



RADIANCEKIT

# Uživatelská příručka

Fotorealistická 3D rekonstrukce  
pomocí Gaussian Splatting

---

Verze 1.5.0 · macOS 26.0+ · květen 2026

BJOERN KINDLER · KINDLER-DEV.DE

# Přehled

---

Úvod — Co byste měli vědět .....	3
Co je RadianceKit? .....	3
Co je Gaussian Splatting? .....	3
Kapitola 1 — Řádek nabídek .....	5
Menu File .....	5
Menu Mode .....	9
Menu Training .....	10
Menu Viewport .....	14
Menu Export .....	19
Menu Help .....	24
Poznámka: Cmd-Z v menu Edit .....	28
Přehled klávesových zkratk .....	29
Kapitola 2 — Inspektor (Expert View) .....	30
Sekce Look (L1–L5) .....	33
Sekce Předvolby (I1–I11) .....	36
Sekce Konfigurace tréninku (I12–I22) .....	42
Sekce Enhancements (I26–I29, I42–I44) .....	48
Sekce Metriky (I30–I38) .....	55
Sekce Loss Diagram (I39–I41) .....	60
Kdy sáhnout po Inspektoru? .....	63
Kapitola 3 — Nastavení .....	65
Záložka General .....	66
Záložka AI Helpers .....	71
Settings zrcadlené z Inspektoru .....	74
Kdy co? .....	75
Kapitola 4 — Pomocná okna .....	76
User Guide (W1–W4) .....	77
Keyboard Shortcuts (W5–W6) .....	80
Manage Storage (W7–W12) .....	82
Pareto Dashboard (W13–W22) .....	86
Holdout Analysis (W23–W29) .....	92
BayesOpt Console (W30–W39) .....	97
Hlavní okno: průběh ztráty a Gaussian Count (I39–I41, křížový odkaz) .....	103
Pravidlo box .....	104
Kapitola 6 — Konfigurace tréninku .....	105
Iterace (T1–T2) .....	107
Learning Rates (T3–T10) .....	109

Densifikace — Classic (T11–T16) .....	114
Loss (T17–T20) .....	118
Progrese SH stupně (T21) .....	120
Výkon (T22–T25) .....	121
Diagnostika a příprava mraku bodů (T26–T30) .....	123
Regularizace (T31–T37) .....	126
Refinement (T38–T44) .....	129
Sky-Dome (T45–T48) .....	133
Adam + LR schedule (T49–T55) .....	135
Post-processing + Apple AI (T56–T60) .....	138
MCMC densifikace (T61–T73) .....	141
Mip-Splatting (Q1.5) (T74–T76) .....	147
Adaptivní densifikace (Q5) (T77–T79) .....	149
Curriculum (Q6) (T80–T81) .....	151
Statické předvolby (TP1–TP9) .....	151
Metoda: resolveMcmcMaxGaussians .....	154
Které pole na co? (Cheat-sheet) .....	155
Nebezpečná pole .....	155
Kapitola 7 — Vestavěné kvalitativní předvolby .....	157
Kdy kterou předvolbu? .....	167
Rychlé srovnání .....	168
Vlastní předvolby .....	170
Kapitola 8 — Formáty exportu .....	171
Který formát kdy? .....	184
Rychlé srovnání .....	185
Kapitola 9 — SfM Backendy .....	186
Který backend kdy? .....	191
Rychlé srovnání .....	192
Kapitola 10 — Režim pro začátečníky .....	193
Z1 — Import (vybrat obrázky a předvolbu) .....	193
Z2 — Zpracování (SfM + trénink) .....	201
Z3 — Náhled (otáčet 3D modelem) .....	207
Z4 — Export (vybrat formát a uložit) .....	210
Přepnutí do Expert Mode .....	215
Časté otázky .....	215

# Jak číst tuto příručku

Každý záznam v této příručce má stejné schéma. Na levé straně jsou uvedeny cesty k ovládání a technické detaily, vpravo v teplém postranním panelu najdeš vždy jedno-  
duché vysvětlení. Malé ikony na začátku řádku ti na první pohled prozradí, jaký typ informace následuje.

## ČTYŘI IKONY



**Kde to najdu?** Konkrétní cesta klikání aplikací — řádek nabídek, sekce Inspektoru nebo krok režimu pro začátečníky. Najdeš zde i příslušné klávesové zkratky. Ikona je špendlík na mapě a ukazuje, kde se funkce v uživatelském rozhraní nachází.



**Detaily.** Výchozí hodnoty, rozsahy hodnot a cesty v kódu. Setkáš se s tím především u trénovacích nastavení, která nejsou položkou menu, ale číselným parametrem. Ikona zobrazuje malou kartu se specifikací.



**Technicky.** Co funkce interně dělá, jaké parametry mají vliv, na co reaguje a jaké má vedlejší účinky. Pro čtenáře, kteří chtějí pochopit, co se děje v zákulisí. Ikona je blok posuvníků a symbolicky představuje šroubky pod kapotou.



**Jednoduše řečeno.** Hlavní myšlenka v jasných slovech — bez odborného žargonu, bez kódu. Tuto sekci si přečti jako první, pokud chceš jen rychle zjistit, k čemu funkce slouží a kdy ji potřebuješ. Ikona je řečová bublina a znamená „shrnutí v pár slovech,“. Tento sloupec má vždy teplé pískové pozadí, aby ho oko okamžitě našlo.

## BARVY KAPITOL

Každá kapitola má svou vlastní akcentovou barvu, kterou poznáš podle ID-značky (na-  
příklad **M1**) vlevo vedle nadpisu každého záznamu a podle malých ikon před ním. Při listování tak okamžitě vidíš, ve které kapitole se právě nacházíš.

- |                    |                    |                    |                                 |                  |
|--------------------|--------------------|--------------------|---------------------------------|------------------|
| <b>1</b> Menu      | <b>2</b> Inspektor | <b>3</b> Nastavení | <b>4</b> Pomocná okna           | <b>6</b> Trénink |
| <b>7</b> Předvolby | <b>8</b> Exporty   | <b>9</b> SfM       | <b>10</b> Režim pro začátečníky |                  |

**Rychlý start.** Pokud tě zajímá jen ovládání aplikace, přejdi přímo na **Kapitolu 10 — Režim pro začátečníky**. Je to provázená varianta se čtyřmi kroky a nevyžaduje žádné předchozí znalosti.

**Hlubší vstup.** **Kapitola 2 — Inspektor** a **Kapitola 7 — Předvolby** vysvětlují ovládací prvky a přednastavené kvalitativní profily, které máš k dispozici v expertním režimu.

