcej pamięci GPU.

I P8 — Quality (MCMC)

GDZIE

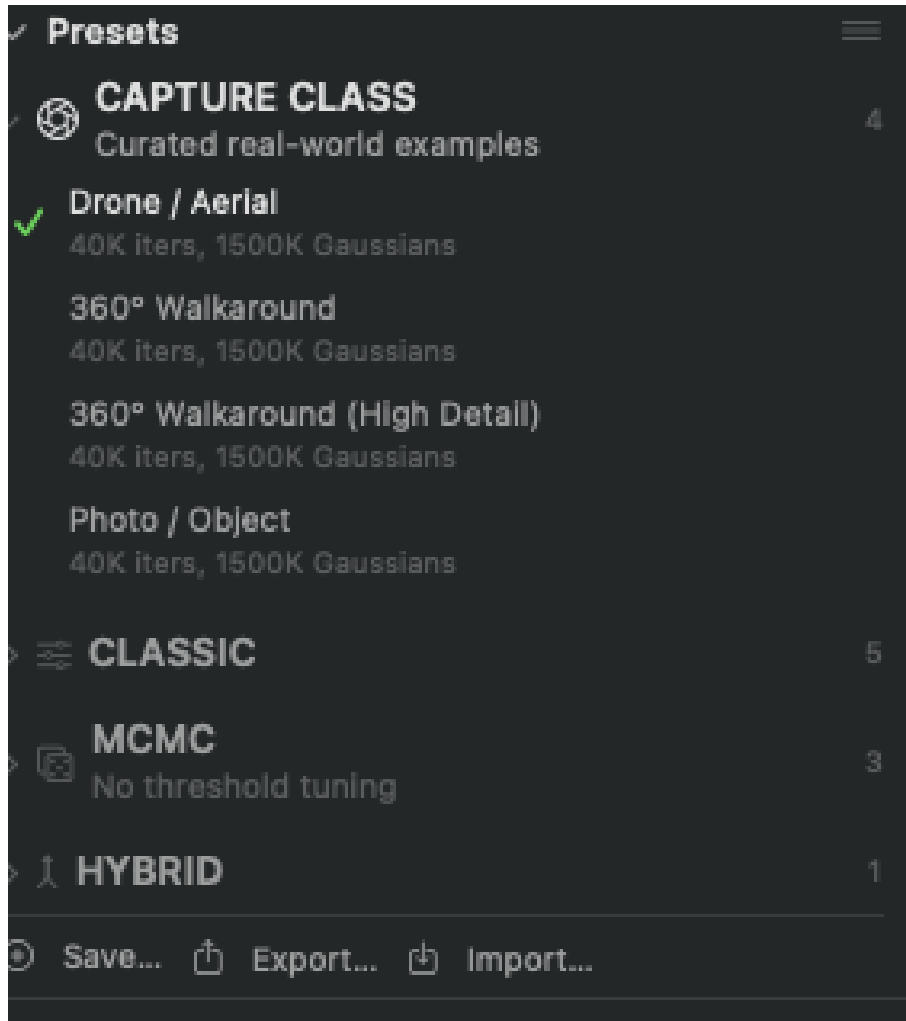
Inspector → sekcja Ustawienia wstępne → grupa „MCMC„ → pozycja „Quality“. Końcówka UUID ...004 .

TECHNICZNI

200 000 iteracji MCMC przy limicie 150 000 gaussianów, loss SSIM 0.05, MCMC-Noise-Decay przez 80% iteracji. Najlepszy pojedynczy przebieg L1 w 560+ eksperymentach: 0.0238 na Horse Full, uśrednione przez 3 próby 0.0246 (w porównaniu do P4 0.0230 na tej samej scenie). MCMC dostarcza przy tym 71% mniej gaussianów (150K vs ~524K) — decydujące, gdy chcesz dostarczyć wynik w internecie, ponieważ mniejsza chmura daje znacząco mniejsze pliki eksportowe. Czas treningu na M3 Ultra typowo 20–35 minut na 200 obrazach; na zestawach 1 000+ raczej 1–2 godziny. Najlepszy wybór, gdy pożądana jest maksymalna jakość obrazu przy minimalnym rozmiarze końcowym.

PROSTYMI SŁOWAMI

Najlepszy wariant MCMC. Daje bardzo czyste, zwarte chmury splat — idealne, gdy chcesz potem osadzić wynik jako webowy widok 3D lub przesać jako plik (plik jest mniejszy niż przy P4 Quality przy porównywalnej jakości optycznej). Potrzebujesz jednak do tego cierpliwości — na dużych ujęciach ponad godzina oczekiwania. Zaplanuj to raczej jako przebieg „przez noc„.



Rysunek 28: Grupa CAPTURE CLASS rozwinięta ze wszystkimi czterema kuratorowanymi ustawieniami dla rzeczywistych ujęć — Drone / Aerial (MCMC, 40K iter., limit 1,5 M), 360° Walkaround (MCMC, 40K, limit 1,5 M), 360° Walkaround (High Detail) (Hybrydowy, 40K, limit 1,5 M, opcjonalny) i Photo / Object (Hybrydowy, 40K, limit 1,5 M). Ta grupa stoi na samej górze i w Trybie początkującym jest jedyną widoczną.

CO WIDAĆ NA OBRAZIE Inspector z rozwiniętą grupą CAPTURE CLASS — grupą ustawień podstawowych od v1.6, w Trybie początkującym jedyną wyświetlaną. Każdy wpis to wzrokowo zwalidowana na prawdziwym materiale społeczności receptura dla konkretnego rodzaju ujęcia (dron, obejście 360°, foto-obiekt), a nie wartość zoptymalizowana względem akademickiego zestawu testowego. Wybór kliknięciem zapisuje przypisaną konfigurację treningu do bieżącego stanu.

| P9 — Drone / Aerial

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Capture Class„ → pozycja „Drone / Aerial“. Końcówka UUID ...010 .

TECHNICZNI

Ustawienie Capture-Class dla orbit z powietrza i z drona wokół budynków i krajobrazów. Densyfikator MCMC, 40 000 iteracji, limit 1,5 mln gaussianów, loss SSIM 0.5 plus człon Edge-Aware 0.1. Decydująca jest kara za anizotropię z wagą 0.003 przy progu proporcji 6 — „zabójca spaghetti„ przeciwko typowym igłowym artefaktom, które generuje materiał z drona. Zwalidowane na rzeczywistym locie drona DJI 4K nad wiaduktem Pensford (sprawdzone wzrokowo, nie tylko metrycznie).

PROSTYMI SŁOWAMI

Dla ujęć z powietrza — loty drona wokół budynku, nad krajobrazem, wzdłuż fasady. Mocna kara za anizotropię uprzęta igłowe lub spaghetti-podobne artefakty, które materiał z drona chętnie produkuje. Jeśli twój materiał jest nakręcony z ziemi, lepiej pasuje Photo / Object lub ustawienie Classic.

| P10 — 360° Walkaround

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Capture Class„ → pozycja „360° Walkaround“. Końcówka UUID ...011 .

TECHNICZNI

Ustawienie Capture-Class dla filmów typu walkaround 360°. Densyfikator MCMC, 40 000 iteracji, limit 1,5 mln gaussianów, loss SSIM 0.5 plus człon Edge-Aware 0.1, łagodna kara za anizotropię (waga 0.001 przy progu proporcji 15). Maski osób i nieba jest aktywna. Ustawienie oczekuje filmu 360° equirect, który jest wewnętrznie reprojekowany na perspektywiczne kadry o szerokości około 90°, zanim ruszy trening. Zwalidowane na obejściach 360° w 8K z kijem do selfie (scena Monument, sprawdzone wzrokowo).

PROSTYMI SŁOWAMI

Dla filmów typu obejście 360° — przechodzisz z kamerą 360° lub kijem do selfie przez pomieszczenie albo wokół obiektu. RadianceKit sam rozkłada panoramę kulistą na normalne kąty widzenia i maskuje przechodniów oraz niebo. Aby uzyskać maksymalną ostrość na tym samym materiale, wypróbuj dodatkowo wariant High Detail (P11).

| P11 — 360° Walkaround (High Detail)

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Capture Class„ → pozycja „360° Walkaround (High Detail)“. Końcówka UUID ...013 (opcjonalny).

TECHNICZNI

Opcjonalne ustawienie Capture-Class dla filmów typu walkaround 360° z maksymalnym detalem. Densyfikator hybrydowy (klasyczne clone/split na bazie gradientu absolutnego + szum MCMC + relokacja), 40 000 iteracji, limit 1,5 mln gaussianów, kara za anizotropię 0.0015 przy progu proporcji 15, loss SSIM 0.2 i człon Edge-Aware 0 — zablokowana receptura screen-split „r50“. Na materiale 360° bije standardowe ustawienie MCMC „360° Walkaround„ (P10) pod względem PSNR, LPIPS i widocznego konfetti, i to przy około jednej trzeciej liczby splatów. Stoi celowo opcjonalnie *obok* standardowego ustawienia 360°, dopóki nie zostanie zwalidowane na większej liczbie scen.

PROSTYMI SŁOWAMI

Ostrzejsza alternatywa dla standardowego ustawienia 360° (P10): więcej detalu, mniej konfetti, znacznie mniejszy plik. Stoi celowo obok zamiast je zastępować — jak dotąd potwierdzone na garstce scen. Jeśli twoje obejście 360° jest czysto nakręcone, wypróbuj najpierw to ustawienie i porównaj wynik z P10.

| P12 — Photo / Object

GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Capture Class„ → pozycja „Photo / Object“. Końcówka UUID ...012 .

TECHNICZNI

Ustawienie Capture-Class dla orbit obiektów z ostrych pojedynczych zdjęć (bez wideo). Densyfikator Hybrid-t1 (z relokacją), 40 000 iteracji, limit 1,5 mln gaussianów, loss SSIM 0.5 plus człon Edge-Aware 0.1, łagodna kara za anizotropię (waga 0.001 przy progu proporcji 15), Opacity Decay 0.9995 co 50 iteracji, **bez** maskowania. Zwalidowane na 163 wysokorozdzielczych zdjęciach 41-MP szkieletu (sprawdzone wzrokowo). Niewiele widoków (do około 600) pozostaje przy tym poniżej progu Hybrid-Collapse.

PROSTYMI SŁOWAMI

Dla ujęć obiektów z ostrych pojedynczych zdjęć — okrążasz rzeźbę, model, produkt z kamerą i robisz zdjęcia zamiast wideo. Bez maskowania, bo ostre zdjęcia mają zwykle czyste tło. Dla źródeł wideo weź zamiast tego ustawienie 360° lub Drone.

I P13 — Zrównoważony (Hybrydowy)



GDZIE

Inspector → sekcja Ustawienia wstępne → grupa „Hybrydowy„ → pozycja „Zrównoważony (Hybrydowy)“. Końcówka UUID `...009`.



TECHNICZNI

20 000 iteracji z hybrydową strategią densyfikacji: klasyczne, sterowane gradientem clone/split lokuje pojemność tam, gdzie potrzebuje jej loss, szum SGLD z MCMC dalej eksploruje, a martwe gaussiany są relokowane, zamiast przepadać w pruningu. Opacity Decay (V546) zastępuje resety opacity; kara za anizotropię (waga 0.001, próg proporcji 15) trzyma w ryzach igłowate splaty. Górny limit gaussianów skaluje się ze sceną (baza 150K, scene-aware $\times 3.0$). Zwalidowane na pięciu scenach względem czystego MCMC przy tym samym budżecie: średnio +0.45 dB PSNR przy 20–30% mniejszej liczbie gaussianów (stonehenge +1.23, family +0.82, garden +0.47 dB). Na M3 Ultra typowo 5–10 minut na 200 obrazach.



PROSTYMI SŁOWAMI

Mocny pierwszy wybór dla finalnego wyniku: ostrzejszy detal niż ustawienia MCMC przy podobnie zwartym pliku, w ułamku czasu treningu P8. Jeśli masz czas tylko na jeden przebieg jakościowy i żadna z klas ujęcia wyraźnie nie pasuje, zacznij tutaj. Ustawienia Classic pozostają lepsze do szybkich testów, a ustawienia Capture-Class (P9–P12) są pierwszym wyborem, gdy twoja scena wyraźnie pasuje do jednego z tych rodzajów ujęcia.

Kiedy które ustawienie wstępne?

Scenariusz	Test wstępny	Główny przebieg
Test funkcjonalny nowych obrazów, < 30s	P1 Quick	—
Orbit obiektu z ostrych pojedynczych zdjęć	P2 Preview	P12 Photo / Object
Skan pojedynczego obiektu (wideo), < 500 zdjęć	P2 Preview	P4 Quality lub P8 Quality MCMC
Film typu walkaround 360°	P6 Preview MCMC	P10 360° Walkaround (ostro: P11 High Detail)
Orbit z powietrza / drona, krajobraz	P6 Preview MCMC	P9 Drone / Aerial
Dostawa do sieci (małe, zwarte)	P2	P8 Quality MCMC (najmniejszy plik przy pełnej jakości)
Ostry detal w krótkim czasie, zwarty eksport	P2 lub P6	P13 Zrównoważony (Hybrydowy)
Maksymalna wierność detalu, strategia Classic	P3 lub P6	P5 Ultra Detail
Druk, marketing, pełny detal	P3 lub P6	P4 Quality (Classic) lub P5 Ultra Detail

Szybkie porównanie

Usta- wie- nie	Strat- egia	Iter.	Maks. Gs	Skala renderu	Typowy czas (200 obr., M3 Ultra)	Q-Swe- ep
P1 Qu- ick	Classic	1 000	∞	0.25x	~30 s	—
P2 Preview	Classic	5 000	∞	0.5x	2–3 min	—
P3 Ba- lan- ced	Classic	20 000	∞	1.0x	30–60 s	—
P4 Qu- ality	Classic	35 000	∞	1.0x	2–4 min	V546 HTGS
P5 UI- tra De- tail	Classic	~35 000	∞	1.0x	2–5 min	Ma- trix Δ+0.94 dB
P6 Preview MCMC	MCMC	60 000	100 K	1.0x	5–8 min	—
P7 Ba- lan- ced MCMC	MCMC	120 000	150 K	1.0x	8–15 min	—
P8 Qu- ality MCMC	MCMC	200 000	150 K	1.0x	20–35 min	V544a
P9 Dro- ne/ Aerial	MCMC	40 000	1.5 M	1.0x	10–25 min	Wzrok / wia- dukt
P10 360° Wal- ka- ro- und	MCMC	40 000	1.5 M	1.0x	10–25 min	Wzrok / Mo- nu- ment
P11 360° Wal- ka- ro- und (High De- tail)	Hybry- dowy	40 000	1.5 M	1.0x	10–25 min	Wzrok (opt- -in)
P12 360° to/ Ob- jekt ny (Hy-	Hybry- dowy	20 000	1.5 M	1.0x	15–25 min	Wzrok / Delta 0.45 dB

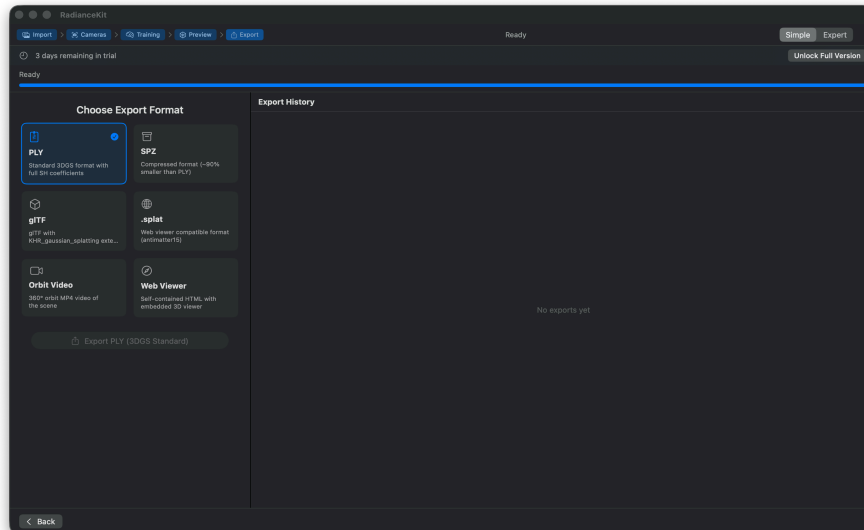
Własne ustawienia wstępne

Przyciskiem **Save...** w sekcji Ustawienia wstępne (l1 w rozdziale 2) zapisujesz bieżącą konfigurację treningu jako własne ustawienie wstępne. Własne ustawienia nie są „Built-in”, i można je zmieniać nazwę, eksportować (jako JSON), udostępniać przeciągnięciem, duplikować i usuwać. Trzydzieści wbudowanych ustawień P1–P13 pozostaje nietknięte przez przycisk usuwania.