**Vyhledávání.** Obsah a celotextové vyhledávání v PDF pomohou najít konkrétní funkci. Příručku nemusíš číst od začátku do konce.

# Úvod — Co byste měli vědět

---

## Co je RadianceKit?

RadianceKit je nativní aplikace pro macOS, která z řady běžných fotografií nebo videa vytvoří průchozí 3D rekonstrukci. Vstupem je například 50 až 500 snímků, které jsi pořídil kolem nějakého objektu, v místnosti nebo nad krajinou. Výstupem je takzvaná Gaussian-Splatting scéna — 3D model, který si můžeš na Macu v reálném čase prohlédnout z libovolného úhlu, který lze exportovat a vložit na webové stránky a který v hlavních aspektech vypadá fotorealisticky.

Aplikace běží zcela lokálně na tvém Macu — žádné obrázky se nenahrávají do cloudu, není požadováno žádné přihlášení a žádné předplatné. Aplikace intenzivně využívá GPU tvého Macu s Apple Silicon (řada M): kompletní trénink může v závislosti na scéně a předvolbě trvat od dvou minut do několika hodin. Během výpočtu můžeš na Macu zcela normálně pokračovat v práci, RadianceKit běží na pozadí a ozve se, jakmile je výsledek hotový.

Existují dva režimy ovládání: *Režim pro začátečníky* (Simple Mode) tě ve čtyřech krocích provede pracovním postupem Import → Výběr předvolby → Trénink → Export. *Expertní režim* (Expert Mode) otevře velký Inspektor se všemi ovládacími prvky, oknem živého náhledu a diagnostickými grafy. Mezi režimy můžeš kdykoli přepínat; data ve scéně přitom zůstanou zachována.

## Co je Gaussian Splatting?

Gaussian Splatting (často zkracovaně 3DGS nebo jednoduše *Splatting*) je relativně nová metoda pro fotorealistické 3D zobrazení, představená v roce 2023 v paperu z Grazu a INRIA. Myšlenka: namísto modelování scény jako klasické polygonální sítě (trojúhelníky) nebo jako voxelové mřížky se scéna skládá z milionů malých měkkých 3D obláčků — každý jednotlivý obláček je trojrozměrné Gaussovo rozdělení (odtud název) s vlastní pozicí, velikostí, tvarem, barvou a průhledností. Tyto obláčky se trénují tak, aby ze všech úhlů pohledu tvých vstupních fotografií dohromady vytvářely správný obraz.

V praxi to znamená: Gaussian Splatting dokáže zobrazit odrazy, světelné odlesky, jemné listoví, vlasy nebo záclony tak, jak to klasické 3D modelování neumí nebo jen s

ohromným úsilím. Výměnou za to není výsledek editovatelným 3D modelem v klasickém slova smyslu — nemůžeš jednoduše posunout jednu stěnu nebo přemístit vázu. Je to spíše *zamrzlý záznam* prostoru, kterým se můžeš volně pohybovat. Pro mnoho aplikací — architektonickou vizualizaci, prezentaci produktů, virtuální prohlídky, forenzní vědy, kulturní dědictví — je právě to přesně ta správná silná stránka.

Aby ze vstupních obrázků vznikla 3D scéna, jsou potřeba dva kroky. Nejprve aplikace pomocí postupu zvaného *Structure-from-Motion (SfM)* vypočítá, kde stála kamera u každé fotografie. Přitom zároveň vzniká hrubý mrak bodů scény. Pak začíná samotný Gaussian-Splatting trénink: vycházejí z tohoto hrubého mraku se miliony 3D obláčků postupně rozprostírají, zvětšují, zjemňují a upravují v pozici a barvě, dokud ze všech úhlů vstupního pohledu nedávají odpovídající obraz.

Z obojího nemusíš nic vědět, abys mohl RadianceKit používat. Režim pro začátečníky tyto kroky úplně skrývá. Ale pokud chceš porozumět tomu, co znamenají diagnostická čísla v expertním režimu (iterace, loss, Gaussiany, SSIM ...) nebo proč některé scény vyjdou krásnější než jiné, najdeš odpovědi v pozdějších kapitolách příručky.

## KAPITOLA

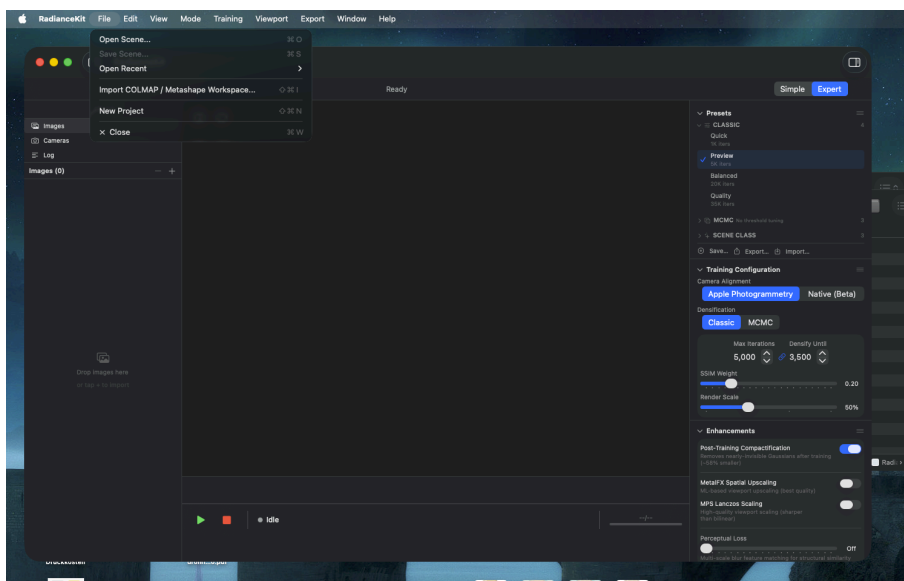
# Kapitola 1 — Řádek nabídek

Řádek nabídek RadianceKitu sdružuje všechny funkce, které neleží bezprostředně v hlavním okně nebo v Inspektoru. V první řadě to jsou akce, které působí na celou scénu (otevírání, ukládání, nový projekt), řídí trénink (start, pauza, pokračování), ovládají náhled (auto rotace, snímek obrazovky, barva pozadí) a spouštějí exporty do různých 3D a mediálních formátů. K tomu se přidávají odkazy do všech pomocných oken (User Guide, Pareto Dashboard, Holdout Analysis, BayesOpt Console).