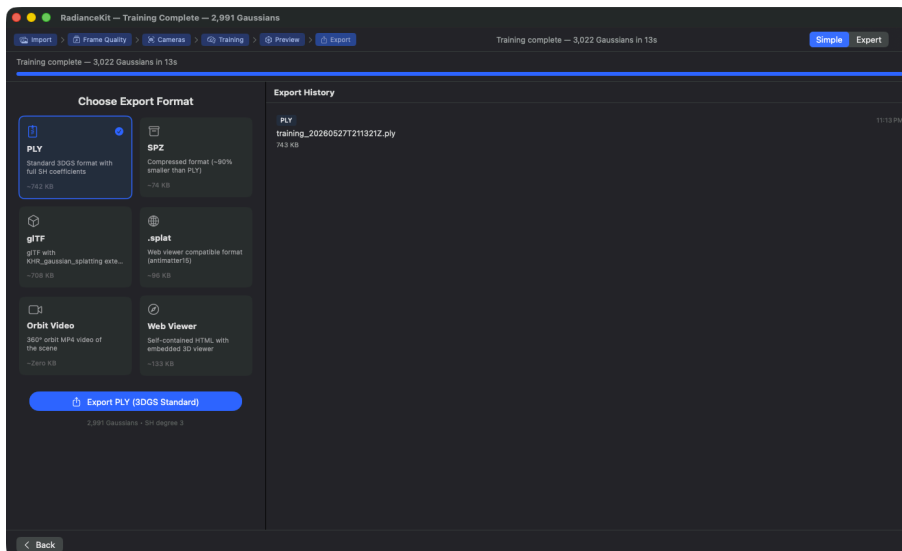
Reguła kciuka: Jeśli zmieniasz w jakimś ustawieniu coś, czego częściej będziesz potrzebować — Sky-Dome włączony, wyższa waga SSIM dla konkretnej klasy sceny, odmienne liczby iteracji — to zapisz ten wariant jako własne ustawienie. Tak będziesz wiedzieć przy następnym przebiegu, że to konfiguracja odbiegająca od standardu.

ROZDZIAŁ

Rozdział 8 — Formaty eksportu



Rysunek 29: Wybór formatu eksportu w Trybie początkującym — sześć kart formatów



Rysunek 30: Siatka formatów eksportu na żywo po treningu 5K iter na bukietcie flowers — wszystkie sześć kart z dynamicznym obliczaniem rozmiaru (PLY 742 KB wybrane, SPZ 74 KB, glTF 708 KB, .splat 96 KB, Orbit Video ~Zero KB, Web Viewer 133 KB), historia eksportu po prawej z już zapisanym PLY

Co widać na obrazie (2 991 gaussianów, SH degree 3, syntetyczny bukiet Bjoerna z Blendera jako zestaw testowy bez problemów IP): Wartości rozmiaru pod każdą

kartą formatu są obliczane na żywo z bieżącej liczby gaussianów i narzutu formatu — nie zakodowane sztywno. Z 2 991 gaussianów (SH degree 3) powstaje 742 KB PLY, 74 KB SPZ (współczynnik $\sim 10\times$ mniejszy dzięki kwantyzacji), 708 KB glTF (z rozszerzeniem `KHR_gaussian_splatting`, dlatego prawie równoważny PLY), 96 KB `.splat` (skompresowany format 24 bajty na gaussian). Orbit Video pokazuje „~Zero KB”, ponieważ rozmiar jest znany dopiero po kodowaniu MP4. Web Viewer (133 KB) pakuje samodzielny plik HTML z osadzonym viewerem WebGL i skompresowanymi danymi splat — większy niż samo `.splat` z powodu narzutu viewera. Historia eksportu po prawej listuje już zakończony eksport PLY („training_20260527T211321Z.ply, 743 KB, 23:13”) z pigułką formatu i akcją `Reveal-w-Finderze`.

Zakończony trening daje chmurę gaussianów — zbiór kilkuset tysięcy do milionów rozkładów Gaussa 3D, które razem rekonstruują scenę. RadianceKit zna dziesięć dróg zapisu tej chmury na dysk. Sześć z nich to czyste formaty danych 3D (PLY, Compressed PLY, SPZ, SOG, glTF, `.splat`), jeden pakuje chmurę wraz z gotowym viewerem HTML (Web Viewer), jeden renderuje plik MP4 z orbitalnej ścieżki kamery (Orbit Video), a dwa eksportują nie zawartość gaussianów, lecz tylko wynik SfM (pozy kamer i zgrubną chmurę punktów) do ponownego użycia w innych potokach treningu (`transforms.json` + workspace COLMAP).

Który format kiedy jest właściwy, zależy od celu. Do archiwizacji pełnych danych bez utraty jakości bierze się PLY. Do viewerów webowych na własnej stronie zwykle wystarcza `.splat` lub wbudowany Web Viewer. Gdy plik musi być minimalny, opłaca się SPZ lub SOG. Do ponownego użycia wyniku SfM w Nerfstudio, Postshot lub Brush właściwymi drogami są `transforms.json` i workspace COLMAP.

Wszystkie funkcje eksportu znajdują się w menu „Export”, oraz w Trybie początkującym na ostatnim etapie kreatora. Większość formatów jest w pełni zgodna z sandboxem i działa w wersji App Store. Tylko SOG wymaga zewnętrznego pliku wykonywalnego (`cwebp`), który w buildzie App Store niekoniecznie jest dostępny — szczegóły w E4.

I E1 — PLY (.ply)

GDZIE

Pasek menu → Export → 3D Formats → Export PLY... (⌘E). Tryb początkujący: krok kreatora Export → karta formatu „PLY„. **Rozmiar:** typowo 100% (wartość referencyjna). **Zgodny z:** SuperSplat, PolyCam, wszystkie viewery 3DGS.

TECHNICZNIE

PLY jest kanonicznym formatem zapisu dla 3D Gaussian Splatting. RadianceKit zapisuje binarny plik Little-Endian ze standardowym układem właściwości 3DGS: na każdego gaussiana trójkomponentowa pozycja, trzy normalne stałe ustawione na zero, trzy współczynniki SH DC (`f_dc_0..2`) dla bazowego koloru RGB, następnie do 45 dalszych współczynników SH (`f_rest_0..44`) w transponowanym układzie Channel-Major zdefiniowanym w artykule Kerbl 2023 (najpierw wszystkie współczynniki kanału R, potem wszystkie G, potem wszystkie B), za nimi opacità w postaci logit (surowe wartości pre-sigmoid), trzy skale w przestrzeni log oraz rotacja jako kwaternion wxyz. Maksymalny eksportowany stopień SH jest ograniczony do minimum z życzenia użytkownika i faktycznie wyuczonego stopnia; domyślnie 3 (45 reszt współczynników). Przed zapisem rozmiar payloadu jest obliczany w liczbach 64-bitowych, aby wychwycić przepełnienie przy ekstremalnie dużych chmurach. Plik zapisywany jest atomowo, co przy dużych chmurach przejściowo zajmuje podwójną ilość miejsca na dysku.

PROSTYMI SŁOWAMI

To „plik oryginalny„. Największy plik, największa zgodność, brak strat. Jeśli nie wiesz, który format wybrać, weź PLY — otwiera się w niemal każdym narzędziu 3DGS. Dla 1 miliona gaussianów to zależnie od stopnia SH między 200 a 800 MB. Jeśli plik jest zbyt duży, spójrz na E2 (skompresowane PLY) lub E3 (SPZ).

I E2 — Compressed PLY (.ply)

GDZIE

Pasek menu → Export → 3D Formats → Export Compressed PLY... Tryb początkujący: karta formatu „Compressed PLY„. **Rozmiar:** ok. 10–20% względem PLY (5- do 10-krotna kompresja). **Zgodny z:** SuperSplat, silnik PlayCanvas, viewery webowe.

TECHNICZNIE

Wariant formatu PLY od PlayCanvas z chunkowaną kwantyzacją. Gaussiany są grupowane w chunki po 256. Na chunk w nagłówku osobno zapisywane są granice Min/Max dla pozycji, skali i koloru; poszczególne gaussiany referencjonują swoje wartości względem tych granic i są kompresowane do po 32 bity: pozycja i skala z pakowaniem 11-10-11 bitów, rotacja jako kwaternion 2-10-10-10 bitów „Smallest-Three„, kolor jako RGBA 8-8-8-8. Wyższe współczynniki SH są kwantyzowane tylko do 8 bitów na komponent (`shCoeffCount * 3` uchar na gaussian). Sam format wciąż jest PLY z nagłówkiem ASCII i przez to zasadniczo walidowalny narzędziami PLY, ale właściwości wierzchołków są zadeklarowane jako pola `uint`. Stopień SH jest domyślnie 0 (brak współczynników reszty), aby maksymalizować kompresję — wyższe stopnie SH można jawnie wybrać.

PROSTYMI SŁOWAMI

Oszczędna na miejscu wersja PLY. Identyczna zgodność z silnikami jak normalne PLY, ale 5- do 10-krotnie mniejsze. SuperSplat i PlayCanvas czytają natywnie. Do wdrożeń webowych prawie zawsze lepsze niż normalne PLY. Strata jakości z kwantyzacji wizualnie zwykle niezauważalna, dopóki scena nie zawiera ekstremalnie wysokich częstotliwości szczegółów.

E3 — SPZ (.spz)

GDZIE

Pasek menu → Export → 3D Formats → Export SPZ... Tryb początkujący: karta formatu „SPZ„. **Rozmiar:** ok. 10% względem PLY (90% mniejszy). **Zgodny z:** Niantic Scaniverse, Niantic Spatial Fields, MetalSplatter.

TECHNICZNIE

Format SPZ-v2 od Niantica. Pozycje są pakowane jako 24-bitowy Fixed-Point (daje ok. 0,25 mm rozdzielczości), skale jako kwantyzacja 8-bitowa w przestrzeni log, rotacje jako 8-bitowy Smallest-Three (w v2 zapisywane są tylko xyz, w jest wyprowadzane przez dekodery z normy kwaternionu), opacności jako wartości 8-bitowe po sigmoidzie. SH DC jest zapisywany ze specyficznym dla SPZ wzorem pakowania ($dc_raw * 0.15 * 255 + 0.5 * 255$), wyższe pasma SH z 5 bitami (Band 1) lub 4 bitami (Band 2-3) na współczynnik. Cały zapakowany blob binarny jest następnie kompresowany standardowym gzip (RFC 1952), co daje format kontenera gzipped z magic bytes `1f 8b`. RadianceKit wywołuje do tego systemowy `gzip`, ponieważ wbudowane API zlib firmy Apple generuje zastrzeżone framing Apple, które nie byłoby zgodne z czytnikami SPZ w Spatial Fields ani MetalSplatter. Systemowy `gzip` jest również wewnątrz sandboxa macOS możliwy do spawnowania.

PROSTYMI SŁOWAMI

Najmniejszy standardowy plik. Jeśli znasz Scaniverse od Niantica — to format, którego używa ta aplikacja. Bardzo mały, bardzo przyjazny dla ładowania w aplikacjach mobilnych. W chmurowym viewerze Niantica (Spatial Fields) bezpośrednio użyteczny. Około 90% mniejszy niż PLY z tymi samymi danymi, przy tym dla większości scen optycznie ledwo do odróżnienia.

E4 — SOG (.sog)

GDZIE

Pasek menu → Export → 3D Formats → Export SOG... Tryb początkujący: karta formatu „SOG”.

Rozmiar: ok. 5–6% względem PLY (15- do 20-krotna kompresja — najmniejsza opcja). **Zgodny z:** silnik PlayCanvas, edytor SuperSplat.

TECHNICZNIE

„Spatially Ordered Gaussians” — format PlayCanvas, który zapisuje chmurę gotową dla GPU w kilku obrazach WebP bezstratnych. Najpierw wszystkie gaussiany są przestrzennie sortowane kodem Morton 3D (30-bit Z-Order, po 10 bitów na oś), co zapewnia obrazom późniejszą lokalność cache w renderze. Następnie pozycje są symetryczną transformacją log (dla lepszego zakresu dynamicznego) kwantyzowane do wartości 16-bitowych i dzielone na dwa obrazy RGBA (`means_l.webp` dla dolnych 8 bitów, `means_u.webp` dla górnych). Rotacje są kodowane jako Smallest-Three z 3×8 bitami plus 2-bitowym trybem w jednym obrazie RGBA (tryb łąduje w alfa jako `252 + largest`). Skale i SH DC są kwantyzowane każdy z 256-wpisowym codebookiem (rozproszonym percentylowo na wszystkich wartościach), indeksy łądują w `scales.webp` i `sh0.webp` . Pięć obrazów plus `meta.json` z codebookami i granicami pakowane jest do pliku ZIP (własny enkoder, ponieważ sandbox blokuje systemowy `zip`) i zapisywane z rozszerzeniem `.sog` .

Uwaga sandbox: SOG to jedyna opcja formatu wymagająca zewnętrznego pliku wykonywalnego. Etap enkodera WebP wywołuje `cwebp` z `/usr/local/bin/cwebp` lub `/opt/homebrew/bin/cwebp`. Jeśli żaden plik wykonywalny `cwebp` nie zostanie znaleziony, kod spada na surowe kodowanie PNG — ale: **fallback PNG nie działa w SuperSplat**. W wersji App Store ocenij dostępność na podstawie wariantu builda; w wariacie deweloperskim `cwebp` musi być zainstalowane przez Homebrew (`brew install webp`).

PROSTYMI SŁOWAMI

Najmniejszy format 3DGS w ogóle, znacznie mniejszy niż SPZ. Ale: potrzebuje narzędzia `cwebp` na twoim Macu, ponieważ RadianceKit sam nie potrafi generować wszystkich formatów obrazu. Zainstaluj je jeden raz przez Homebrew, potem wszystko działa. W wersji App Store ewentualnie nie w pełni funkcjonalny — jeśli przy eksporcie wyjdzie PNG zamiast WebP, nie możesz pliku otworzyć bezpośrednio w SuperSplat. Kto pracuje bez Homebrew, bierze zamiast tego SPZ (E3).

E5 — glTF (.glb)

GDZIE

Pasek menu → Export → 3D Formats → Export glTF... Tryb początkujący: karta formatu „glTF„. **Rozmiar:** porównywalny z PLY. **Zgodny z:** viewery glTF z rozszerzeniem KHR_gaussian_splatting (standard draft Khronos).

TECHNICZNIE

Zapisuje samowystarczalny plik binarny `.glb` (brak osobnego załącznika bin) zgodnie ze specyfikacją rozszerzenia KHR_gaussian_splatting. Pozycje są zapisywane jako zwykłe dane wierzchołków `POSITION glTF (float3)`, wszystkie inne atrybuty (rotacja jako `float4`, skala jako `float3`, opacity jako `float`, współczynniki SH jako `float3 × shCoeffCount`) leżą w dodatkowych atrybutach wierzchołków i są referencjonowane przez rozszerzenie. Ważne: glTF używa praworęcznego układu Y-up, COLMAP/3DGS pracuje Y-down/Z-forward. Eksporter stosuje więc obrót o 180 stopni wokół osi X — pozycje są przepisane jako `(x, -y, -z)`, kwaterniony dostosowywane do `(w, x, -y, -z)`. To daje geometrycznie poprawną, zgodną z ręcznością (nie odbitą lustrzanie) prezentację w viewerach glTF. Chunki JSON i binarne są paddingowane do wyrównania 4-bajtowego, jak wymaga standard GLB.

PROSTYMI SŁOWAMI

Oficjalny format standardu Khronos dla danych 3D, w świeżym rozszerzeniu dla splotów gaussowskich. Zaleta: glTF jest rozpowszechniony we wszystkich dużych silnikach 3D (Babylon.js, Three.js, Unity, Unreal). Wada: rozszerzenie jest w 2026 jeszcze w stadium draftu, wiele viewerów jeszcze go nie potrafi. Sensowne przede wszystkim, gdy integrujesz dane splot z istniejącym potokiem glTF lub piszesz viewer, który już jest zdolny do glTF.

I E6 — Splat (.splat)

GDZIE

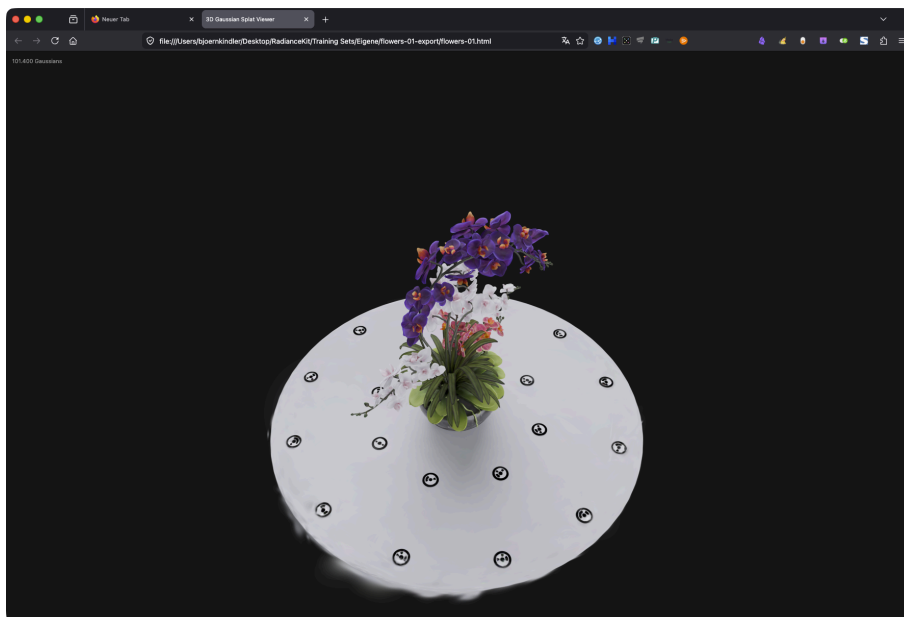
Pasek menu → Export → 3D Formats → Export .splat... Tryb początkujący: karta formatu „.splat„. **Rozmiar:** dokładnie 32 bajty na gaussiana. **Zgodny z:** gsplat.js, viewery webowe (referencja antimatter15), większość przeglądarkowych dem 3DGS.

TECHNICZNIE

Format `.splat` antimatter15 — 32 bajty na gaussiana, brak nagłówka, brak indykcji. Układ na wpis: 3 × float32 pozycja (współrzędne świata), 3 × float32 skala (po transformacji exp z przestrzeni log wewnętrznego bufora), 4 × uint8 kolor RGBA (współczynnik SH DC skalowany przez `SH_C0 = 0.282...` i ograniczony do `[0,255]`), 4 × uint8 kwaternion (w,x,y,z, znormalizowany i zakodowany w zakresie bajtu jako `128 + 128*q`). Tylko SH DC jest zapisywane — wyższe pasma SH są odrzucane. To czyni format niezwykle zwartym, ale kosztuje zmiany koloru zależne od widoku, które występują przy odbiciach lub spekularnych refleksach. Kolejność zapisu to dokładnie kolejność indeksów chmury (bez sortowania przestrzennego), viewery webowe takie jak `gsplat.js` renderują od tego punktu wyjścia.

PROSTYMI SŁOWAMI

Format z wyboru, gdy chcesz prezentować splat w własnym viewerze webowym z `gsplat.js`. Bardzo zwarty (32 bajty/gaussian), ale brak wyższego stopnia SH — czyli brak błyszczących odbić ani drobnych zmian koloru zależnych od kąta widzenia. Dla większości zastosowań webowych żaden problem, ponieważ kolor DC całkowicie wystarcza, a brak zależności od widoku ledwo rzuca się w oczy.



Rysunek 31: Web Viewer otwarty w Firefoxie — splat bukietu Bjoerna renderowany z otaczającymi sferami znaczników kamer, pasek kart przeglądarki widoczny u góry, brak konfiguracji CDN/serwera. Samodzielny `flowers-01.html` otwarty bezpośrednio z Findera podwójnym kliknięciem w domyślnej przeglądarce — osadzony program WebGL2 renderuje chmurę gaussianów natychmiast, bez sieci ani serwera. Czarne znaczniki wokół bukietu to kamery treningowe, opcjonalnie wyświetlane. Przeciągnięcie myszką obraca, scroll zoomuje.

| E7 — Web Viewer (.html)

GDZIE

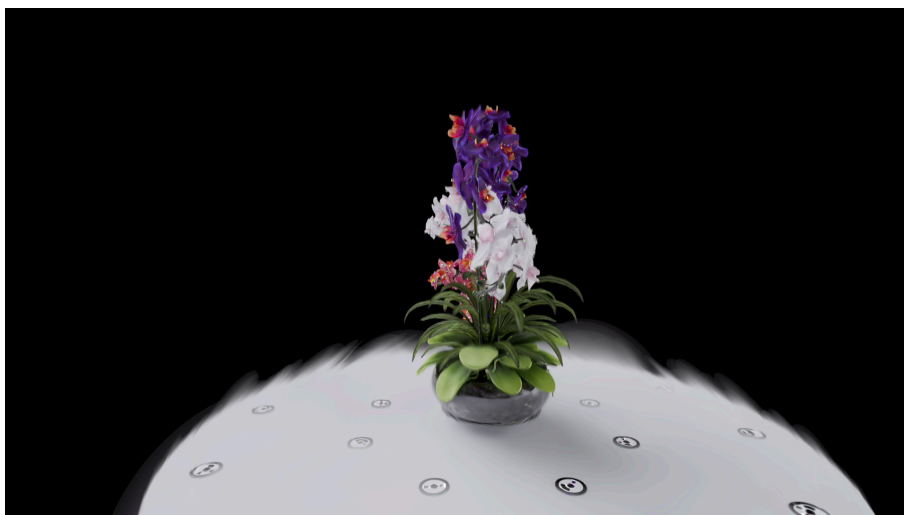
Pasek menu → Export → Media → Export Web Viewer... Tryb początkujący: karta formatu „Web Viewer„. **Rozmiar:** dane splat kodowane base64 ($\approx 4/3$ narzutu) + ok. 5 KB powłoki HTML/JS. **Zgodny z:** każda nowoczesna przeglądarka z WebGL2 (wszystkie desktopy, iOS 15+, Android 5+).

TECHNICZNIE

Pakuje chmurę gaussianów wraz z w pełni inlinem zapisanym renderem WebGL2 w pojedynczy plik `.html`. Nie ma zależności od CDN, nie ma WASM, nie ma drugiego pliku. Chmura jest wewnętrznie najpierw kodowana jako binarna `.splat` (ta sama logika 32 bajtów jak E6), potem osadzona w base64, potem dekodowana w przeglądarce przez `atob`. Wbudowany renderer wykonuje własne sortowanie WebGL2, sterowanie orbitą myszy i sortowanie CPU na ramkę; cały kod JS (shadery, matematyka, pętla) jest widoczny w wyjściowym HTML. Konwencja osi na granicy zapis-do-rendera jest dokładnie taka sama jak w E5: pozycja `(x, -y, -z)`, kwaternion `(w, x, -y, -z)`. Opcjonalnie może być wyświetlany branding overlay (przełącznik Free Tier). Ponieważ wszystko jest inlinem, plik działa również bezpośrednio z protokołu `file://` — brak lokalnego serwera WWW potrzebnego do testowania.

PROSTYMI SŁOWAMI

Pojedynczy plik HTML, który możesz komuś przesłać mailem lub osadzić na stronie internetowej. Podwójne kliknięcie w Finderze i przeglądarka pokazuje twoją scenę z obracaniem myszką. Brak konieczności wysyłania do chmury, brak drugiego pliku, brak serwera. Idealne do prezentacji klientom, portfolio, załączników mailowych. Wada: plik jest dzięki kodowaniu base64 około jedną trzecią większy niż czysty `.splat` — dla bardzo dużych scen opłaca się więc osobne hostowanie pliku `.splat` razem ze standardowym viewerem.



Rysunek 32: Pojedyncza klatka wyekstrahowana z flowers-01.mp4 — bukiet Bjoerna w renderze profilowym, biała platforma ze znacznikami kamer widoczna, czarne tło (domyślne tło widoku, zmienne w Ustawieniach). Kamera okrąża scenę po parametrycznym torze (Elevation + Dystans stałe, Yaw obraca), czas trwania typowo 6–10 sekund przy 30 lub 60 fps. Rozdzielczość klatki skalowalna od 480p do 8K przez VideoPreset.

| E8 — Orbit Video (.mp4/.mov)

GDZIE

Pasek menu → Viewport → Record Turntable Video LUB pasek menu → Export → Media → Export Orbit Video... Tryb początkujący: karta formatu „Orbit Video”, z suwakiem czasu trwania 3–30 s. **Rozmiar:** zależnie od czasu trwania, rozdzielczości, bitrate. **Zgodny z:** wszystkie platformy (H.264 i HEVC to standard Apple).

TECHNICZNIE

Renderuje chmurę gaussianów wzdłuż parametrycznej orbitalnej ścieżki kamery i koduje każdą klatkę przez AVAssetWriter do pliku MP4 lub MOV. Konfiguracja orbity steruje liczbą obrotów (obroty), dystansem, elewacją, FOV, czasem trwania i współczynnikiem Ease-In/Out. Eksport wideo orbitalnego przechodzi przez WŁASNY ForwardPass RadianceKit z pełną ewaluacją SH — pikselowo identyczny z viewportem w aplikacji (WYSIWYG). Na klatkę macierz dopasowania świata (obliczana przez renderer, aby obrócić wewnętrzne współrzędne do orbitalnego świata Y-up) jest mnożona przez kamerę, następnie stosowane jest odbicie konwencji kamery (camFlip: orbitalne Y-up → COLMAP Y-down). Cel renderingu offscreen jest przez IOSurface przyciągany do CVPixelBuffera dla enkodera. Enkoder wspiera H.264 i HEVC, konfigurowalny bitrate i rozdzielczość od 480p do 8K. Przed pierwszą klatką renderer czeka 200 ms, aby początkowe sortowanie splotów zostało ukończone. Ten eksport jest GPU-bound — przy 8K i milionach gaussianów czas renderowania na klatkę leży na poziomie kilku sekund, czyli całkowite czasy renderowania 10–30 minut dla 6 s wideo są możliwe.

PROSTYMI SŁOWAMI

Gotowy plik MP4 z obrotem wokół twojej sceny. Idealny do social media, marketingu, prezentacji. Możesz ustawić czas trwania (3–30 sekund), kierunek obrotu i prędkość. Plik można osadzać bezpośrednio na YouTube, Instagramie, w PowerPointcie i wszędzie indziej. Idzie częściowo wolno, ponieważ aplikacja musi renderować każdą klatkę kompletnie — dla wideo 8K możesz planować pięć do trzydziestu minut, zależnie od liczby gaussianów.

I E9 — Transforms SfM (transforms.json)

GDZIE

Pasek menu → Export → Photogrammetry → Export SfM (transforms.json)... **Rozmiar:** typowo 1–10 KB (tylko pozy + intrinsics, brak obrazów, brak gaussianów). **Zgodny z:** nerfstudio, Brush, gsplat, OpenSplat, Meshroom, wszystkie nowoczesne trenerzy 3DGS feed-forward.

TECHNICZNIE

Zapisuje format nerfstudio `transforms.json` z listą pól kamer plus dzielonymi intrinsics. Na kamerę macierz widoku (wewnętrznie w RadianceKit: World-to-Camera w konwencji COLMAP) jest odwracana, następnie kameralne wektory bazowe Y i Z są odbijane lustrzanie, aby przejść do konwencji nerfstudio (styl OpenGL, kamera patrzy wzdłuż `-Z`, `+Y` jest w górze). Finalna macierz 4x4 łąduje jako row-major zagnieżdżona tablica double w polu `transform_matrix` każdej klatki. Intrinsics są zapisywane na poziomie top-level (ogniskowa x/y , punkt główny x/y , szerokość/wysokość obrazu, `camera_model = "OPENCV"`, plus współczynniki dystorsji `k1, k2, p1, p2`) — chyba że eksporter wykryje kilka różnych zestawów intrinsics, wtedy są zapisywane na klatkę. Ścieżki obrazów są zapisywane jako `images/<filename>` względem pliku JSON; użytkownik musi utworzyć siostrzany folder `images/` z fotografiami treningowymi.

PROSTYMI SŁOWAMI

Ten plik JSON opisuje dla każdego zdjęcia, gdzie stała kamera i dokąd patrzyła. Sam plik jest mały i bezużyteczny — używany jest razem z oryginalnymi obrazami w folderze. Nerfstudio, Brush i kilka innych trenerów czyta dokładnie ten format, a ty możesz przekazać swoje wyniki SfM z RadianceKit do innego narzędzia bez konieczności ponownego liczenia rekonstrukcji kamer tam. Przy dużych scenach oszczędza godziny.

I E10 — Workspace COLMAP (sparse/0/)

GDZIE

Pasek menu → Export → Photogrammetry → Export SfM (COLMAP Workspace)... **Rozmiar:** trzy pliki binarne razem typowo 4–8 MB — `points3D.bin` dominuje (jedna linia na punkt 3D chmury sparse), `images.bin` i `cameras.bin` są każdy znacznie poniżej 100 KB. **Zgodny z:** sam COLMAP, Nerfstudio, Postshot, Meshroom, wszystkie narzędzia oczekujące katalogu `sparse/ COLMAP`.

TECHNICZNIE

Zapisuje standardowy układ `sparse/0/ COLMAP` z trzema plikami binarnymi: `cameras.bin`, `images.bin`, `points3D.bin`. Referencja formatu to oficjalna dokumentacja COLMAP. `cameras.bin` zawiera zdeduplikowaną listę `intrinsics` (kamery z identycznymi `intrinsics` + rozmiarem obrazu są scalane do pojedynczego wpisu); używany model kamery to `OPENCV` (Model 4), z `fx/fy/cx/cy` plus czterema współczynnikami dystorsji `k1/k2/p1/p2`. `images.bin` listuje na obraz pozę jako kwaternion `wxyz` plus translację, za nimi ID kamery i nazwa pliku; nie są zapisywane korespondencje 2D-3D. `points3D.bin` zawiera chmurę punktów SfM z pozycją, kolorem (RGB 0-255) i wartościami domyślnymi dla reprojekcji i długości ścieżki. Wszystko zapisywane w Little-Endian. Re-import do Radian-ceKit działa przez menu File → „Import COLMAP/ Metashape Workspace...” (zobacz Q3 w rozdziale o backendach SfM).

PROSTYMI SŁOWAMI

Oficjalny format COLMAP. Jeśli chcesz kontynuować trening w Postshot, Nerfstudio lub innym oprogramowaniu zdolnym do COLMAP, to jest droga. Trzy małe pliki plus twoje oryginalne obrazy, a program docelowy akceptuje to, jakby sam COLMAP był programem źródłowym. Więcej programów to rozumie niż format `transforms.json` (E9), jednocześnie nieco mniej poręczne, ponieważ binarne zamiast tekstowych.

Który format kiedy?

Cel	Format
Viewer webowy na własnej stronie	E7 Web Viewer (.html)
Viewer webowy z <code>gspLat.js</code>	E6 Splat (.splat)
Reuse potoku w Postshot / Nerfstudio	E9 transforms.json + E10 Workspace COLMAP
Edycja w SuperSplat	E1 PLY lub E2 Compressed PLY
Niantic Scaniverse / Spatial Fields	E3 SPZ
Maksymalna kompresja	E4 SOG (wymagane cwebp)
Wideo marketingowe/społecznościowe	E8 Orbit Video

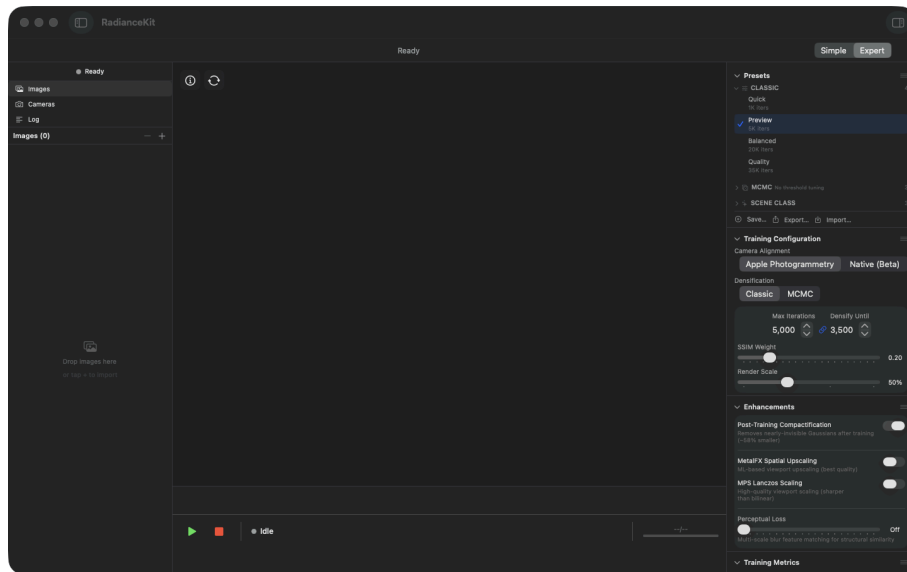
Szybkie porównanie

Format	Rozszerzenie	Sandbox	Rozmiar (1M Gauss)	Najlepsze użycie
E1 PLY	.ply	tak	~250 MB	Archiwum, najwyższa zgodność
E2 Compressed PLY	.ply	tak	~40 MB	Web + SuperSplat
E3 SPZ	.spz	tak (spawn gzip)	~40 MB	Niantic + Mobile
E4 SOG	.sog	warunkowo (cwebp)	~20 MB	Maksymalna kompresja
E5 glTF	.glb	tak	~250 MB	Potok Khronos
E6 Splat	.splat	tak	~32 MB	Viewer webowy gsplat.js
E7 Web Viewer	.html	tak	~45 MB	Samodzielny plik przeglądarki
E8 Orbit Video	.mp4 / .mov	tak	zmienny	Social/Marketing
E9 SfM Transforms	.json	tak	~5 KB	Przekazanie póż
E10 Workspace COLMAP	Katalog	tak	~4–8 MB	Przekazanie póż binarne

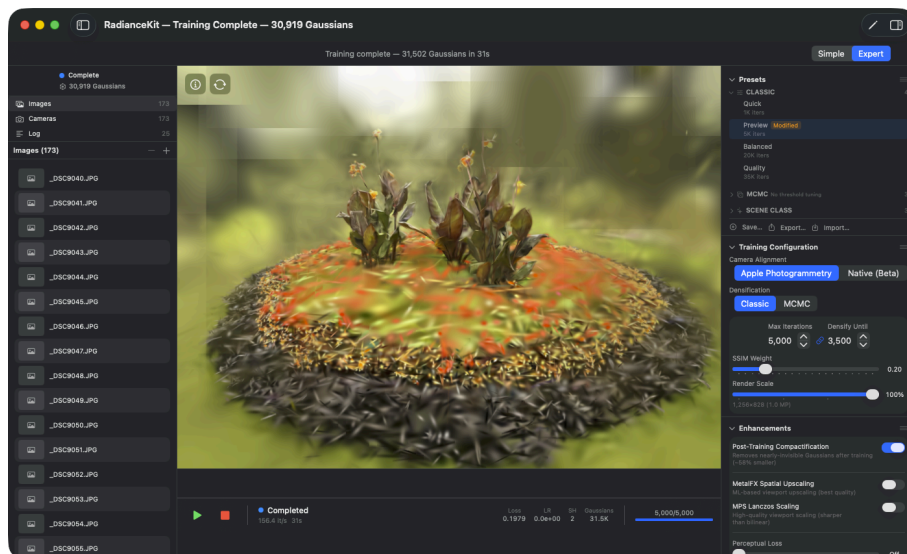
Kolumna rozmiarów to przybliżone wartości referencyjne dla 1 mln gaussianów ze stopniem SH 3. Rzeczywiste wartości zmieniają się zależnie od kompresowalności sceny; stopień SH 0 redukuje PLY/glTF o współczynnik 4.