Klávesové zkratky stojí vždy vpravo od položky menu. Konvence: **⌘** znamená klávesu Command (klávesa Apple), **⇧** je Shift, **⌥** je Option (Alt) a **⌘** je Control. Příklad: **⇧⌘T** znamená Shift+Command+T. Všechny zde dokumentované zkratky jsou navíc uvedeny v samostatném přehledovém okně přes Help → Keyboard Shortcuts (**⌘/**).

Následujících 42 položek je dokumentováno v pořadí inventáře (M1–M42), seskupeno podle příslušného top-level menu. Všechny položky byly ověřeny proti aktuálnímu stavu kódu v (řádky 175–477). Žádné položky nebyly odstraněny ani překonány oproti inventáři; nová položka menu Edit (Cmd-Z pro „Remove Image,“) je přijímána přes systémový framework NSUndoManager a neobjevuje se proto v kódu RadianceKitApp (viz poznámka na konci kapitoly).

## Menu File



Obrázek 1: Menu File rozbalené — položky M1 až M6

Menu File nahrazuje standardní položku Apple „New Window“, projektově specifickými akcemi. Zahrnuje načítání/ukládání scén, dynamický seznam Recent, import workspace a tvrdý reset do prázdného stavu.

### M1 File > Open Scene...



Řádek nabídek → File → Open Scene... (⌘O).

#### TECHNICKY

Otevírá souborový dialog pro formáty bundle `RadianceScene`, `.ply`, `.splat` a `.spz`. Jednoduchý výběr, může zobrazovat soubory i adresáře (pro bundle formát). Po úspěšném výběru se cesta zapíše do seznamu Recent a scéna se asynchronně načte — předchozí je nahrazena a tréninková pipeline se inicializuje s načteným stavem. PLY/SPZ/Splat soubory jsou čteny příslušnými format loadery; bundle `.radiance_scene` je adresář s manifestem, snapshotem cloudu a výsledky SfM.

#### JEDNODUŠE ŘEČENO

Takto načteš již natrénovanou scénu zpět do aplikace. Funguje s vlastním formátem RadianceKitu i se standardními formáty PLY, SPLAT a SPZ, které generují jiné splatting programy. Použij to, např. když jsi scénu trénoval přes noc a další den chceš pokračovat nebo exportovat. Při otevření je dosavadní stav v hlavním okně nahrazen — proto si nejdříve ulož, pokud ti záleží na aktuální scéně. Cesta automaticky končí v „Open Recent“, (M3), abys k ní příště měl rychlejší přístup.

### M2 File > Save Scene...



Řádek nabídek → File → Save Scene... (⌘S).

#### TECHNICKY

Otevírá dialog Save s typem obsahu bundle `RadianceScene` a předvyplněným názvem souboru `scene.radiance_scene`. Zapisuje adresářový balík s `manifest.json`, serializovaným Gaussian Cloud (PLY snapshot) a dumpem výsledku SfM, takže po opětovném otevření funguje i Continue Training. Položka je deaktivovaná, dokud žádné Gaussiany neexistují. Neukládá do cesty tréninkových logů, ale tam, kam ukazuje dialog Save — typicky pod `~/Documents/`.

#### JEDNODUŠE ŘEČENO

Uloží tvou aktuální scénu jako soubor (přesněji: jako balíčkový adresář, který vypadá jako soubor). Teprve poté můžeš scénu později znovu otevřít přes „Open Scene...“, (M1). V balíčku končí jak Gaussian Cloud, tak výsledek SfM, takže můžeš později přidat i Continue Training (M12–M14). Dokud nemáš dokončený trénink, je položka zašedlá. Standardní název je `scene.radiance_scene` — ale v dialogu Save můžeš zadat vlastní název.

**M3 File > Open Recent > [Názvy scén]**

Řádek nabídek → File → Open Recent → (Seznam).

**TECHNICKY**

Dynamické podmenu, které je generováno ze seznamu naposled otevřených cest (uloženo v nastavení). Každá položka seznamu je pojmenována názvem souboru a při kliknutí načtena. Pokud je seznam prázdný, objeví se místo toho deaktivovaný štítek „No Recent Scenes,“. Typicky pro Apple drží seznam N naposled otevřených scén — omezení se odehrává při zápisu do nastavení a ne v builderu menu samotném.

**JEDNODUŠE ŘEČENO**

Zde vidíš naposledy otevřené scény a jedním kliknutím můžeš znovu skočit dovnitř, aniž bys musel projít souborový dialog. Pokud jsi právě začal, seznam je prázdný a šedě stojí v menu. Každá scéna, kterou otevřeš přes „Open Scene...“ (M1), v tomto seznamu automaticky přistane. Pokud ti seznam někdy bude příliš plný nebo ho chceš vyprázdnit z důvodu soukromí, použij „Clear Recent“ (M4).

**M4 File > Open Recent > Clear Recent**

Řádek nabídek → File → Open Recent → Clear Recent.

**TECHNICKY**

Vyprázdní seznam Recent v nastavení. Působí okamžitě, bez potvrzovacího dialogu. Položka se v podmenu objevuje pouze tehdy, pokud v seznamu Recent vůbec nějaké položky jsou (stojí pod dividerem za cestami).

**JEDNODUŠE ŘEČENO**

Smaže seznam naposled otevřených scén. Praktické, když sis pohrával s testovacím datasetem a cesty už nechceš vidět. Samotné soubory scén se přitom nemazou — pouze odkaz v menu. Akce působí okamžitě, bez dotazu; poté se v podmenu objeví „No Recent Scenes,“. Položka se vynořuje pouze tehdy, pokud jsou v seznamu vůbec nějaké scény — u prázdného seznamu není viditelná.

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

Řádek nabídek → File → Import COLMAP / Metashape Workspace... (⇧⌘I).

**TECHNICKY**

Otevírá výběr složky. Očekává složku s layoutem COLMAP workspace (např. `sparse/0/cameras.{bin,txt}` plus `images/`). Po výběru se provádí předkontrola workspace — ta rozpozná tři layouty (`sparse/0/`, `sparse/`, `root`) a zda je rekonstrukce binární (`cameras.bin`) nebo jako ETH3D text (`cameras.txt`). Při úspěchu se workspace importuje; jinak se v logu aplikace objeví pouze varování. Viz také Kapitulu 9 „SfM backendy“, Q6 pro plnou logiku pipeline.

**JEDNODUŠE ŘEČENO**

Pokud používáš Metashape, COLMAP, RealityCapture nebo podobný software pro rekonstrukci kamer a máš export, načteš složku zde. RadianceKit pak přeskočí fázi SfM a začne přímo s tréninkem — u velkých scén to šetří hodiny. Drag-and-drop na hlavní okno funguje stejně. Očekává se složka s layoutem COLMAP (tedy `sparse/0/s cameras.*` plus složkou `images/`). Více o podporovaných layoutech a workflow stojí v Kapitole 9 „SfM backendy“.

**M6 File > New Project**

Řádek nabídek → File → New Project (⇧⌘N).

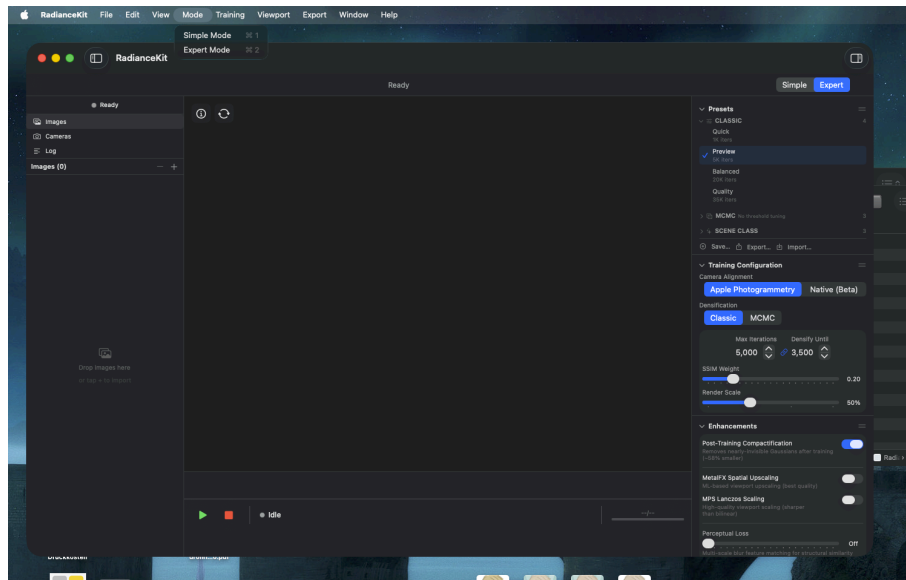
**TECHNICKY**

Kontroluje, zda existuje neuložená práce. Pokud ano, objeví se potvrzovací dialog, než se cokoli ztratí. Pokud není co ukládat, reset proběhne přímo — vyprázdní importované obrázky, výsledek SfM, Gaussian Cloud, stav tréninku a všechny závislé UI indikátory. Pozor: uživatelem založená knihovna předvoleb zůstává zachována, protože leží v nastavení aplikace a ne ve stavu projektu.

**JEDNODUŠE ŘEČENO**

Vrátí vše do prázdného startu — jako bys aplikaci právě čerstvě otevřel. Pokud máš ještě neuloženou práci, aplikace se předtím zeptá. Použij to, když chceš začít s úplně jinou scénou. Importované obrázky, výsledek SfM, Gaussian Cloud a stav tréninku se kompletně vyprázdní. Tvé vlastní předvolby ale zůstanou zachovány, protože leží v nastavení aplikace a nepatří ke scéně.

## Menu Mode



Obrázek 2: Menu Mode s přepínači Simple a Expert Mode

Dva jednoduché přepínače mezi řízeným Simple Mode (wizard, 4 kroky) a plným Expert Mode (klasický inspector layout se všemi posuvníky).

### M7 Mode > Simple Mode



Řádek nabídek → Mode → Simple Mode (⌘1).



Přepíná stav aplikace na Simple Mode. Hlavní oblast aplikace pak místo Expert layoutu zobrazuje řízený workflow. Stav Mode se ukládá v nastavení (viz S1 „Default Mode,, v Kapitole 3 Settings).

### JEDNODUŠE ŘEČENO

Přepíná na variantu krok za krokem, kde tě aplikace provede importem, zpracováním, náhledem a exportem. Doporučeno, pokud teprve začínáš nebo rychle potřebuješ výsledek. Většina detailních posuvníků je skrytá — pracuješ se smysluplnými přednastavenými hodnotami. Pokud chceš později jít hlouběji, jednoduše přepni do Expert Mode (M8). Který režim je aktivní při startu aplikace, můžeš nastavit v nastavení (Kapitola 3, S1).

## M8 Mode > Expert Mode



Řádek nabídek → Mode → Expert Mode (⌘2).

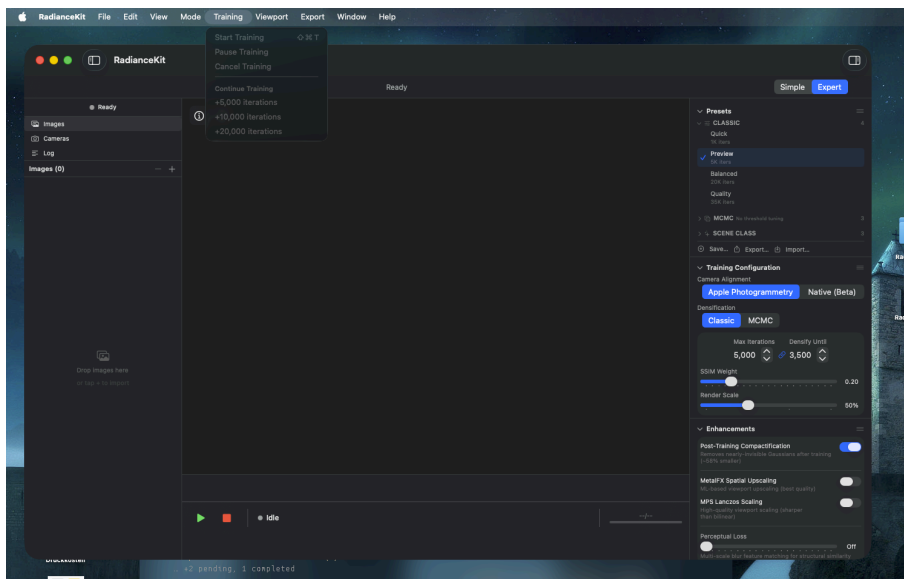


Přepíná stav aplikace na Expert Mode. Objevuje se plný inspector layout se všemi sekcemi (Presets, TrainingConfig, Enhancements, Metrics, LossChart, ProjectNavigator). V Expert Mode jsou přístupné všechny tréninkové parametry, COLMAP picker, mid-compact toggles a diagnostika. Také Live Preview funguje pouze v tomto režimu.