ROZDZIAŁ

Rozdział 9 — Backendy SfM



Rysunek 33: Tryb Eksperta z selektorem Camera Alignment w Inspectorze (Apple Photogrammetry / Native (Beta))



Rysunek 34: Inspector z aktywnym Native (Beta) — w selektorze Camera Alignment wybrana druga opcja, wszystkie pozostałe parametry konfiguracji treningu bez zmian

CO WIDAĆ NA OBRAZIE Selektor Camera Alignment w Inspectorze to kontrolka segmentowa z dwoma opcjami — Apple Photogrammetry (domyślne w buildach App Store, w pełni zgodne z sandboxem) oraz Native (Beta) (własny backend RadianceKit oparty na

potoku FAST+BRIEF+GLOMAP, opracowany w Fazie 3.8/3.9, stan na 2026-05). Native (Beta) jest zwalidowane tylko dla scen orbitalnych i szybsze przy ≥ 1000 klatkach niż Apple Photogrammetry, ale nie spełnia jeszcze bramki jakości §5 Fazy 3 (finalLoss ≤ 0.0115) — stąd oznaczenie Beta. Zewnętrzne wyniki SfM z Metashape, COLMAP lub innego oprogramowania fotogrametrycznego można dodatkowo zaimportować przez menu File (Q3 format tekstowy COLMAP, Q6 import workspace) — selektor się nie zmienia, ale zaimportowane pozy zastępują wynik SfM.

SfM oznacza **Structure from Motion**. Z zestawu nakładających się zdjęć oprogramowanie rekonstruuje dla każdego obrazu pozycję i kierunek patrzenia kamery we wspólnym układzie współrzędnych 3D. Przy tej okazji powstaje zgrubna chmura punktów 3D, która inicjalizuje trening Gaussian Splatting. Wynik SfM jest wejściem dla właściwego treningu i decyduje w znaczącym stopniu o późniejszej jakości obrazu.

RadianceKit oferuje pięć dróg SfM: dwa wbudowane backendy (Q1 Apple Photogrammetry, Q4/Q5 Native), dwie ścieżki importu z zewnętrznych narzędzi (Q3 format tekstowy COLMAP, Q6 binarny import workspace) oraz Q2 COLMAP-Binary, dostępne tylko w buildach deweloperskich poza App Store. Który jest właściwy, zależy od typu sceny (orbita wokół obiektu, wnętrze, lot dronem) oraz od tego, czy zewnętrzne oprogramowanie już dostarcza rekonstrukcję.

I Q1 — Apple Photogrammetry

GDZIE

Expert View → Inspector → Konfiguracja treningu → selektor Camera Alignment, pozycja „Apple Photogrammetry„.

TECHNICZNIE

Opakowuje wbudowany framework Photogrammetry firmy Apple, pierwotnie opracowany dla Object Capture. Apple wewnętrznie ekstrahuje cechy za pomocą zastrzeżonego potoku (kroki nie są publicznie udokumentowane), weryfikuje je przez dopasowywanie wielowidokowe i rozwiązuje Bundle Adjustment na Apple Silicon Neural Engine + GPU.

Backend jest w pełni zgodny z App Store (brak zewnętrznego pliku wykonywalnego, Sandbox=true, on-device), ale dostarcza tylko pozy kamer plus zgrubną chmurę punktów — bez metryk diagnostycznych takich jak długość ścieżki czy błąd reprojekcji. Skaluje się zgodnie z zaleceniami Apple do kilkuset zdjęć. Przy ponad ~500 klatkach w liniowych lotach dronem lub dużych scenach plenerowych obserwowano powtarzalne crashe lub ciche odrzucanie pojedynczych kamer.

PROSTYMI SŁOWAMI

To najprostsza droga. Wrzucasz zdjęcia, aplikacja liczy. Działa bardzo dobrze przy klasycznych skanach obiektów — gdy obchodzisz mebel lub rzeźbę i robisz 50–200 zdjęć. Przy lotach dronem nad krajobrazem lub przy bardzo wielu zdjęciach (powyżej 500) procedura Apple chętnie staje się jednak niestabilna. W takich scenach przetestuj backend Native (Q4/Q5) lub policz kamery w Metashape i wczytaj je przez import workspace (Q6).

POWER-USER

Q2 COLMAP-Binary — uruchamia zewnętrzny program COLMAP jako podproces i dlatego w wersji App Store (sandbox) **nie jest dostępny**. Działa tylko w buildach deweloperskich poza App Store. Dla jakości, jaką dostarcza COLMAP, w wersji App Store jest dostępny import workspace (Q3 lub Q6): policz SfM w COLMAP-ie lub Metashape na zewnątrz i wczytaj wynik.

Q3 — format tekstowy COLMAP (Metashape / ETH3D) **GDZIE**

Menu „File → Import COLMAP / Metashape Workspace...„ (Cmd+⌘+I) LUB przeciągnięcie folderu z `sparse/0/cameras.txt`.

 **TECHNICZNIE**

Czyta standardowy eksport tekstowy COLMAP — trzy pliki tekstowe `cameras.txt`, `images.txt`, `points3D.txt` w podfolderze `sparse/0/` — i konwertuje do wewnętrznego modelu wyniku SfM. Ta sama definicja formatu co binarny eksport COLMAP, tylko jako ASCII zamiast formatu binarnego. Jest tak właśnie wysyłany przez Agisoft Metashape, RealityCapture, PolyCam oraz benchmark ETH3D. Parser dzieli rozpoznawanie modelu kamery z parserem binarnym (wszystkie typowe modele: SIMPLE_PINHOLE, PINHOLE, OPENCV, OPENCV_FISHEYE, FULL_OPENCV). Odporny na linie komentarzy i puste linie. W testach skaluje się do ~1 400 kamer (ETH3D Tunnel) bez problemów.

 **PROSTYMI SŁOWAMI**

Jeśli już pracowałeś z Metashape, RealityCapture lub innym komercyjnym oprogramowaniem foto-3D i wyeksportowałeś wynik — ten eksport możesz wczytać do RadiancKit bezpośrednio, bez ponownego liczenia w aplikacji. To oszczędza godziny czekania. Po prostu wczytaj cały folder przez menu File lub przeciągnij go do okna.

I Q4 — Native SfM (inkrementalne)

GDZIE

Expert View → Inspector → Konfiguracja treningu → selektor Camera Alignment, pozycja „Native (Beta) „. Tryb inkrementalny jest domyślnym trybem tego backendu — w Inspectorze nie ma osobnego selektora mappera. Przez CLI tryb można jawnie ustawić za pomocą `--native-sfm` lub `--sfm-mapper incremental`.

TECHNICZNIE

Własna implementacja całego potoku SfM przyspieszona przez GPU: cechy FAST+BRIEF LUB SuperPoint+LightGlue przez CoreML (z `--coreml-features`), następnie Hamming-KNN matching, RANSAC z macierzą fundamentalną, budowanie ścieżek, wybór pary początkowej, two-view bootstrap (F→E plus DLT), zachłanny inkrementalny mapper z rejestracją PnP i triangulacją wielowidokową oraz końcowy Bundle Adjustment przez Schur-zredukowany Levenberg-Marquardt z funkcją straty Hubera i analitycznymi jacobianami przez Cholesky-Solve. W pełni zgodny z App Store: brak zewnętrznego pliku wykonywalnego, `Sandbox=true`. Z wysłanym w Fazie 3.10 detektorem R2-Collapse: jeśli aplikacja zarejestruje mniej niż 60% klatek wejściowych lub współczynnik punktów na kamerę spadnie poniżej 13, automatycznie przełącza się na mapper globalny (Q5). Empirycznie czyste na scenach orbitalnych/obrotowych; na bardziej ogólnych ruchach (lot dronem, wnętrza ze złożoną geometrią) wskaźnik powodzenia jest niższy — detektor wychwytuje jednak te przypadki. Skaluje się niezawodnie do ~200 kamer, wyżej z istotnie dłuższym czasem działania.

PROSTYMI SŁOWAMI

Mocne strony Apple (zgodność z App Store, szybkość dla orbit) z dodatkowymi wartościami diagnostycznymi. Działa szczególnie dobrze, gdy jak przy Object Capture obchodzisz motyw. Przy bardziej skomplikowanych ujęciach (lot dronem lub salon) RadianceKit automatycznie wykrywa, że nic z tego nie wyjdzie, i przeskakuje na procedurę globalną. Oznaczone „Beta,,, ponieważ wciąż w testach — standardowa rekomendacja to nadal Apple Photogrammetry dla prostych skanów obiektów i import workspace (Q3 lub Q6) dla wymagających zestawów plenerowych.

I Q5 — Native SfM (globalne)

GDZIE

Wywoływane automatycznie, gdy inkrementalny mapper (Q4) wyzwole detektor kolapsu (zarejestrowane mniej niż 60% klatek wejściowych lub współczynnik punktów na kamerę poniżej 13). Ręczne wymuszenie tylko przez CLI `--sfm-mapper global`. W Inspectorze procedura globalna nie jest dostępna przez osobny selektor — aplikacja sama decyduje, kiedy się przełączyć.

TECHNICZNIE

Globalna odmiana potoku natywnego. Najpierw ekstrakcja cech + matching jak w Q4, potem oszacowanie pozy względnej dla wszystkich zweryfikowanych par, następnie Rotation Averaging (synchronizuje wszystkie obroty kamer w układzie współrzędnych świata) i Translation Averaging (LSQR oparte na bezmacierzowym sformułowaniu sparse, aby uniknąć przepełnienia liczb całkowitych przy dużych zbiorach kamer). Skaluje się w zasadzie do ~5 000 kamer, w praktyce z pogorszoną jakością powyżej kilkuset kamer — pomiar bramki akceptacji §5 Fazy 3.8 na K-1351 dał finalLoss 0.07 zamiast docelowych 0.0115. Traktowane jako „tier zapasowy”: wchodzi do akcji, gdy mapper inkrementalny degeneruje, ale sam nie jest ponownie sprawdzany pod kątem jakości.

PROSTYMI SŁOWAMI

Ścieżka planu B dla silnika natywnego. Wywoływana automatycznie, gdy szybsza ścieżka inkrementalna zawodzi. Dostarcza użytecznego wyniku, ale w bardzo dużych lub trudnych scenach zwykle nie jest tak precyzyjna jak to, co otrzymujesz z Metashape lub zewnętrznej instalacji COLMAP. Jeśli Native stanie się twoim standardowym przepływem pracy, w takich przypadkach opłaca się obejście przez import workspace (Q3 lub Q6).

I Q6 — Import workspace Metashape / COLMAP-Text (Faza Q7)

GDZIE

Menu File → „Import COLMAP / Metashape Workspace...„ (Cmd+⇧+I). Przeciągnięcie folderu z `sparse/0/cameras.{bin,txt}` oraz `images/`.

TECHNICZNIE

Automatycznie rozpoznaje, czy folder wybrany przez przeciągnięcie lub Open-Panel odpowiada jednemu z trzech układów workspace COLMAP (`sparse/0/`, `sparse/` lub `root`) oraz czy rekonstrukcja jest w formie binarnej (`cameras.bin`) czy tekstowej (`cameras.txt`). Ścieżka binarna używa parsera binarnego COLMAP, ścieżka tekstowa — loadera ETH3D — oba produkują ten sam model wyniku SfM i reszta potoku (import obrazów, start treningu MCMC) jest agnostyczna wobec źródła. Obrazy są otwierane przez system zakładek sandboxa aplikacji w trybie security-scoped, dzięki czemu import działa również w wersji App Store. Specjalnie dla przypadku „eksport Metashape bez ponownego liczenia rekonstrukcji,„. Rozpoznawanie wspomniane w pozycji menu File ostrzega w logu aplikacji, jeśli wybrany folder nie jest rozpoznawalnym workspace.

PROSTYMI SŁOWAMI

Specjalnie funkcja dla użytkowników Metashape. Jeśli masz licencję na Metashape lub RealityCapture i tam wykonałeś rekonstrukcję kamer, możesz po prostu wciągnąć folder eksportu tutaj i od razu rozpocząć trening. Przy dużych scenach oszczędza to wiele godzin czasu obliczeniowego, ponieważ RadianceKit nie wykonuje wtedy SfM samodzielnie.

Który backend kiedy?

Scenariusz	Zalecany backend
Skan obiektu, 50–200 zdjęć	Q1 Apple Photogrammetry
Duży plener / dron / >500 zdjęć	Q6 Workspace-Import (policz w Metashape lub COLMAP, potem wczytaj)
Istnieje eksport Metashape/RealityCapture	Q6 Import (SfM niepotrzebne)
ETH3D / akademicki zestaw COLMAP-Text	Q3 COLMAP-Text-Import
Ściśle zgodne z App Store + scena orbitalna	Q4 Native inkrementalne
Q4 zawodzi	Q5 Native globalne (automatycznie)
Dane benchmarku ETH3D	Q3 (autotest precomputed)

Szybkie porównanie

Bac- kend	App Store	Sand- box	Zewnętrzny plik wykonywalny	Najlep- sze uży- cie	Maks ~ka- mer
Q1 Apple PG	✓	✓	—	Orbit-Ob- ject	~300
Q2 COL- MAP Bi- nary	✗ (tylko bu- ild dev.)	—	colmap/glomap	Duży ple- ner	~5 000
Q3 COL- MAP- -Text-Im- port	✓	✓	—	Stano- wiska benchm.	~1 500
Q4 Nati- ve incre- mental	✓	✓	—	Orbit-Ob- ject	~200
Q5 Nati- ve global	✓	✓	—	Fallback Q4	~1 351
Q6 Work- space- -Import	✓	✓	—	Reuse Me- tashape	zależnie od źródła

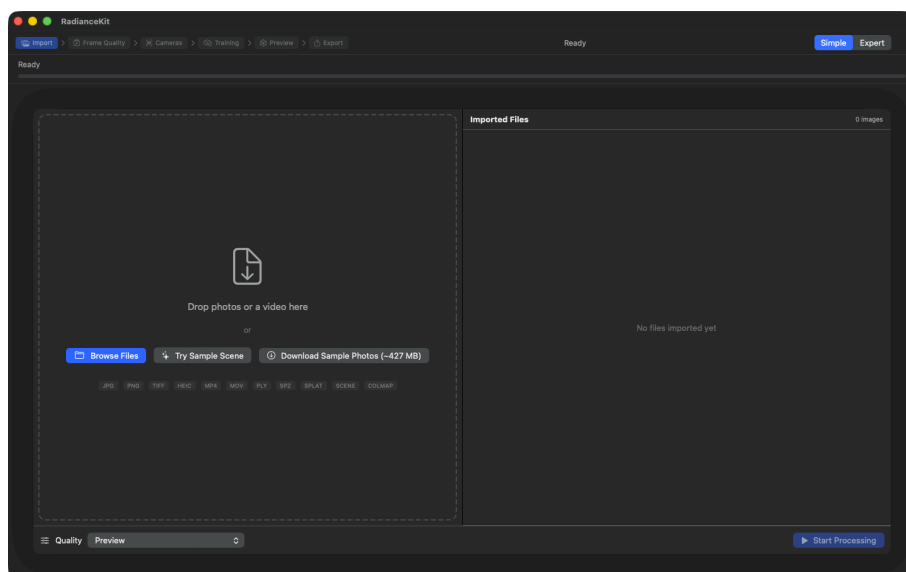
ROZDZIAŁ

Rozdział 10 — Tryb początkujący

Der Einsteiger-Modus (engl. Simple Mode, Cmd+1) ist der geführte Workflow für alle, die zum ersten Mal eine 3D-Gaussian-Splatting-Szene rekonstruieren. Statt eine Sidebar voller Inspector-Felder anzuzeigen, führt die App durch vier Schritte: zuerst Bilder oder ein Video importieren und ein Qualitäts-Preset wählen, dann läuft die Verarbeitung (SfM + Training), anschließend kann die fertige Szene in einer 3D-Vorschau begutachtet werden, und zum Schluss wird in das gewünschte Format exportiert. Eine schmale Fortschrittsleiste am oberen Fensterrand zeigt jederzeit an, in welchem Schritt du gerade bist.

Im Vergleich zum Expert-Modus (Cmd+2), der alle Bedienfelder gleichzeitig zeigt, blendet der Einsteiger-Modus ungenutzte Optionen aus, gibt Validierungs-Warnungen bei zu wenigen oder schlechten Bildern und bietet auf jedem Schritt nur die Buttons an, die im aktuellen Zustand sinnvoll sind. Du kannst jederzeit zwischen Einsteiger- und Expert-Modus wechseln (Cmd+1 / Cmd+2), der gesamte Zustand — importierte Bilder, gewähltes Preset, gerade laufendes Training, fertige Punktwolke — bleibt erhalten und ist im jeweils anderen Modus sofort verfügbar.

Z1 — Import (Bilder & Preset wählen)



Rysunek 35: Simple-Mode Schritt 1 — leere Drop-Zone vor Bild-Import, Crumb-Trail oben (Import → Frame Quality → Cameras → Training → Preview → Export), Format-Pills JPG/PNG/TIFF/HEIC/MP4/MOV/PLY/SPZ/SPLAT/SCENE/COLMAP

WAS IM BILD ZU SEHEN IST Crumb-Trail (Import aktiv) zeigt den vierschrittigen Workflow. Linke Drop-Zone mit drei CTAs: „Browse Files„ (NSOpenPanel), „Try Sample Scene“ (bundled Demo), „Download Sample Photos (~427 MB)„ (Mip-NeRF360 flowers Subset). Format-Pills darunter listen alle akzeptierten Dateitypen. Rechts „Imported Files“ mit Counter „0 images„ und Empty-State „No files imported yet“. Unten Quality-Picker (Default: Preview) und „Start Processing„ (deaktiviert solange keine Bilder da sind).