### JEDNODUŠE ŘEČENO

Přepíná do plného zobrazení se všemi posuvníky. Zde vidíš loss grafy v reálném čase, můžeš všechny parametry doladit a spravovat několik konfigurací pro porovnání paralelně přes předvolby. Doporučeno, pokud chceš pochopit, co trénink interně dělá, nebo pokud chceš cíleně experimentovat. Také Live Preview, COLMAP picker a diagnostika jsou přístupné jen tady. Pokud se cítíš zahlcen, jdi přes M7 zpět do Simple Mode — tvá scéna přitom zůstane zachována.

## Menu Training



Obrázek 3: Menu Training s podmenu Continue — položky M9 až M14

Čtyři akce kolem tréninkového běhu: start, pauza, zrušení a prodloužení o předem daný počet iterací. Všechny tři položky Continue jsou gatovány přes IAP (ve free trial verzi nelze klikat).

**M9 Training > Start Training**

KDE

Řádek nabídek → Training → Start Training (⇧⌘T).

## TECHNICKY

Spouští tréninkovou pipeline asynchronně. Předpoklad: existuje výsledek SfM a momentálně neběží žádná jiná pipeline. Obě podmínky položku blokují, pokud nejsou splněny. Při startu se čtou aktuální hodnoty konfigurace, zakládá se nový JSONL log pod `~/Documents/RadianceKit/Logs/training_YYYY-MM-DD_HHmss.jsonl` a podle volby strategie se jede klasická nebo MCMC cesta. Stav tréninku se mění z „idle„ na „training“.

## JEDNODUŠE ŘEČENO

Stiskne velké zelené tlačítko — jakmile máš importované fotografie a je hotová rekonstrukce kamer, tím začíná samotný Gaussian Splatting trénink. Nech aplikaci běžet; podle předvolby mezi 1 minutou (Quick) a několika hodinami (MCMC Quality). Položka zůstává šedá, dokud výsledek SfM ještě neexistuje nebo dokud běží jiná pipeline. Každý běh paralelně zapisuje log do `~/Documents/RadianceKit/Logs/`, který můžeš později vyhodnotit přes Pareto Dashboard (M40).

**M10 Training > Pause Training**

KDE

Řádek nabídek → Training → Pause Training.

## TECHNICKY

Pozastavuje běžící trénink. Aktivuje se pouze tehdy, pokud je stav tréninku „training„. Pauza zastaví iterativní smyčku na nejbližším bezpečnostním sync pointu, zachová plný GPU stav (Gaussian buffery, optimizer momenty, pozice scheduleru) a přepne na „paused“. Resume probíhá opětovným stiskem (titulek položky je statický — ale aplikace přepíná mezi pause/resume ve vlastní logice). Pozastavené tréninky nepřežijí ukončení aplikace; v tom případě místo toho ulož scénu a pozdější přidání udělej přes položku Continue Training (M12–M14).

## JEDNODUŠE ŘEČENO

Krátce zastaví trénink, aniž bys ztratil postup. Praktické, když počítač chvíli potřebuješ pro něco důležitějšího. Další klik pokračuje. Nefunguje přes restarty aplikace — pokud chceš opravdu pokračovat později, ukonči trénink přes Cancel (M11), ulož scénu pomocí Save Scene (M2) a poté použij Continue Training (M12–M14). Během pauzy GPU úplně odpočívá; paměť ale zůstává obsazená.

**M11 Training > Cancel Training**

Řádek nabídek → Training → Cancel Training.

 TECHNICKY

Ruší běžící trénink. Aktivní, pokud stav tréninku není „idle“. Nastavuje cancel flag v tréninkovém enginu, což čistě ukončí iterativní smyčku na nejbližším sync pointu, zapíše finální summary záznam do JSONL logu a vrátí stav na „idle“. Dosud trénovaný cloud zůstává zachován (lze uložit nebo exportovat), je ale označen jako „cancelled“.

 JEDNODUŠE ŘEČENO

Ukončí běžící trénink definitivně. Dosavadní stav zůstává — pokud tedy máš po pár tisících iteracích už ukazatelný výsledek, můžeš ho i poté exportovat. Pokud chceš jen krátce přerušit, použij místo toho Pause (M10). V tréninkovém logu se běh označí jako „cancelled“, finální hodnota lossu se přesto zapíše. Přerušenu scénu můžeš později pokračovat i přes Continue Training (M12–M14), dokud aplikace mezitím nebyla ukončena.

**M12 Training > Continue Training > +5 000 iterací**

Řádek nabídek → Training → Continue Training → +5,000 iterations.

 TECHNICKY

Pokračuje v tréninku o 5 000 iterací. Aktivní, pokud dokončený trénink je pokračovatelný a plná verze je odemčená. Pokračovatelnost platí, pokud existuje dokončený trénink a plný stav optimizera je ještě v paměti. Při Continue se pokračuje s Adam momenty a LR schedulerem, takže pokračování se chová jako průběžný 25K/45K/60K běh místo restartu. JSONL log dostává nový config záznam s inkrementálním setupem. Dostupné pouze v plné verzi.

 JEDNODUŠE ŘEČENO

Přidává 5 000 dalších tréninkových kroků. Použij to, když výsledek po prvním běhu je blízko cíli, ale ještě ne úplně ostrý. Funguje pouze v placené plné verzi. Na rozdíl od kompletně nového běhu zůstává stav optimizera zachován, takže pokračování působí jako průběžný běh. Pokud potřebuješ více než 5 000 kroků, vezmi rovnou M13 (+10 000) nebo M14 (+20 000).

**M13 Training > Continue Training > +10 000 iterací**

Řádek nabídek → Training → Continue Training → +10,000 iterations.



Identické s M12, ale s 10 000 dodatečnými iteracemi. Stejně předpoklady, stejná cesta LR scheduleru. Doporučeno, pokud iniciální trénink běžel se středně velkou předvolbou a chceš vidět významné zvýšení kvality, aniž bys běh úplně restartoval.

**JEDNODUŠE ŘEČENO**

Prodlužuje trénink o 10 000 kroků — střední ze tří dostupných Continue hodnot. Dobrá volba, pokud první běh byl sice v pořádku, ale chceš být ještě jasně lepší. Jako u M12 a M14 průběh learning rate plynule pokračuje místo restartu. Dostupné pouze v plné verzi.

**M14 Training > Continue Training > +20 000 iterací**

Řádek nabídek → Training → Continue Training → +20,000 iterations.

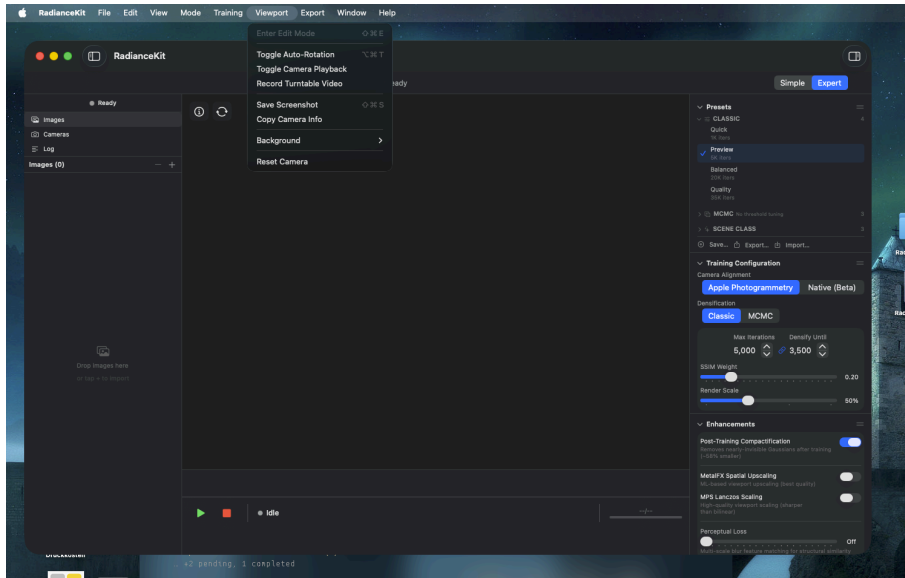


Identické s M12 / M13, ale s 20 000 dodatečnými iteracemi. Největší předdefinovaný Continue skok. U MCMC tréninků to je často to, co dělá rozdíl mezi „odpovídá“, a „benchmark-tauglich“; u Classic od 35–40K je podle zkušeností přírůstek malý.

**JEDNODUŠE ŘEČENO**

Přidává 20 000 dalších tréninkových kroků, maximální Continue hodnota. Použij to, pokud chceš opravdu vyždímat poslední kapku kvality. U klasického tréninku po 40 000 krocích to často už moc nepřinese — u MCMC se to naopak často vyplatí, protože tam konvergence nastupuje pomaleji. Plánuj podle scény s výrazně delší dobou běhu. Jako M12 a M13 je i tato položka dostupná pouze v plné verzi.

## Menu Viewport



Obrázek 4: Menu Viewport s Edit Mode, ovládáním kamery a podmenu pozadí

Řídí 3D náhled: Edit Mode pro selekci a cleanup Gaussianů, ovládání kamery (auto rotace, playback, recording), snímek obrazovky, barva pozadí a reset.

### M15 Viewport > Enter/Exit Edit Mode



Řádek nabídek → Viewport → Enter Edit Mode (nebo „Exit Edit Mode“, podle stavu). ⌘⌘E.



Titulek položky je dynamický a podle stavu ukazuje „Exit Edit Mode“, nebo „Enter Edit Mode“. Při stisku se přepíná Edit Mode v rendereru náhledu. Při opuštění Edit Modu se navíc resetuje aktuální výběr. Edit Mode aktivuje klik selekci na Gaussianech, box selekci a mazání označených Gaussianů (viz oblast editoru v UI). Deaktivováno, dokud není připojen viewport renderer.

#### JEDNODUŠE ŘEČENO

Přepíná mezi normálním 3D pohledem a režimem úprav, ve kterém můžeš označit a smazat jednotlivé Gaussiany (např. floatery nebo odlehlé body v pozadí). Při opuštění se výběr automaticky resetuje. Položka zůstává šedá, dokud v náhledu není viditelná žádná scéna. Popisek se mění podle stavu mezi „Enter Edit Mode“, a „Exit Edit Mode“ — vždy vidíš, v jakém režimu právě jsi.

**M16 Viewport > Toggle Auto-Rotation**

Řádek nabídek → Viewport → Toggle Auto-Rotation (⌘⇧T).



Přepíná spojitou rotaci kamery náhledu kolem vertikální osy středem scény. Osa a rychlost pocházejí z konfigurace ovládání kamery. Auto rotace je čistě efekt náhledu a neovlivňuje ani trénink, ani recording — pokud paralelně používáš Turntable Video Recorder (M18), auto rotace dodává přesně tu dráhu, kterou recorder zachytí.

**JEDNODUŠE ŘEČENO**

Otáčí kamerou pomalu stále kolem tvé scény, abys ji mohl vidět ze všech stran, aniž bys musel táhnout myš. Další klik rotaci zastaví. Praktické při posuzování hotových trénovaných scén nebo jako pozadí animace pro live demo. Pokud paralelně nahráváš video (M18), auto rotace dodává přesně pohyb, který recorder zachytí.

**M17 Viewport > Toggle Camera Playback**

Řádek nabídek → Viewport → Toggle Camera Playback.



Přepíná playback dráhy kamery. Pokud existuje zaznamenaná dráha kamery (např. z předchozího recordingu nebo proto, že byla načtena `transforms.json`), dráha se přehrává — kamera náhledu se už nepohybuje podle vstupu myši/trackpadu, ale reprodukuje trajektorii frame po framu. Opětovný stisk playback pauzuje.

**JEDNODUŠE ŘEČENO**

Nechá přehrát dříve zaznamenanou nebo importovanou jízdu kamery. Tak můžeš sledovat originální dráhu, kterou byla scéna pořízena, nebo zkontrolovat plánovaný orbitální pohyb před exportem videa. Během playbacku jsou vstupy myši a trackpadu deaktivovány — kamera striktně sleduje dráhu. Opětovný klik přehrávání pauzuje. Pokud jsi ne-načetl ani nezaznamenal žádnou dráhu kamery, nic se nestane.

**M18 Viewport > Record Turntable Video**

Řádek nabídek → Viewport → Record Turntable Video.

**TECHNICKY**

Přepíná nahrávání náhledu. Při prvním stisku startuje zápis framů do dočasné cesty; při druhém stisku se nahrávání ukončí, zenkóduje a zapíše do MP4 cesty (cesta je dotázána přes dialog Save). Na rozdíl od Export → Media → Orbit Video (M31), který vytváří pevnou 360° dráhu při nastavitelné délce, nahrává Turntable Recorder živě to, co vidíš v náhledu — můžeš tedy zaznamenat i ruční jízdu kamery.