Der erste Schritt besteht darin, der App Bildmaterial zu geben. Per Drag-and-Drop in das große, gestrichelte Feld in der Mitte, per „Browse Files„-Button oder per Klick auf die mitgelieferte Sample-Szene. Rechts erscheint eine Liste aller importierten Bilder mit Auflösung und Dateigröße; unten in der schwebenden Werkzeugleiste wählst du das Qualitäts-Preset und startest mit „Start Processing“ die Pipeline. Validierungs-Warnungen (rot bei < 3 oder < 10 Bildern, orange bei 10–19) zeigen an, ob die App eine sinnvolle Rekonstruktion erwartet oder nicht.

C-01 ProgressIndicator (Schritt-Anzeige)

GDZIE

Oben über dem Workflow, immer sichtbar.

TECHNICZNIE

Zeigt eine horizontale Fortschrittsleiste über die gesamte Pipeline (Frame-Quality → SfM → Training) mit Stage-Allocation: Frame-Quality belegt 0–5 % (Phase 3.11, sehr kurz), SfM belegt 0–30 % der Bar, Training 30–100 %. Daneben Status-Text und phasen-benannte Prozent-Anzeige („SfM 41 %“, „Training 12 500/20 000“), damit Anwender nicht den scheinbaren Rückschritt „41 % SfM → 25 % Training“, als Fehler lesen — die Bar zeigt den gesamten Pipeline-Fortschritt, nicht die Sub-Stage. ETA-Berechnung beginnt, sobald genug Trainings-Tempo gemessen ist (typischerweise nach den ersten 100 Iterationen). Dieselbe Anzeige wird auch im Expert-Modus oberhalb des Inspectors verwendet.

PROSTYMI SŁOWAMI

Die schmale Leiste ganz oben ist deine Karte durch den Workflow. Sie sagt dir nicht nur, was die App gerade macht (Kamera ausrichten, Training läuft, ...), sondern auch, wie weit sie insgesamt schon ist. Die Aufteilung ist absichtlich so, dass die Kamera-Berechnung das erste Drittel der Leiste belegt und das eigentliche Training die hinteren zwei Drittel — sonst würde es wirken, als wäre der Fortschritt nach SfM plötzlich zurück bei null. Du kannst dich also entspannt zurücklehnen, ein Blick auf die Leiste genügt, um die grobe Etappe zu sehen. Der Text daneben sagt dir, ob du gerade in der SfM-Stufe (z. B. „SfM 41 %“) oder im Training (z. B. „Training 12 500/20 000“) steckst, damit die Zahlen nicht verwirrend wirken. Wenn du die ETA nicht angezeigt bekommst, ist das Training einfach noch zu jung — die App schätzt erst, sobald sie genug Tempo gemessen hat.

C-03 DropZoneView (Drag-and-Drop-Bereich)

GDZIE

Linke Seite des Import-Schritts, großes gestricheltes Rechteck mit Symbol. Wird im Einsteiger-Modus mit dem Label „Drop photos or a video here,“ angezeigt.

TECHNICZNIE

Drop-Bereich, der das Symbol kurz hüpfen lässt und den Hintergrund einfärbt, sobald Drag-Items über dem Feld schweben. Akzeptiert JPG, PNG, TIFF, HEIC, MP4, MOV, PLY, SPZ, .splat, .radiancescene-Bundles und Verzeichnisse. Drop-Routing nach Typ: Bilder werden gesammelt und sortiert übergeben, Videos triggern den Frame-Sampling-Pfad, Splat-Dateien öffnen direkt die Vorschau, Scene-Bundles werden eingelesen. Verzeichnisse werden enumeriert und alle enthaltenen Bilder importiert. Security-scoped Bookmarks für sandbox-konformen Zugriff werden korrekt aufgenommen und freigegeben. Nicht-unterstützte Endungen werden als Warnbanner für 5 Sekunden angezeigt.

PROSTYMI SŁOWAMI

Das große gestrichelte Feld ist die Hauptbedienung des ersten Schritts. Zieh einfach Fotos oder ein Video hinein, oder einen ganzen Ordner — die App nimmt alles, was sie kennt, und ignoriert den Rest. Wenn das Feld blau wird und das Symbol kurz hüpfert, hat die App den Drag erkannt. Lass los, und der Import startet sofort: Bilder wandern in die Liste rechts, Videos triggern automatisch den Frame-Sampling-Schritt, und bereits trainierte .ply / .spz / .splat Dateien öffnen direkt die Vorschau. Falls ein Format gar nicht passt (etwa PDF oder BMP), erscheint ein kurzer Hinweis am oberen Rand — die App schluckt unbekanntes Material nicht stumm.

C-05 Browse Files Button

GDZIE

Innerhalb der Drop-Zone, prominenter Button.

TECHNICZNIE

Button, der den macOS-Datei-Dialog mit Mehrfach-Auswahl und den Dateitypen JPG, PNG, TIFF, MP4, MOV, Ordner sowie dem App-eigenen Scene-Format öffnet. Ergebnis-URLs sind security-scoped und werden über dieselben Import-Pfade weitergeleitet wie Drag-and-Drop. Wenn der Benutzer einen Ordner auswählt, wird er rekursiv nach Bildern enumeriert.

PROSTYMI SŁOWAMI

Wenn dir Drag-and-Drop unbequem ist, klick einfach diesen Button und navigiere im macOS-Datei-Dialog zu deinen Fotos. Du kannst mehrere Dateien gleichzeitig wählen (Cmd-Klick auf die einzelnen Bilder) oder einen ganzen Ordner auswählen — die App durchsucht den Ordner dann rekursiv nach allen unterstützten Bildtypen. Das ist besonders praktisch, wenn deine Aufnahmen verschachtelt in Unterordnern liegen (z. B. „shoot-day1/,,, „shoot-day2/“) — ein Klick auf den Hauptordner reicht. Funktional macht der Button exakt das, was auch Drag-and-Drop macht; wähl einfach den Weg, der dir bequemer ist.

C-06 Try Sample Scene Button**GDZIE**

Innerhalb der Drop-Zone, nur sichtbar wenn das App-Bundle die Sample-Scene enthält und noch keine Bilder/Splats importiert sind.

TECHNICZNI

Erscheint nur, wenn (a) eine `sample-scene.splat`, `.spz` oder `.ply` im App-Bundle vorhanden ist UND (b) noch keine Bilder/Videos importiert und noch keine Punktwolke vorhanden ist. Bei Klick lädt die fertige Punktwolke (bevorzugt das kleinste Format — `.splat` ~3 MB, `.spz` ~1.4 MB, Fallback `.ply`) und setzt nach 400 ms hartkodierte Kamera-Werte aus dem Original-Metadaten der Blumen-Szene für eine ästhetisch sinnvolle Eingangs-Perspektive.

PROSTYMI SŁOWAMI

Wenn du die App zum ersten Mal startest und einfach mal sehen willst, was am Ende rauskommt — klick hier. Das öffnet eine fertig trainierte Blumen-Szene, die du sofort drehen und exportieren kannst, ohne dass die App rechnen muss. Die Kamera ist voreingestellt auf eine ästhetisch sinnvolle Eingangsperspektive, so dass du gleich was Schönes siehst. Perfekt, um die 3D-Steuerung und den Export-Schritt risikolos auszuprobieren, bevor du auf eigene Aufnahmen losgehst. Sobald du eigene Bilder importierst, verschwindet der Button automatisch — er wird nur angezeigt, solange das Projekt komplett leer ist.

C-07 Download Sample Photos Button**GDZIE**

Innerhalb der Drop-Zone, neben „Try Sample Scene“; gleiche Sichtbarkeits-Bedingungen.

TECHNICZNI

Triggert einen Download (Repo github.com/bkindler/radiancekit-sample-photos), der ca. 427 MB an 960 Vollauflösungs-Frames lädt und in die App speist. Während des Downloads wird der Button deaktiviert. Der Fortschritt erscheint in der oberen Progress-Bar als „Downloading X %“, in einer eigenen Stage, weil diese Stage ihre eigene 0–100 %-Skala behält und der spätere SfM-Stage nicht überlappt.

PROSTYMI SŁOWAMI


Genau wie die Sample-Scene, nur mit den Ausgangs-Fotos statt mit dem fertigen Ergebnis. So kannst du die gesamte Pipeline einmal selbst durchlaufen lassen und sehen, wie lange SfM und Training auf deinem Mac wirklich dauern. Der Download ist groß (etwa eine halbe DVD = 427 MB), passiert aber nur einmal — danach sind die Fotos lokal und du kannst die Pipeline beliebig oft mit verschiedenen Presets neu starten. Während der Download läuft, zeigt die obere Fortschrittsleiste den aktuellen Download-Stand in Prozent, sodass du abschätzen kannst, wann es losgeht. Tipp: nimm dazu am besten ein schnelles WLAN oder Kabel-Netz — die 427 MB ziehen sich sonst.

C-09 Quality Presets Picker

GDZIE

Schwebende untere Werkzeugleiste des Import-Overlays, links neben dem Start-Button.

TECHNICZNI

Bedienelement mit Label „Quality“, gruppiert die verfügbaren Presets nach Kategorie (Classic / MCMC / Custom). Built-in Presets werden nach Kategorie gruppiert; die Abschnitts-Header sind hartkodiert. Custom-Presets nur sichtbar wenn welche existieren. Locked-State: Presets, die nicht in der Free-Liste (Quick + Preview) liegen, bekommen ein „“-Suffix am Namen, wenn der User nicht gekauft hat; bei Auswahl springt der Picker auf Preview zurück und öffnet automatisch das Purchase-Sheet. Bei Wahl wird das Preset angewendet, was die gesamte Trainings-Konfiguration ersetzt.

PROSTYMI SŁOWAMI

Hier wählst du, wie genau und wie lange die App rechnen soll. „Quick“, und „Preview“ sind ohne Kauf nutzbar und liefern in wenigen Minuten ein erstes Ergebnis — ideal, um zu testen, ob deine Bilder überhaupt sinnvoll sind. „Balanced“, und „Quality“ brauchen die Vollversion und liefern deutlich saubere Modelle, dauern dafür aber Stunden statt Minuten. MCMC ist eine andere Strategie, die mit weniger Gauss-Splats auskommt — gut, wenn du das Modell später kompakt exportieren oder ins Web stellen willst. Premium-Presets erkennst du am kleinen Schloss-Symbol am Namen; tippst du eines ohne Lizenz an, springt der Picker zurück auf Preview und das Kauf-Sheet öffnet sich automatisch. Faustregel: starte immer mit Preview, schau dir das Ergebnis an, und entscheide dann, ob sich ein längerer Lauf lohnt.

C-10 Start Processing Button

GDZIE

Schwebende untere Werkzeugleiste des Import-Overlays, rechts neben dem Preset-Picker.

TECHNICZNI

Button, der grau bleibt, solange weder Bilder noch ein Video importiert sind. Bei Klick startet die Pipeline und schaltet die Stage-Maschine in die Reihenfolge Frame-Quality → SfM → Training um. Der Button selbst hat keinen weiteren Status; eine laufende Verarbeitung erscheint stattdessen als separater Verarbeitungsbildschirm.

PROSTYMI SŁOWAMI

Der „Loslegen“-Knopf. Solange er grau ist, fehlen noch Eingabe-Bilder oder ein Video. Sobald du Fotos hereingezogen hast, wird er aktiv und du klickst ihn an, um SfM und Training nacheinander zu starten. Ab da übernimmt die App den gesamten Ablauf und du landest automatisch auf dem Verarbeitungsbildschirm (Z2). Du musst nichts weiter klicken — erst nach Trainings-Ende wechselt die App wieder in die Vorschau (Z3). Wenn du es dir nochmal anders überlegst, kannst du auch danach noch jederzeit per Cancel abrechnen.

C-11 Video Sampling Slider

GDZIE

Rechte Bilderliste, sichtbar nur wenn ein Video (statt Bildern) importiert wurde.

TECHNICZNI

Schieberegler 0.5 fps – 30 fps in 0.5er-Schritten. Bei Änderung wird die Frame-Dichte aktualisiert und zusätzlich die Anzahl der Zielframes (mindestens 10) aus Dichte und Videolänge berechnet. Der Schieberegler liegt außerhalb der Bilderliste, weil Listen-Elemente Maus-Events von Schieberegler blockieren würden. Unter dem Schieberegler stehen die berechneten Zielframes („247 frames“) und die Videolänge („1m23s video“). Tooltip warnt: „Doubling the density doubles the number of frames and increases SfM time by ~100%.“

PROSTYMI SŁOWAMI

Wenn du statt Fotos ein Video importiert hast, entscheidet dieser Schieberegler, wie viele Einzelbilder die App aus dem Video herausziehen soll. Mehr Bilder = bessere Qualität, aber linear mehr Rechenzeit. Für ein 30-Sekunden-Orbit-Video sind 5 fps (150 Bilder) ein guter Anfang; bei 1-minütigen Aufnahmen reicht oft 3 fps völlig. Unter dem Regler zeigt die App live an, wie viele Frames bei der aktuellen Einstellung herauskommen — so siehst du sofort, ob du in den sinnvollen Bereich von etwa 100–300 Bildern triffst. Wenn das Ergebnis schlecht wird, zieh den Regler nach rechts und probier es nochmal; die Verdopplung der Frame-Rate verdoppelt aber auch grob die SfM-Dauer.

C-12 Clear All Button

GDZIE

Rechte Bilderliste, unten rechts; sichtbar nur wenn Bilder importiert wurden.

TECHNICZNI

Roter Button. Klick öffnet einen Bestätigungs-Dialog mit Titel „Clear all imported files?“ und Message „N images will be removed.“. Bestätigung leert alle importierten Bilder/Videos, Staging-Verzeichnisse, die Punktwolke, den Trainings-Status, das SfM-Ergebnis und alle Caches; die Stage springt zurück auf Import. Auf Cancel bleibt alles erhalten. Der Dialog ist als zerstörungsfreier Default-Pfad konfiguriert (zerstörender Button rot markiert).

PROSTYMI SŁOWAMI

Wenn du komplett neu anfangen willst, klick hier. Die Bestätigungs-Frage erscheint, weil das Löschen alle aktuellen Importe samt ggf. bereits berechneten Kameras und Trainings-Resultaten verwirft — du kannst es nicht rückgängig machen. Sinnvoll, wenn du das gewählte Bildmaterial komplett austauschen willst oder ein altes Projekt loswerden möchtest, bevor du ein neues startest. Beachte: ein einzelnes Bild rauszunehmen geht über die Liste rechts (siehe nächster Punkt), nicht über diesen Button. Deine Dateien auf der Festplatte werden dabei nicht gelöscht — die App vergisst nur ihre Referenzen.

C-13 File List ForEach (Einzel-Image-Entfernen)**GDZIE**

Rechte Bilderliste, jeder Eintrag.

**TECHNICZNIE**

Liste über die importierten Bilder mit Swipe-zum-Löschen. Pro Bild eine Zeile mit Icon, Dateiname, Auflösung („1920 × 1080“) und Dateigröße (formatiert KB/MB). Auflösung kommt aus einem Metadaten-Cache, der asynchron aus den Bild-Headern befüllt wird, damit die Oberfläche nicht blockiert. Die Löschen-Aktion bietet macOS-typisches Swipe-Delete (Trackpad-Swipe links auf einer Zeile) sowie Tastatur-Delete bei selektierter Zeile. Hinweis: Der erweiterte Image-Delete-Pfad mit explizitem Minus-Button, Backspace und Cmd-Z zum Rückgängigmachen wurde *nur im Expert-Modus* im Project Navigator ergänzt — im Einsteiger-Modus bleibt es bei Swipe-Delete.

**PROSTYMI SŁOWAMI**

Die Liste rechts zeigt jedes importierte Bild mit Auflösung und Dateigröße — praktisch, um auf einen Blick zu sehen, ob du gemischt hochauflösendes mit niedrigauflösendem Material zusammengewürfelt hast. Um ein einzelnes Bild rauszunehmen, wische es mit zwei Fingern nach links auf dem Trackpad — wie in iOS Mail — oder wähle es an und drücke Delete. Die App löscht die Datei selbst nicht; sie nimmt sie nur aus dem aktuellen Projekt. Falls du eine richtige Minus-Schaltfläche oder Cmd-Z-Rückgängig brauchst, wechsle in den Expert-Modus (Cmd+2), dort gibt es das im Project Navigator. Im Einsteiger-Modus bleibt es bewusst beim einfachen Swipe-Pattern.

C-15 Validation Warnings (3-Stufen-Tier)**GDZIE**

Unter der Bilderliste, oberhalb des Clear-All-Buttons.

**TECHNICZNI**

Drei aufeinander folgende Schwellen basierend auf der Anzahl importierter Bilder (nur aktiv wenn Bilder vorhanden und kein Video): - < 3 Bilder: rotes Banner (red octagon), Text „At least 3 images are required. Camera alignment cannot be computed from fewer images.“ - 3–9 Bilder: rotes Banner, Text „With fewer than 10 images, SfM often fails and the trained scene tends to overfit [...]. 15–20 images minimum recommended; 30+ for object captures.“ - 10–19 Bilder: orangefarbenes Banner (warning triangle), Text „Workable, but quality usually improves with 20+ images and good coverage around the scene.“

Ab 20 Bildern verschwindet das Banner. Schwellenwerte sind hartkodiert und basieren auf empirischen 560+-Trainings-Experimenten.

**PROSTYMI SŁOWAMI**

Die App schaut sich an, wie viele Bilder du importiert hast, und gibt dir eine farbige Einschätzung. Rot heißt: das wird mit hoher Wahrscheinlichkeit nichts — entweder kann SfM keine Kameron berechnen oder das Training überfittet auf zu wenig Material. Orange heißt: könnte klappen, aber rechne nicht mit Top-Qualität, weil der Algorithmus zwischen den Bildern wenig Überlapp findet. Kein Banner heißt: gute Voraussetzungen, du hast genug Material. Wenn du wirklich saubere Modelle willst, peile mindestens 30–50 gleichmäßig verteilte Aufnahmen rund um dein Motiv an — gerne auch deutlich mehr bei Außenszenen oder großen Räumen. Du kannst trotz Warnung starten, aber sei nicht überrascht, wenn SfM kommentarlos abbricht oder das Modell löchrig aussieht.