**JEDNODUŠE ŘEČENO**

Nahrává video přímo v náhledu. Jedno, zda se kamera otáčí automaticky nebo zda ji posunuješ sám myší — vše, co vidíš, se ukládá do MP4 souboru. Na rozdíl od exportu „Orbit Video„ (M31) jízdu kamery zadáváš sám. První kliknutí nahrávání spustí, druhé kliknutí ho ukončí a zeptá se na místo uložení. Praktické, když chceš ukázat např. konkrétní detailní švenk, který by s rigidním orbitálním pohybem nebyl možný.

**M19 Viewport > Save Screenshot**

Řádek nabídek → Viewport → Save Screenshot (⇧⌘S).

**TECHNICKY**

Zachycuje jeden frame náhledu v plném render rozlišení (tedy ne pixel layout okna, ale plný obsah render targetu) jako PNG soubor. Cesta je dotázána přes dialog Save. Barva pozadí (M21–M23) se zapaluje do snímku. Nastavení MetalFX/MPS upscalingu z Enhancements (viz I27/I28) se uplatňují, pokud jsou aktivní — snímek tedy ukazuje upscalovaný výstup.

**JEDNODUŠE ŘEČENO**

Ukládá snímek tvého aktuálního 3D pohledu jako PNG obrázek. Praktické pro marketingový materiál nebo rychlé porovnání. Pozor: pozadí je součástí obrazu — pokud potřebuješ průhlednost, exportuj raději soubor scény. Rozlišení odpovídá internímu render targetu, ne velikosti okna — obrázek je tedy často ostřejší, než vypadá v okně. Případná nastavení upscalingu (Inspektor → Enhancements) se rovněž promítají.

**M20 Viewport > Copy Camera Info**

Řádek nabídek → Viewport → Copy Camera Info.



Čte aktuální pozici kamery náhledu (pozice, look-at bod, up vektor) a hodnoty FOV z ovládání kamery a zapisuje je jako víceřádkový text do schránky. Formát je čitelný pro lidi (label = value na řádek), ne JSON. Praktické pro reprodukci konkrétního pohledu při ladění nebo sdílení se supportem.

**JEDNODUŠE ŘEČENO**

Kopíruje aktuální pozici kamery a směr pohledu jako text do schránky. Když chceš např. ukázat spoluvývojáři, odkud to vypadá v scéně podivně, jednoduše vložíš text do mailu nebo chatu. Formát je čitelný pro lidi (jeden řádek na hodnotu), ne JSON. Hlavně určeno pro bug reporty nebo dotazy pro support.

**M21 Viewport > Background > Dark Gray**

Řádek nabídek → Viewport → Background → Dark Gray.



Nastavuje barvu pozadí náhledu na tmavou šedou (RGB 0.1/0.1/0.1). Renderer používá tuto barvu jako pozadí, před kterým jsou Gaussiany compositovány. Výchozí barvu při startu aplikace řídí volba S3 „Default Viewport Background“.

**JEDNODUŠE ŘEČENO**

Obarví pozadí 3D náhledu na tmavě šedou. Standardní volba pro většinu scén — poskytuje dobrý kontrast ke světlým i tmavým Gaussianům, aniž by se oko zaseklo na čistě černé nebo bílé ploše. Barva se přejímá i ve snímcích (M19) a Orbit videích (M31). Pokud ti je Dark Gray příliš nevýrazná, vyzkoušej pro porovnání i Black (M22) nebo White (M23). Kterou barvu je aktivní při startu aplikace, můžeš nastavit v Nastavení (S3).

**M22 Viewport > Background > Black**

Řádek nabídek → Viewport → Background → Black.



Nastavuje barvu pozadí náhledu na čistou černou (RGB 0/0/0). Pomáhá, pokud scéna obsahuje mnoho světlých floatery a chceš je identifikovat, nebo pro marketingový materiál s tmavým look and feel.

**JEDNODUŠE ŘEČENO**

Černé pozadí. Dobré pro velmi světlé scény nebo když chceš v Edit Modu zahlédnout dovnitř a hledat malé světlé Gaussiany (floatery), které se v šedé ztrácejí. Také ideální pro marketingový materiál s tmavým, dramatickým vzhledem. Barva se ve snímcích a Orbit videích zapaluje — pokud potřebuješ průhlednost pro pozdější composit, je černá nejhorší volbou. Pro tmavé floatery přepni v opačném směru na White (M23).

**M23 Viewport > Background > White**

Řádek nabídek → Viewport → Background → White.



Nastavuje barvu pozadí náhledu na čistou bílou (RGB 1/1/1). Užitečné, pokud scéna obsahuje převážně tmavý obsah a chceš vidět tmavé floatery (typický outdoorový šum pozadí).

**JEDNODUŠE ŘEČENO**

Bílé pozadí. Praktické, pokud motiv vyzní lépe světlý na tmavém, nebo pro nalezení tmavých odlehklých bodů, které poté můžeš v Edit Mode (M15) odstranit. U outdoorových scén je bílá často užitečnější než černá, protože typické outdoorové floatery jsou spíše tmavé. Jako u ostatních voleb pozadí se barva přejímá do snímků a videí.

**M24 Viewport > Reset Camera**

Řádek nabídek → Viewport → Reset Camera.

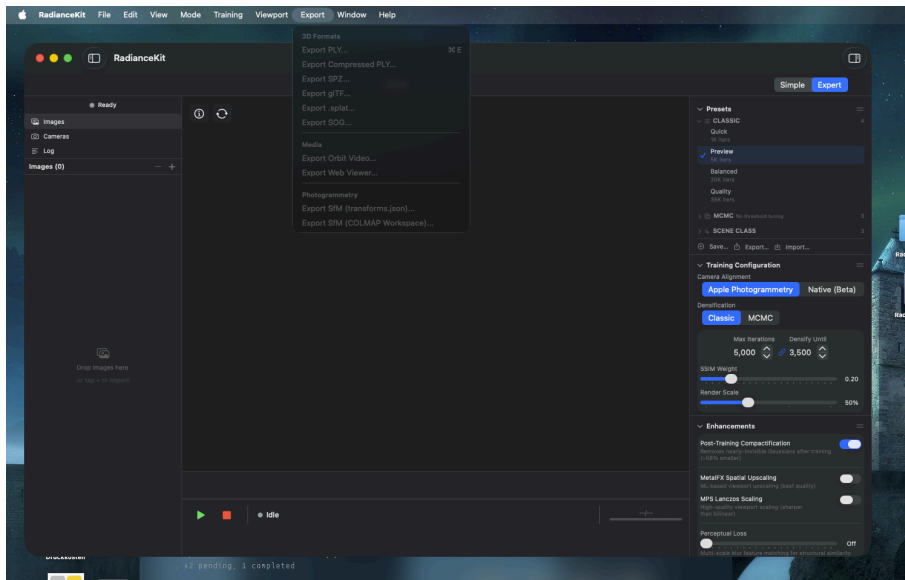


Resetuje kameru náhledu, opouští pohled training camera a zastavuje auto rotaci. Kamera je tím zpět na počáteční pozici (typicky: před scénou, mírně shora hledící), auto rotace je vypnutá a pokud renderer právě ukazoval training camera (jednu z pozic SfM), vrací se na free camera.