C-16 COLMAP Workspace Detection **GDZIE**

Beim Drop eines Ordners — keine sichtbare Schaltfläche, sondern Erkennungs-Logik.

 **TECHNICZNI**

Beim Drop eines Verzeichnisses wird geprüft, ob darin eine der drei kanonischen Workspace-Layouts vorliegt: `sparse/0/cameras.bin`, `sparse/cameras.bin` oder direkt `cameras.bin` im Root. Trifft das zu, wird die Standard-Bild-Enumeration abgebrochen und stattdessen ein modaler Alert geöffnet, der den User fragt, ob die bestehende Rekonstruktion verwendet oder die Bilder neu durch Apple Photogrammetry geschickt werden sollen. Gleicher Pfad auch für Text-Format-Workspaces (`cameras.txt`) und ETH3D-Exporte. Siehe Kapitel 9 Backend Q6 für Details. Wirkt im Einsteiger-Modus genauso wie im Expert-Modus.

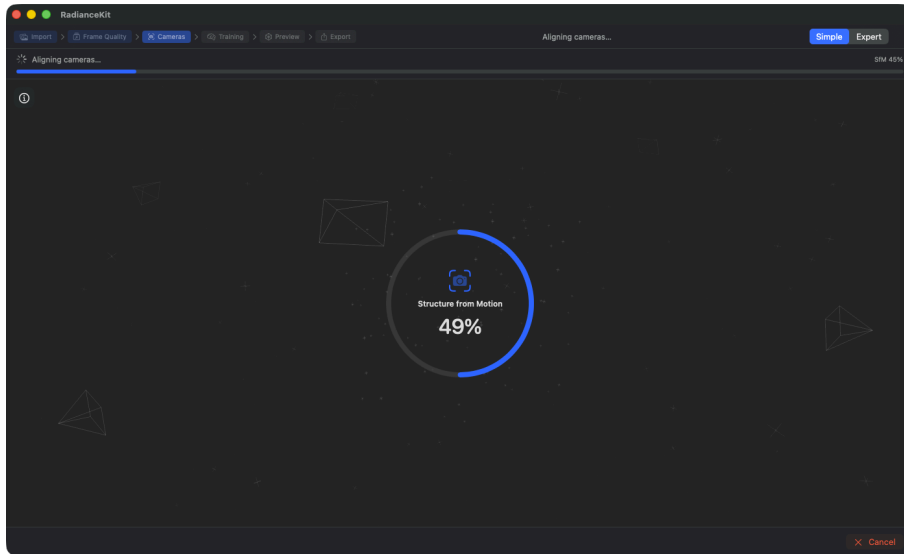
 **PROSTYMI SŁOWAMI**

Wenn du schon einmal mit MetaShape, RealityCapture oder COLMAP gearbeitet hast und dort die Kamera-Berechnung schon laufen ließest, kannst du den Export-Ordner einfach hier reinziehen. RadianceKit erkennt am Inhalt automatisch, dass es sich um ein COLMAP-Workspace handelt (es prüft auf `sparse/0/`, `cameras.bin` und Co.), und fragt dich, ob es die fertige Rechnung übernehmen oder selbst neu rechnen soll. Übernehmen spart bei großen Szenen Stunden Wartezeit, weil SfM komplett übersprungen wird — das Training startet sofort. Auch Text-Format-Workspaces (`cameras.txt`) und ETH3D-Exporte werden erkannt. Diese Funktion ist im Einsteiger-Modus genauso wie im Expert-Modus verfügbar; mehr Details stehen in Kapitel 9 unter Backend Q6.

Wann zur nächsten Stufe?

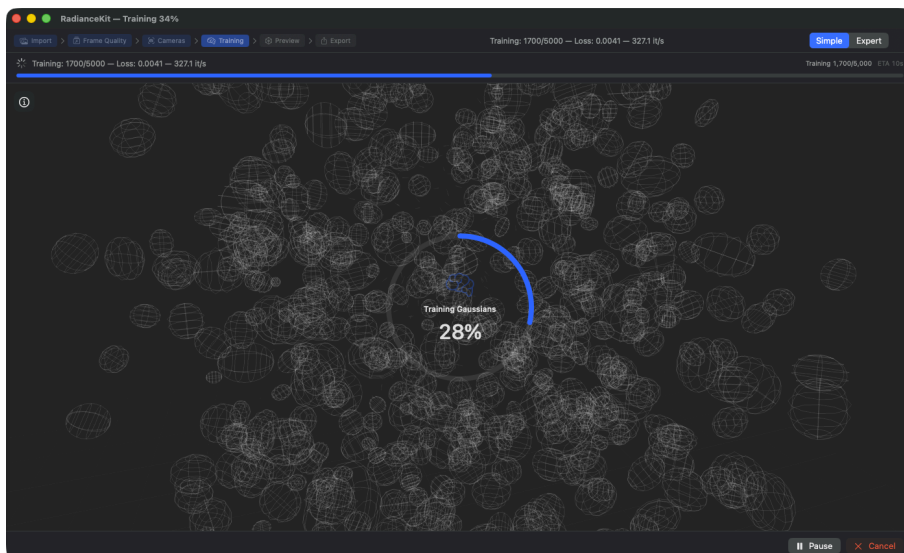
Du kannst Start Processing klicken, sobald (a) mindestens ein Bild oder ein Video importiert ist und (b) das Validation-Banner orange oder verschwunden ist. Bei rotem Banner lässt die App dich zwar trotzdem starten, du kannst aber mit hoher Wahrscheinlichkeit die Verarbeitung gleich wieder abbrechen. Empfohlen: mindestens 20 Bilder, scharf, mit deutlichem Überlapp zwischen aufeinander folgenden Aufnahmen, alle aus etwa der gleichen Distanz zum Motiv. Wähle vor dem Start ein Preset, das zu deinem Zeitbudget passt — bei 30 Bildern und Quick-Preset bist du in wenigen Minuten durch, bei Quality dauert es eher 1–2 Stunden.

Z2 — Verarbeitung (SfM + Training)



Rysunek 36: Z2 SfM-Phase — Stage-Icon „Structure from Motion“, mit 41 % im großen Kreis, oberer Statusbalken bei „SfM 25 %“, Cancel-Button unten rechts

SfM-Phase (Kameras werden ausgerichtet): Großer Fortschrittskreis zeigt Sub-Stage-Progress (hier 41 % der laufenden Apple-Photogrammetry-Session). Status-Text „Aligning cameras...“, oben links. Crumb-Trail markiert „Cameras“ als aktive Stufe. Oberer Statusbalken zeigt Pipeline-Gesamtprogress (25 %) — SfM belegt die erste Hälfte des Balkens. Schwebende Wireframe-Kameras im Hintergrund deuten an, dass Posen geschätzt werden.



Rysunek 37: Z2 Training-Phase — Stage-Icon „Training Gaussians“, mit 6 %, Live-Metriken oben (Training: 400/5000 — Loss: 0.1642 — 138.7 it/s), ETA 33s, Pause/Cancel unten

Training-Phase (Gaussians werden optimiert): Sub-Stage-Icon wechselt zu „Training Gaussians“, Prozent zählt Iterationen vom gewählten Preset (hier 400 / 5 000 für Preview-Preset = 8 % der Stage). Live-Metrik-Zeile zeigt Loss-Wert (0.1642), Iterationen-pro-Sekunde (138.7 it/s) und ETA (33 s). Pipeline-Gesamtprogress klettert von 50

% bis 100 % während dieser Phase. Pause-Button (statt Cancel-Only in SfM-Phase) erlaubt Resume später; Cancel verwirft das Training-Ergebnis und kehrt zurück nach Z1.

Sobald die Pipeline läuft, blendet die App das Import-Overlay aus und zeigt einen vollflächigen Verarbeitungsbildschirm. Mittig läuft ein großer Fortschrittskreis (220 × 220 Pixel) mit Stage-Icon, Status-Text und Prozent-Zahl; im Hintergrund visualisiert eine dezente Splat-Animation symbolisch die laufende Berechnung. Oben links lässt sich ein Info-Panel einblenden, das Live-Metriken aus Training und SfM zeigt. Unten gibt es Pause/Resume, Cancel und im Fehlerfall einen Retry-Button.

C-18 SplatTrainingView (Hintergrund-Animation)

GDZIE

Vollflächiger Hintergrund hinter dem Progress-Kreis, ausgeblendet bei Abbruch oder Fehler.

TECHNICZNIÉ

Dekorative Animation, die je nach Pipeline-Fortschritt (0...1) eine zunehmende Anzahl kleiner animierter Splat-Partikel rendert. Die Quelle ist ein berechneter Fortschrittswert, der SfM-Phasen auf 0–0.2 abbildet und Training auf 0.2–1.0 (Frame-Quality auf 0–0.05). Damit „bauen„ sich die Splats sichtbar auf, während das Training läuft. Ausschließlich dekorativ — die Anzeige zeigt keine echten Zwischenergebnisse des aktuellen Trainings (das wäre Live-Preview im Expert-Modus). Bei Cancel oder Failure wird sie ausgeblendet und nur der Status-Kreis bleibt sichtbar.

PROSTYMI SŁOWAMI

Im Hintergrund läuft eine kleine Animation aus tanzenden Punkten, damit der Bildschirm nicht so leer wirkt während der Berechnung. Das ist nicht dein echtes 3D-Modell — das siehst du erst nach dem Training in Schritt Z3. Die Animation hat aber dieselbe Tonalität, sodass du am ungefähren Verdichtungsgrad ablesen kannst, wie weit das Training fortgeschritten ist. Anfangs sind nur wenige Punkte sichtbar, gegen Ende füllt sich der Hintergrund deutlich dichter — ein hübscher visueller Indikator zusätzlich zur Prozent-Anzeige im Kreis. Wenn dich die Animation stört (z. B. weil du im Hintergrund nebenher arbeiten willst), kannst du in den Expert-Modus wechseln, wo sie wegfällt.

C-19 Großer Progress-Kreis GDZIE

Mittig auf dem Verarbeitungsbildschirm, 220 × 220 Pixel.

 TECHNICZNIÉ

Zwei übereinander gerenderte Ringe: außen ein gedämpfter Track-Ring, innen ein gefüllter Fortschritts-Ring mit Akzent- oder Rot-Stroke (rot bei Fehler). Innerhalb des Kreises ein Stage-Icon (Gehirn für Training, Kamera für SfM, Film für Video-Frame-Extraktion, Sparkles für Frame-Quality), Stage-Titel und die live animierte Prozent-Zahl in 32-Punkt-Rounded-Font. Das Icon pulsiert sanft, solange die Verarbeitung aktiv ist. Die Anzeige interpoliert auf einem 30-Hz-Timer sanft in Richtung des aktuellen echten Fortschritts — mit Constant-Creep (0.0003/Frame) plus Proportional-Anteil (4 % des Gaps) und einem Soft-Ceiling, das auf 80 % des nächsten erwarteten Milestone setzt (für SfM aus einer hartkodierten Milestone-Tabelle). So wirkt der Fortschritt flüssig, selbst wenn die echten SfM-Updates nur alle paar Sekunden eintreffen.

 PROSTYMI SŁOWAMI

Der große Kreis in der Mitte ist deine Haupt-Anzeige während die App rechnet. Er füllt sich sanft, auch wenn die echten Berechnungs-Updates nur alle paar Sekunden kommen — das gibt dir das Gefühl, dass etwas passiert, statt minutenlang auf ein eingefrorenes Prozent zu starren. Das Symbol in der Mitte wechselt, je nachdem ob gerade Frames extrahiert werden (Film-Icon), Kameras ausgerichtet werden (Kamera-Icon), oder Gaussians trainiert werden (Gehirn-Icon). Die Prozent-Zahl bezieht sich auf den aktuellen Teilschritt — die Gesamt-Pipeline siehst du in der schmalen Leiste ganz oben. Bei einem Fehler färbt sich der Ring rot statt blau, und das Icon pulsiert nicht mehr, sodass du sofort merkst, dass etwas schiefgelaufen ist.

C-22 Info Button (Metriken einblenden) GDZIE

Oben links auf dem Verarbeitungsbildschirm, 32 × 32 Pixel.

 TECHNICZNIÉ

Schlichter Button mit Material-Hintergrund. Schaltet das Info-Panel ein oder aus. Icon wechselt zwischen Info-Kreis-Outline und Info-Kreis-Gefüllt, wenn aktiv. Sanfte Einblend-Animation. Im Tooltip „Show detailed processing metrics“.

 PROSTYMI SŁOWAMI

Standardmäßig ist der Bildschirm bewusst aufgeräumt — nur der große Fortschrittskreis, mehr siehst du erstmal nicht. Wenn du als technisch interessierter Nutzer genauer wissen willst, was passiert (welche Iteration, wie hoch der Loss, wie viele Gaussians), klick auf das i-Symbol oben links. Ein kleines Panel klappt unten auf und zeigt alle Live-Werte. Ein erneuter Klick blendet es wieder aus. Die Einstellung ist nicht persistent — bei jedem neuen Trainingslauf ist das Panel zunächst wieder ausgeblendet, was bewusst so gewählt ist, um Einsteiger nicht zu erschrecken.

C-23 Info Panel (Live-Metriken) **GDZIE**

Unten links auf dem Verarbeitungsbildschirm, sichtbar nur wenn `showProcessingInfo == true`.

 **TECHNICZNI**

Zwei-spaltiges Panel mit Ultra-Thin-Material-Background. Linke Spalte: stage-spezifische Info-Zeilen — für SfM Status-Text und Prozent; für Training Iteration, kombinierter Loss, L1-Loss, D-SSIM-Loss, Gaussian-Count (orange gefärbt), Speed (it/s), Elapsed-Time, berechnete ETA, SH-Degree und Learning-Rate. Rechte Spalte: Status-Text, Time-Info-String, inline Loss-Chart (siehe C-28) und ein Discoverability-Nudge (siehe C-32). Alle Werte werden aus dem Trainings-Status gelesen, der bei jedem Trainings-Tick aktualisiert wird.

 **PROSTYMI SŁOWAMI**

Das Info-Panel zeigt alle Live-Werte, die im Expert-Modus dauerhaft in der Inspector-Sidebar stehen würden: aktuelle Iteration, Loss-Wert (kleiner = besser), Anzahl der Gaussians, Geschwindigkeit, geschätzte Restzeit, SH-Degree und Learning-Rate. Auf der rechten Seite läuft zusätzlich eine winzige Loss-Kurve mit, die dir auf einen Blick verrät, ob das Training in die richtige Richtung läuft. Wenn das Training zäh wirkt, hilft ein Blick hier — ein Loss, der nicht mehr fällt, oder eine ETA, die nicht mehr sinkt, deuten auf Probleme hin. Wenn der Loss explodiert (plötzlich riesig wird) oder NaN anzeigt, ist das Training instabil geworden und ein Cancel + Retry oder Wechsel auf ein anderes Preset ist sinnvoll.

C-25 Pause/Resume Button **GDZIE**

Untere Navigationsleiste, sichtbar nur während der Trainings-Stage (NICHT während SfM) und solange die Verarbeitung läuft.

 **TECHNICZNI**

Bordered Button. Ruft je nach Status Pause oder Resume auf. Label wechselt zwischen „Pause„ (mit Pause-Icon) und „Resume“ (Play-Icon). Während des SfM-Schritts wird der Button nicht gezeigt, weil Apple Photogrammetry keine Pause-Semantik kennt. Der Pause-Zustand erhält Iteration, Gaussian-Status und Optimizer-Momentum komplett — Resume macht da weiter, wo zuvor angehalten wurde.

 **PROSTYMI SŁOWAMI**

Während das Training läuft, kannst du es jederzeit anhalten und später fortsetzen. Sinnvoll, wenn du zwischendurch etwas anderes auf dem Mac machen willst, das viel GPU braucht — z. B. Videoschnitt, Spieletest oder ein Render-Export aus einer anderen App. Klick Pause, mach dein Ding, klick Resume, das Training läuft genau dort weiter, wo es war. Iterations-Zähler, Gaussian-Anzahl und Optimizer-Momentum bleiben dabei vollständig erhalten, der Pause-State kostet dich nichts an Qualität. Während der SfM-Phase ist Pause nicht verfügbar — Apple Photogrammetry kennt keine Anhalte-Funktion, da musst du im Notfall mit Cancel arbeiten.

C-26 Cancel Button **GDZIE**

Untere Navigationsleiste, sichtbar während die Verarbeitung läuft (SfM oder Training).

 **TECHNICZNI**

Roter Bordered-Button. Öffnet einen Bestätigungs-Dialog mit Titel „Stop and discard progress?“, Buttons „Discard Progress“ (zerstörend) und „Keep Running“ (Cancel). Auf Bestätigung wird das Cancel-Flag gesetzt, der Trainings-Task beendet, der SfM-Subprozess wenn nötig beendet und eine Summary-Zeile mit Abbruch-Status in das JSONL-Log geschrieben. Im Gegensatz zu Pause werden Trainings-Buffer und Status verworfen.

 **PROSTYMI SŁOWAMI**

Der Abbruch-Knopf. Im Gegensatz zu Pause ist das endgültig — wenn du danach neu starten willst, läuft die Verarbeitung von vorne, sämtliche bereits trainierten Iterationen sind verloren. Sinnvoll, wenn du dich beim Preset vertan hast, das Training viel zu langsam läuft, oder die App offensichtlich Müll-Ergebnisse produziert und du nicht warten willst. Vor dem tatsächlichen Abbruch fragt die App über einen Bestätigungs-Dialog nochmal nach, damit du nicht aus Versehen Stunden Rechenzeit verlierst. Falls du nur kurz unterbrechen willst, nimm lieber Pause.

C-27 Retry Button **GDZIE**

Untere Navigationsleiste, sichtbar wenn die Pipeline fehlgeschlagen ist (SfM-Status startet mit „SfM failed“, oder Training ist im Fehler-Zustand).

 **TECHNICZNI**

Akzent-Button. Startet die gesamte Pipeline neu. Vor dem Start wird geprüft, ob noch importierte Bilder/Videos vorhanden sind. Vorherige Fehler-Logs bleiben im JSONL-Verzeichnis erhalten; ein neuer Run schreibt eine neue Logdatei mit aktuellem Timestamp.

 **PROSTYMI SŁOWAMI**

Falls SfM oder Training mit einer Fehlermeldung abbricht, kannst du es hier neu versuchen. Manchmal hilft das, weil viele Schritte (RANSAC, Den-sification) Zufallsanteile haben und ein zweiter Anlauf erfolgreich sein kann, wo der erste gescheitert ist. Die gesamte Pipeline läuft dann erneut von vorne durch — SfM und Training, in einer frischen JSONL-Log-Datei. Wenn auch der zweite Versuch fehlschlägt, sind meistens die Eingangsbilder das Problem (zu wenige, zu wenig Überlapp, Bewegungsunschärfe, schlechtes Licht); geh dann zurück mit Back und tausch dein Material aus. Tipp: schau parallel in die Training-Logs (Help → Open Training Logs), dort steht detaillierter, an welcher Stelle es konkret hakte.

C-28 Inline Loss Chart**GDZIE**

Im Info-Panel, rechte Spalte, sichtbar nur während Training mit nicht-leerer Verlaufs-Historie.

**TECHNICZNI**

Kompakter Zeichen-Bereich (40 Pixel hoch), zeichnet die Loss-History als 1-Pixel-Linie in Akzent-Farbe. Daten werden auf finite Werte gefiltert (NaN-Schutz für un stabile Trainings). Min/Max werden über die gesamte History berechnet — der Chart auto-zoomt also auf den Wertebereich. Der letzte Loss-Wert steht rechts oben über dem Chart. Die History selbst wird im App-Zustand bei jedem Trainings-Tick aufgebaut (typisch alle 100 Iterationen).

**PROSTYMI SŁOWAMI**

Eine winzige Loss-Kurve, die dir auf einen Blick zeigt, ob das Training „konvergiert“, (Linie fällt nach rechts) oder ob es feststeckt oder explodiert (Linie flach oder steigt). Bei einem gesunden Training fällt die Linie zu Beginn steil und flacht dann ab — das ist der erwartete Verlauf, ähnlich einer Halbierungskurve. Der Chart zoomt automatisch auf den aktuellen Wertebereich, sodass auch kleine Verbesserungen am Ende des Trainings sichtbar bleiben. Wenn die Linie plötzlich nach oben schießt oder einfriert, ist das ein gutes Signal, dass etwas schief läuft — entweder das Material ist problematisch oder ein anderes Preset wäre besser geeignet. Den Chart findest du im Info-Panel, das du oben links mit dem i-Symbol einblendest.

C-32 Discoverability Nudge (Expert-Mode-Hinweis) **GDZIE**

Im Info-Panel, rechte Spalte unten, sichtbar nur während Training UND im Einsteiger-Modus.

 **TECHNICZNI**

Kleine Zeile mit Augen-Icon und Caption-Text „Switch to Expert Mode (⌘2) for live splat preview“, in zurückhaltender Tönung und 10-Punkt-Schrift. Kein interaktives Element, nur Hinweis. Reagiert nicht auf Klick — der User muss tatsächlich Cmd+2 drücken oder das Menü Mode → Expert Mode anklicken.

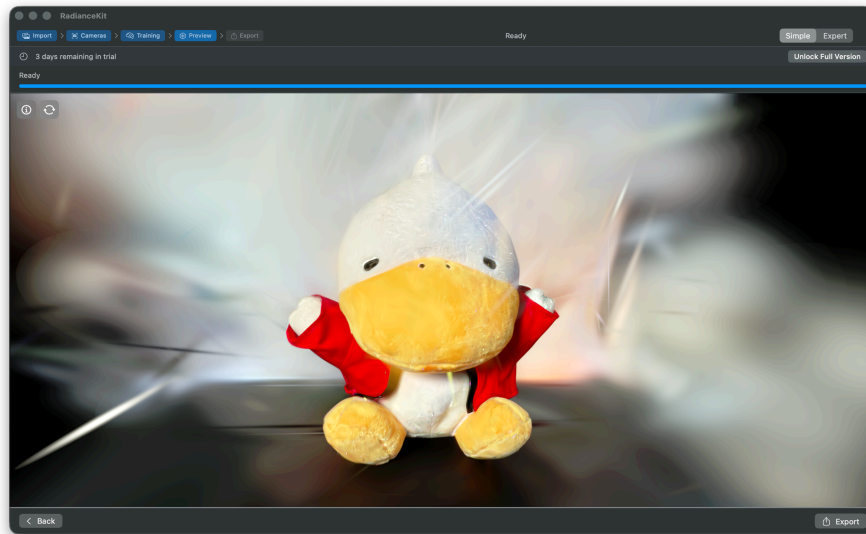
 **PROSTYMI SŁOWAMI**

Ein dezenter Hinweis, dass im Expert-Modus während des Trainings die aktuelle Zwischenversion deines 3D-Modells live im Viewport zu sehen ist. Im Einsteiger-Modus wird das absichtlich ausgeblendet, um die Oberfläche ruhig zu halten — aber viele User wissen gar nicht, dass es diese Funktion gibt, also weisen wir hier sanft darauf hin. Drücke Cmd+2 und das Training läuft im Hintergrund weiter, während du dabei zuschauen kannst, wie sich dein Modell vor deinen Augen zusammensetzt. Das ist auch ein gutes Werkzeug, um schon nach wenigen tausend Iterationen abzuschätzen, ob das Ergebnis was wird, oder ob du lieber abbrichst und neu anfängst. Cmd+1 bringt dich jederzeit wieder zurück in die Einsteiger-Ansicht.

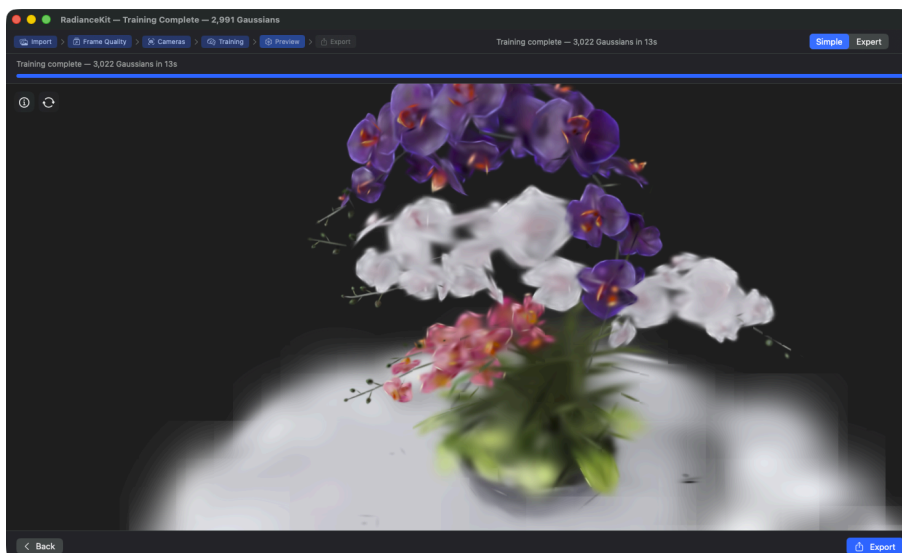
Wann zur nächsten Stufe?

Die App wechselt automatisch in Z3 (Vorschau), sobald das Training erfolgreich abgeschlossen ist — du musst nichts klicken. Die untere Navigationsleiste wechselt dann von Pause/Cancel auf einen Back-Button (zurück zu Import) und einen Export-Button (vorwärts zu Export). Im Fehlerfall (rote Fehlermeldung, Stage-Icon ist X) erscheint stattdessen Retry, und du musst entscheiden, ob du nochmal startest oder mit Back zum Import zurückgehst, um Bildmaterial zu ändern.

Z3 — Vorschau (3D-Modell drehen)



Rysunek 38: Simple-Modus Preview-Schritt mit 3D-Viewer



Rysunek 39: Z3 Preview nach Training-Abschluss — Bjoerns Blender-Bouquet rekonstruiert, Header zeigt „Training complete — 3,022 Gaussians in 13s“, Back- und Export-Buttons unten

WAS IM BILD ZU SEHEN IST Crumb-Trail markiert „Preview“, als aktive Stufe. Vollflächiger 3D-Viewport rendert die fertig trainierte Bouquet-Szene (synthetisches Blender-Test-Set von Bjoern, 60-Frame-Subset aus 960 hemisphärischen Cams). Header-Statusbal-ken: „Training complete — 3 022 Gaussians in 13 s“ — gibt finale Gaussian-Anzahl und Trainingszeit. Drag im Viewport rotiert die Kamera (Yaw/Pitch); Scroll-Wheel zoomt entlang der View-Direction. „Back“-Button (unten links) kehrt zurück zu Z2 für Resume oder Re-Run; „Export“-Button (unten rechts, primary) navigiert weiter zu Z4.

Nach Abschluss des Trainings landet die App automatisch in der Vorschau. Hier siehst du dein fertiges Gaussian-Splatting-Modell in einer Fullscreen-Metal-Ansicht und kannst es mit Maus und Trackpad drehen, zoomen und schwenken. Auf der Oberseite des

Viewports liegt ein kleines Overlay mit Kamerasteuerung und Info — Auto-Rotation, Trainings-Statistik, Reset-Knopf. Vor dem nächsten Schritt (Export) bietet es sich an, das Modell aus verschiedenen Winkeln zu prüfen, um sicherzugehen, dass die Rekonstruktion sauber ist.

C-36 SplatViewportView (3D-Hauptansicht)



GDZIE

Fullscreen-Hintergrund des Vorschau-Schritts.



TECHNICZNI

Metal-basierter 3D-Viewport, der die fertige Punktwolke rendert. Renderer to WŁASNY rasteryzator ForwardPass RadianceKit — ten sam, który renderuje splaty już podczas treningu — więc jest to prawdziwe WYSIWYG (to, co zostaje wytrenowane, jest pokazywane i eksportowane dokładnie tak samo). Tile-basierte Rendering-Pipeline mit Order-Independent Transparency. Wenn der Renderer nicht initialisiert werden kann (z. B. weil Metal auf dem System nicht verfügbar ist), erscheint stattdessen ein schwarzer Hintergrund mit „Metal not available“-Text. Die Ansicht ignoriert die Safe-Area, sodass das Modell bis an die Fensterkante reicht.



PROSTYMI SŁOWAMI

Der Hauptviewport. Hier siehst du dein fertiges 3D-Modell aus deinen Fotos rekonstruiert, gerendert auf der GPU in Echtzeit. Klick und zieh mit der linken Maustaste, um zu drehen. Scrollrad oder Trackpad-Geste mit zwei Fingern, um zu zoomen. Rechte Maustaste oder Cmd+Drag zum Schwenken. Das Modell besteht aus zigtausenden semi-transparenten 3D-Ellipsoiden („Gaussians“), die deine Szene fotorealistisch rekonstruieren — jedes einzelne hat eine Position, Ausrichtung, Form und Farbe, die das Training gelernt hat. Im seltenen Fall, dass dein Mac kein Metal unterstützt, siehst du stattdessen einen schwarzen Hintergrund mit einer Hinweismeldung — RadianceKit braucht zwingend eine Metal-fähige GPU.

C-37 CameraControlsOverlay (Steuerungs-Overlay)**GDZIE**

Über dem Viewport, schwebend.

**TECHNICZNI**

Kompaktes UI-Overlay mit Buttons für Auto-Rotation (Turntable), Reset-Camera, Hintergrund-Auswahl (Gray/Black/White), Save-Screenshot, Toggle-Info-Panel. Bindet an die Kamera-Parameter (Distanz, Azimut, Elevation, Target, FOV) und steuert das Auto-Turntable. Während des Trainings (wenn der User im Expert-Modus den Viewport mit-laufen sehen will) zeigt das Overlay zusätzlich eine kompakte Trainings-Status-Zeile.

**PROSTYMI SŁOWAMI**

Die kleine Schwebelleiste über dem Modell. Hier startest du die Auto-Rotation (das Modell dreht sich von alleine, gut für Screenshots und kurze Demos), setzt die Kamera per Reset zurück auf die Anfangsposition (falls du dich verirrt hast), wechselst den Hintergrund (Grau für neutral, Schwarz für maximalen Kontrast, Weiß für helle Modelle), und machst direkt Screenshots, die unter /Pictures gespeichert werden. Praktisch, wenn du ein bestimmtes Detail aus einem ganz bestimmten Winkel zeigen willst, ohne extra das ganze Modell zu exportieren. Die Auto-Rotation ist auch ein guter Test, ob das Modell von allen Seiten gleich gut aussieht oder ob es eine „Schmuddelseite“, gibt, die durch fehlende Aufnahmen entstanden ist.

C-38 Export Button (Navigationsleiste)**GDZIE**

Untere Navigationsleiste in Z3.

**TECHNICZNI**

Akzent-Button mit Label „Export„ und Share-Icon. Klick triggert den Wechsel zu Z4. Vorher prüft die übergeordnete Ansicht, ob die Vollversion freigeschaltet ist — wenn nicht, wird statt der Export-Bühne die Sperr-Ansicht gezeigt (siehe U-06).

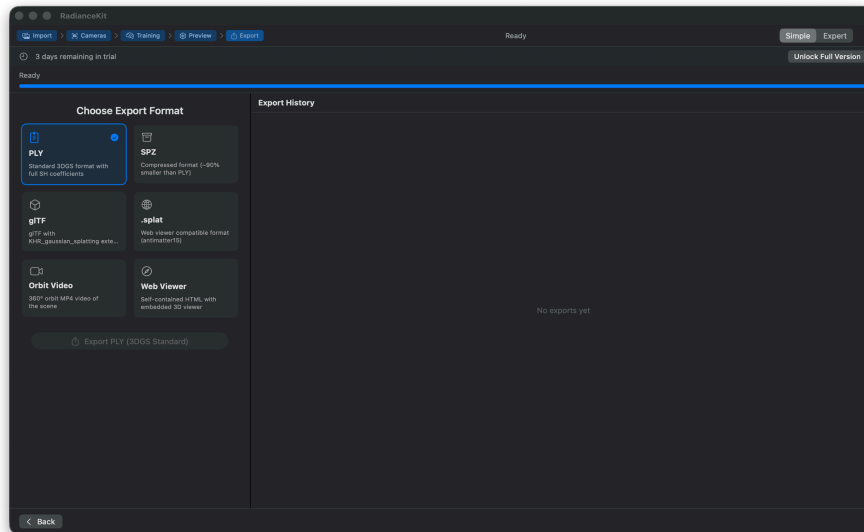
**PROSTYMI SŁOWAMI**

Wenn du mit dem Ergebnis zufrieden bist, klick Export und du landest im letzten Schritt, wo du das Format auswählst und speicherst. Ohne gekaufte Vollversion landest du stattdessen auf einer Bildschirmsperre mit Unlock-Hinweis und Kauf-Button — die App will dir keine Vollversion in die Schuhe schieben, aber der Export ist eines der Premium-Features. Sobald du den Kauf abgeschlossen hast, läuft die App direkt im freigeschalteten Zustand weiter und du landest in der gewohnten Export-Bühne. Wenn du es dir doch anders überlegst, kommst du über den Back-Button wieder in die Vorschau und kannst weiter am Modell drehen.

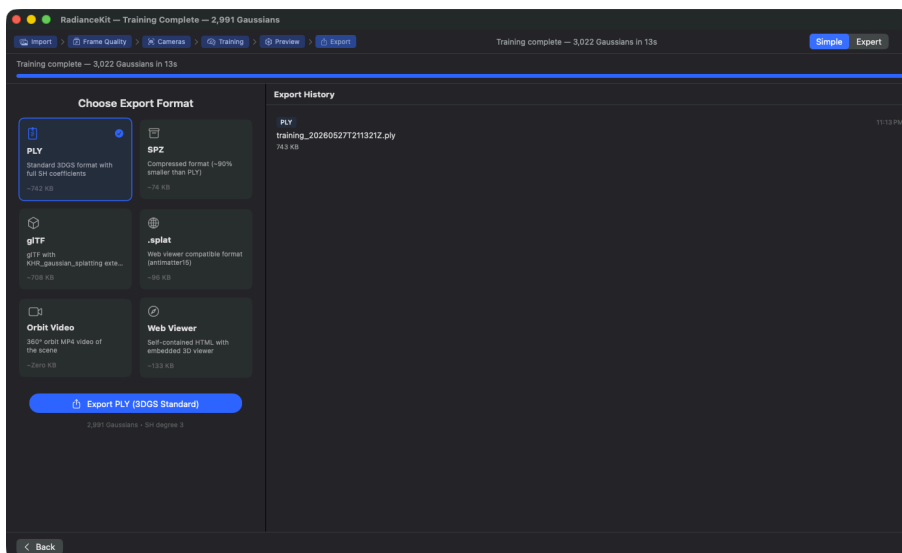
Wann zur nächsten Stufe?

Bevor du exportierst, dreh das Modell einmal komplett herum und prüfe: Sind alle Bereiche, die du in deinen Eingangsbildern abgedeckt hast, vorhanden? Gibt es schwebende „Floater„ (frei in der Luft schwebende Gauss-Splat-Wolken)? Wirkt der Hintergrund/Himmel sauber oder verschmiert? Schwerwiegende Probleme lassen sich nur durch Neu-Training fixieren — entweder mit mehr Bildern, anderem Preset, oder im Expert-Modus mit Floater-Reduction-Settings.

Z4 — Export (Format wählen & speichern)



Rysunek 40: Simple-Modus Export-Schritt mit Format-Karten



Rysunek 41: Z4 Export-Karten — 6 Formate (PLY 742 KB ausgewählt, SPZ 74 KB, glTF 708 KB, .splat 96 KB, Orbit Video, Web Viewer 133 KB), Export-History-Sidebar rechts mit bereits exportiertem PLY

WAS IM BILD ZU SEHEN IST Crumb-Trail markiert „Export“, als aktive Stufe. Linkes Karten-Grid „Choose Export Format“ mit allen sechs Optionen: PLY (Standard-3DGS, 742 KB, mit vollen SH-Koeffizienten — hier vorausgewählt mit blauem Häkchen), SPZ (komprimiertes 3DGS-Format, ~90 % kleiner als PLY, 74 KB), glTF (mit KHR_gaussian_splating-Extension, 708 KB), .splat (Web-Viewer-kompatibel via antimatter15, 96 KB), Orbit Video (360°-MP4 der Szene, Live-Größenberechnung), Web Viewer (eigenständiges HTML mit eingebettetem 3D-Viewer, 133 KB). Größenangaben werden live aus dem aktuellen Gaussian-Count und Format-Overhead berechnet. Rechts „Export History“, listet bereits abgeschlossene Exporte mit Format-Pill, Dateiname und Zeitstempel

— Click reveals im Finder. Primary-CTA unten links: „Export PLY (3DGS Standard)“ mit Gaussian-Subtitle „2,991 Gaussians · SH degree 3“.

Im letzten Schritt wählst du aus 6 Export-Formaten (PLY, SPZ, glTF, .splat, Orbit-Video, Web-Viewer) per 2-spaltiges Karten-Grid, klickst Export und wählst Speicherort im macOS-Dialog. Rechts läuft eine History aller bisherigen Exports — bei der Karten-Auswahl wird unter jeder Karte sofort der geschätzte Dateigröße angezeigt, sodass du z. B. SPZ bevorzugst, wenn du ins Web willst (klein), und PLY, wenn du in eine andere Software (SuperSplat, Postshot, Blender via Plugin) importieren willst (groß und vollständig).

C-39 2-Column Format Grid



GDZIE

Linke Hauptseite des Export-Schritts.



TECHNICZNI

Karten-Raster mit zwei flexiblen Spalten und 12 Punkt Abstand. Iteriert über die im Einsteiger-Modus angebotenen Formate — eine gefilterte Teilmenge der vollen Formatliste, die nur die 6 wichtigsten Formate enthält: PLY, SPZ, glTF, .splat, Orbit-Video, Web-Viewer. Compressed-PLY und SOG werden NUR im Expert-Modus angeboten.



PROSTYMI SŁOWAMI

Ein Karten-Raster mit den 6 Formaten, die im Einsteiger-Modus relevant sind: PLY (Standardformat für andere 3D-Tools), SPZ (komprimierte Variante für Web), glTF (offizieller Web3D-Standard), .splat (für antimatter15-Web-Viewer), Orbit-Video (fertiges MP4 zum Vorzeigen), und Web Viewer (eigenständige HTML-Datei mit eingebettetem 3D-Player). Damit deckst du 90 % der Anwendungsfälle ab. Wenn du eines der weniger gängigen Formate brauchst (Compressed-PLY oder SOG für extreme Komprimierung), wechsle in den Expert-Modus, dort sind alle 8 Formate verfügbar. Die kompakte Auswahl hier ist bewusst, damit Einsteiger nicht von der Vielfalt überfordert werden.

C-40 Format Card Button GDZIE

Jede Karte im Grid.

 TECHNICZNIE

Schlichter Button mit Karten-Layout: Icon (z. B. Dokument-Zipper für PLY, Archivbox für SPZ, Video-Icon für Orbit-Video) oben, Format-Name als Headline, Beschreibungs-Caption (2-zeilig gekürzt), darunter die geschätzte Dateigröße (live aus Format, Gaussian-Count und SH-Degree berechnet und als KB/MB formatiert). Bei Klick wird das Format ausgewählt. Selektierte Karte bekommt Akzent-Hintergrund, Akzent-Border und ein Häkchen-Icon rechts oben. Tooltip ist die Format-Beschreibung.

 PROSTYMI SŁOWAMI

Eine Karte pro Format. Klick eine an, sie wird mit Akzent-Farbe und einem Häkchen hervorgehoben, und der Export-Button darunter passt seinen Text an („Export PLY“, „Export SPZ“, etc.). Jede Karte zeigt ein passendes Symbol, den Namen, eine zweizeilige Kurz-Erklärung und die geschätzte Dateigröße bei deinem aktuellen Trainings-Ergebnis. Die Größe hilft dir, sinnvoll zu wählen — wenn du das Ergebnis per Email schicken willst, nimm die kleinste Variante (meist SPZ oder .splat); wenn du in einer anderen 3D-Software weiterarbeiten willst, nimm die mit der besten Kompatibilität (typischerweise PLY). Beim Hovern über einer Karte zeigt der Tooltip eine ausführlichere Beschreibung, falls du die Unterschiede zwischen den Formaten unklar findest.

C-41 Video Duration Slider GDZIE

Unter dem Format-Grid, sichtbar nur wenn ein Video-Format gewählt ist (Orbit-Video oder Social-Video).

 TECHNICZNIE

Schieberegler 3–30 Sekunden in 1-Sekunden-Schritten, bindet an die Video-Länge im App-Zustand. Maximal-Breite 300 Pixel. Wird nur eingeblendet, wenn ein Video-Format ausgewählt ist. Bei nicht-Video-Formaten wird der Schieberegler komplett aus der Ansicht entfernt — kein toter Platz.

 PROSTYMI SŁOWAMI

Wenn du ein Orbit-Video als Export wählst, kannst du hier die Länge bestimmen. 3 Sekunden = sehr schnelle Drehung, 30 Sekunden = langsame, ruhige Drehung um dein Modell. Für Social-Media-Reels (Instagram, TikTok) ist meistens 6–10 Sekunden ideal — lang genug, um das Modell zu zeigen, kurz genug, dass die Zuschauer nicht abspringen. Bei Präsentationen oder Portfolio-Videos darfst du gerne 15–20 Sekunden nehmen. Der Slider taucht nur auf, wenn ein Video-Format ausgewählt ist; bei Datei-Formaten wie PLY oder SPZ wäre er sinnlos und ist ausgeblendet.

C-42 Export Button **GDZIE**

Unter dem Format-Grid (und unter dem Duration-Slider, falls Video gewählt).

 **TECHNICZNI**

Großer Akzent-Button. Label: „Export {Format-Name}“, Share-Icon. Bei Klick wird der macOS-Speichern-Dialog mit Format-passender Endung und Default-Filename „scene.{ext}“ geöffnet; bei Bestätigung wird der Export an die gewählte URL geschrieben. Deaktiviert, wenn kein Trainings-Ergebnis vorhanden ist oder ein Export bereits läuft.

 **PROSTYMI SŁOWAMI**

Klick, wähle Speicherort im macOS-Dialog, fertig — die App schreibt die Datei im gewählten Format an die ausgewählte Stelle. Default-Name ist „scene.{Endung}“, (z. B. „scene.ply“ oder „scene.spz“), den kannst du im Dialog beliebig ändern, bevor du speicherst. Der Button ist grau, solange noch kein Training-Ergebnis vorhanden ist (sollte hier nie passieren, weil du sonst gar nicht im Export-Schritt wärst) oder ein anderer Export bereits läuft. Sobald der Export läuft, erscheint darunter eine Fortschrittsanzeige; die App bleibt bedienbar, du kannst also schon den nächsten Export vorbereiten.

C-43 Export Progress Bar **GDZIE**

Unter dem Export-Button, sichtbar nur während ein Export läuft.

 **TECHNICZNI**

Fortschrittsanzeige mit Max-Breite 300 Pixel, darunter Caption „Exporting... N %“. Der Wert läuft von 0 bis 1 und wird während des Schreibens aktualisiert — bei PLY in Chunks von 10 000 Gaussians, bei SPZ einmalig nach Quantisierung, bei Orbit-Video in Frame-Intervallen.

 **PROSTYMI SŁOWAMI**

Während der Export läuft, siehst du hier den Fortschritt als schmalen Balken plus Prozent-Anzeige. PLY ist meistens binnen Sekunden fertig, weil die Datei einfach binär weggeschrieben wird. SPZ braucht etwas länger, weil die Daten dabei quantisiert und komprimiert werden. Orbit-Video ist der zeitaufwendigste Export — hier wird jedes einzelne Frame neu gerendert; je nach Auflösung und Länge kann das eine Minute oder länger dauern. Während des Exports bleibt die App bedienbar, du kannst also schon das nächste Format vorbereiten oder im Viewport weiterklicken.

C-44 Export Error Display**GDZIE**

Unter der Progress-Bar, sichtbar nur wenn beim letzten Export ein Fehler aufgetreten ist.

**TECHNICZNI**

Rote Zeile mit Warning-Icon und Fehlertext. Rote 8 %-Hintergrund-Opacity, abgerundete Ecken. Max-Breite 400 Pixel. Häufige Fehlerursachen: SOG erwartet `cwebp` im System-PATH (nicht App-Store-konform); Schreibfehler bei vollem Plattenspeicher; Sandbox-Fehler bei Speicherzielen außerhalb des erlaubten Bereichs.

**PROSTYMI SŁOWAMI**

Wenn der Export schiefgeht, erscheint hier in Rot eine kurze Klartext-Beschreibung des Problems. Meistens ist die Ursache offensichtlich — kein Platz auf der Platte, keine Schreibrechte für den Zielordner, oder ein Zielort außerhalb der Sandbox-erlaubten Bereiche. Speziell beim SOG-Format kommt es vor, dass `cwebp` im System fehlt; in diesem Fall ist SOG nicht nutzbar und du musst auf SPZ ausweichen. Wenn die Fehlermeldung unklar ist, schau in das Log-Verzeichnis (Help → Open Training Logs), dort steht ausführlicher, was schiefgegangen ist. Im Zweifel hilft es, einfach einen anderen Speicherort zu wählen — z. B. den Schreibtisch.

C-46 Export History List**GDZIE**

Rechte Seite des Export-Schritts.

**TECHNICZNI**

Liste über die Export-Historie (persistent als JSON in den UserDefaults gespeichert, nach jedem erfolgreichen Export gepflegt). Jede Zeile zeigt Format-Badge (klein, akzentfarbig), Timestamp (HH:mm), Dateiname (1 Zeile gekürzt) und formatierte Dateigröße. Klick auf eine Zeile öffnet Finder mit selektierter Datei. Empty-State: „No exports yet,,“.

**PROSTYMI SŁOWAMI**

Eine Liste deiner bisherigen Exports — Format, Uhrzeit, Dateiname, Größe, in chronologischer Reihenfolge. Klick eine Zeile an und die Datei wird im Finder hervorgehoben angezeigt, ohne dass du selber durch Ordner navigieren musst. Praktisch, wenn du eine Stunde später nochmal den letzten Export brauchst und nicht mehr weißt, wo du ihn hing gespeichert hast — die History merkt sich das. Wenn du noch nie etwas exportiert hast, steht hier ein freundlicher Hinweis „No exports yet,,“. Die Liste übersteht Neustarts der App, weil sie in den UserDefaults gespeichert ist.

C-48 History Context Menu (Rechtsklick)



GDZIE

Rechtsklick auf eine History-Zeile.



TECHNICZNIE

Kontextmenü auf jedem Listen-Eintrag mit zwei Aktionen: „Reveal in Finder,“ (öffnet Finder mit selektierter Datei, wie der Einfach-Klick) und „Copy Path“ (legt den vollen Dateipfad als Text in die Zwischenablage). Letzteres ist nützlich für Drag-and-Drop in andere Apps oder zur Übergabe an die Kommandozeile.



PROSTYMI SŁOWAMI

Rechtsklick auf einen History-Eintrag öffnet ein kleines Menü mit zwei Aktionen. „Reveal in Finder,“ macht dasselbe wie ein normaler Klick — öffnet den Finder mit selektierter Datei, sodass du sie sofort siehst. „Copy Path“ legt den kompletten Dateipfad in die Zwischenablage, sodass du ihn z. B. in Terminal-Befehlen, in andere Apps oder in eine Notiz einfügen kannst. Besonders praktisch, wenn du den Export an jemanden weitergeben willst oder ihn in einem anderen Programm öffnen möchtest, das per Pfad-Eingabe arbeitet. Funktional ein kleines, aber hilfreiches Detail, das auf Mac-typische Bedienungs-Patterns setzt.

Wann ist der Workflow abgeschlossen?

Nach einem erfolgreichen Export hast du dein 3D-Modell als Datei auf der Platte und die History zeigt einen neuen Eintrag. Es gibt keinen „Done,“-Button — du kannst beliebig viele Exports in unterschiedlichen Formaten anhängen, ohne neu zu trainieren. Wenn du zurück zur Vorschau willst (z. B. um nochmal eine Kameraperspektive zu prüfen), nutze den Back-Button in der unteren Navigationsleiste. Wenn du eine komplett neue Szene anfangen willst, gehe via Back bis Z1 und nutze dort Clear All, oder File → New Project (Cmd+⇧+N).

Wechsel zu Expert-Modus

Drücke jederzeit Cmd+2 oder wähle Mode → Expert Mode (M8). Der gesamte Zustand bleibt erhalten: importierte Bilder, gewähltes Preset, laufendes oder fertiges Training, fertige Punktwolke, Export-History, sogar die aktuelle Stage. Im Expert-Modus wird statt der vier-Schritt-Bühne das volle Inspector-Sidebar gezeigt mit allen ~150 Bedienelementen. Insbesondere: der Project Navigator (siehe Kapitel 2) bietet die erweiterten Bild-Operationen (Minus-Button, Backspace-Delete, Cmd-Z-Undo, Quick-Look-Vorschau), die Live-Preview im Viewport während des Trainings, sowie alle Loss-, MCMC-, Densification- und Mip-Splatting-Parameter. Cmd+1 schaltet zurück in den Einsteiger-Modus — auch das verliert keinen Zustand.

Häufige Fragen

Warum bleibt mein Start-Processing-Button grau?

Du hast noch keine Bilder oder kein Video importiert. Zieh mindestens eine Datei in die Drop-Zone oder benutze „Browse Files,“. Sobald die Bilder-Liste rechts mindestens einen Eintrag enthält, wird der Button aktiv. (Bei nur 1–2 Bildern startet er zwar, aber SfM bricht direkt mit Fehler ab — siehe das rote Validation-Banner.)

Warum ist mein Export-Button gesperrt?

Im Einsteiger-Modus gibt es zwei Stufen: (a) Wenn die Trainings-Pipeline noch nicht fertig ist und du keine hast, ist der Button deaktiviert — du musst erst Z2 abschließen. (b) Wenn du die Vollversion noch nicht gekauft hast (`PurchaseManager.hasAccess == false`), siehst du statt der Export-Bühne eine Sperr-Ansicht mit Schloss-Icon und „Unlock Full Version,“-Button, der das Purchase-Sheet öffnet. Quick- und Preview-Presets erlauben Training kostenlos, aber Export ist Premium.

Warum kann ich kein Preset auswählen?

Du kannst es auswählen — aber wenn du ein Premium-Preset (Balanced, Quality, MCMC-Varianten) ohne gekaufte Vollversion antippst, springt der Picker automatisch zurück auf Preview und das Purchase-Sheet öffnet sich. Quick und Preview sind die einzigen kostenlos benutzbaren Presets.

Warum ist meine Drop-Zone leer und gestrichelt-grau, obwohl ich Bilder reinziehe?

Wahrscheinlich ein UTI-Typ-Mismatch. Die App akzeptiert JPG, PNG, TIFF, HEIC, MP4, MOV plus die App-eigenen Splat-Formate. Andere Bildformate (BMP, GIF, WebP, RAW-Formate) werden NICHT erkannt. Wenn du sicher bist, dass dein Bildtyp dabei sein müsste, prüfe die Dateinamens-Endung — die App geht primär nach Extension, nicht nach Datei-Inhalt.

Warum dauert SfM so lange, obwohl ich nur 30 Bilder habe?

Apple Photogrammetry skaliert nicht linear — bei manchen Bild-Konstellationen (Innenräume mit komplexen Texturen, Bewegungsunschärfe, schlechtes Licht) braucht es deutlich länger als das Bild-Count vermuten lässt. Wenn SfM nach 10+ Minuten bei 30 Bildern noch hängt, breche ab und versuche es nochmal mit besserem Material, oder wechsele in den Expert-Modus und probiere COLMAP/Native-SfM (Cmd+2 → Inspector → Camera Alignment).

Wo finde ich meine Training-Logs?

Help → Open Training Logs (Cmd+⇧+L). Das öffnet `~/Documents/RadianceKit/Logs/`. Jede Trainings-Session schreibt eine eigene JSONL-Datei mit Timestamp im Dateinamen — erste Zeile ist die Konfiguration, danach folgt eine Progress-Zeile alle 100 Iterationen, letzte Zeile ist die Summary mit Final-Loss und Success-Flag.



KOLOFON

*Złożono w SF Pro · Kod w SF Mono · Typst 0.14 ·
22. June 2026*

© 2026 Bjoern Kindler · Bischofshofener Str. 9, 82008 Unterhaching, Niemcy

Made with ❤️ in Unterhaching