**JEDNODUŠE ŘEČENO**

Vrátí kameru náhledu do výchozí pozice. Pokud ses při otáčení ztratil nebo jsi scénu vystrčil z obrazu — jedním kliknutím zde a opět vidíš to, co bys měl vidět. Současně vypíná auto rotaci, pokud zrovna běží, a vrací se z zamrzlé training kamery do volného pohledu. Tak v každém případě získáš čistý restart pohledu.

## Menu Export



Obrázek 5: Menu Export se třemi skupinami podmenu — 3D Formats, Media a Photogrammetry

Osm cílů exportu plus dva fotogrammetrické exporty, seskupené ve třech sekcích (3D Formats, Media, Photogrammetry). Prvních šest se staví přes sdílenou pomocnou rutinu, která otevírá dialog Save a registruje export v katalogu formátů. Položky Photogrammetry mají individuální logiku. Všechny exporty Photogrammetry a některé 3D exporty jsou dostupné jen v plné verzi.

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



Řádek nabídek → Export → 3D Formats → PLY (⌘E).



Otevírá dialog Save s předvyplněným názvem `gaussians.ply`. Při OK se aktuální Gaussian Cloud zapíše do standardizovaného ASCII/binárního PLY formátu — kompatibilní se SuperSplat, PolyCam, PlayCanvas a všemi běžnými 3DGS prohlížeči. Plné SH koeficienty, plná přesnost (Float32 na pole). Velikost souboru často několik set MB u  $\geq 500K$  Gaussianů.

### JEDNODUŠE ŘEČENO

Uloží tvou 3D scénu jako standardní PLY soubor. To je nejuniverzálnější formát — skoro každý software ho umí načíst, od SuperSplatu přes PolyCam po PlayCanvas. Soubory ale bývají velké, často několik set megabajtů. Použij PLY, pokud chceš pokračovat v plné kvalitě nebo archívovat. Pokud chceš scénu sdílet po webu, podívej se raději na SPZ (M27) nebo Compressed PLY (M26) — jsou podstatně menší.

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

Řádek nabídek → Export → 3D Formats → Compressed PLY.



Zapíše Gaussian Cloud ve formátu Compressed PLY s vlastní kvantizací polí pozice, scale, rotation a SH. 5–10× menší soubory než nekomprimovaný PLY (M25) při minimálních vizuálních ztrátách. Kompatibilní se SuperSplat (které čte standard Compressed PLY) a PlayCanvas. Standardní název souboru `gaussians_compressed.ply`.

**JEDNODUŠE ŘEČENO**

Jako normální PLY, ale 5–10krát menší. Kvalita zůstává téměř stejná. Použij to, když chceš soubor sdílet online nebo poslat e-mailem. Funguje přímo se SuperSplat a PlayCanvas. Pokud ale cílový systém potřebuje ještě menší soubory (Mobile, browser demo), vezmi místo toho SPZ (M27) — to je ještě agresivněji komprimováno. Pro plnou editační kvalitu vezmi nekomprimovaný PLY (M25).

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

Řádek nabídek → Export → 3D Formats → SPZ.



Zapíše Gaussian Cloud ve formátu SPZ — komprimovaný splat formát publikovaný Nianticem s agresivní kvantizací (~90 % menší než nekomprimovaný PLY). Optimalizováno především pro web prohlížeče a mobilní aplikace. Kompatibilní s Niantic Splatt3R, `gsplat.js` a Niantic browser viewer.

**JEDNODUŠE ŘEČENO**

Jeden z nejmenších formátů. Asi 10× menší než normální PLY. Použij to především, když chceš scénu zobrazit v prohlížeči nebo prohlížet přes mobilní aplikaci. Pro maximální kvalitu je lepší PLY. SPZ je vyvinut Nianticem a funguje přímo s `gsplat.js`, Splatt3R a Niantic Web Viewerem. Kvůli silné kompresi nemůžeš SPZ soubory bez dalšího trénovat dál — pro editaci vezmi PLY.

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

Řádek nabídek → Export → 3D Formats → glTF.

**TECHNICKY**

Zapisuje `.gltf` soubor (binární glTF) s rozšířením `KHR_gaussian_splatting`. Standardně kompatibilní, vhodné pro pipeline, které používají glTF enginy jako Babylon.js nebo Three.js a implementují rozšíření `KHR_gaussian_splatting`.

**JEDNODUŠE ŘEČENO**

Uloží scénu ve formátu glTF, kterému rozumí mnoho 3D programů a webových enginů — za předpokladu, že podporují rozšíření Gaussian Splatting. Pokud máš specifickou 3D pipeline (např. Three.js nebo Babylon.js), která to rozumí, je to tvůj formát. Soubor vyjde jako binární `.gltf` — jediný balík, který obsahuje vše. Pro klasické splatting workflow je obvykle PLY nebo SPZ lepší volbou, protože je rozumí více nástrojů přímo.

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

Řádek nabídek → Export → 3D Formats → .splat.

**TECHNICKY**

Zapisuje antimatter15 `.splat` formát — fixed-size 32 bajtů na Gaussian (pozice jako 3× Float32, scale jako 3× Float32, rotace jako 4× Uint8 normalizovaný quaternion, RGB+opacity jako 4× Uint8). Žádné SH koeficienty vyšší než DC. Nejmenší soubor s přímou kompatibilitou pro prohlížeč. Pro `gsplat.js` a `antimatter15` online demo viewer.

**JEDNODUŠE ŘEČENO**

Nejjednodušší formát pro web prohlížeč. Malý a okamžitě zobrazitelný v každém prohlížeči. Ztrácí ale detail osvětlení (vyšší SH koeficienty se ztrácí — splat vypadá z každého úhlu pohledu stejně, místo aby reagoval na světlo). Pro maximální web výkon dobré, pro fotorealismus spíše SPZ nebo PLY. Funguje s `antimatter15` online viewerem a `gsplat.js`. Každý Gaussian zabírá fixně 32 bajtů, což činí formát jednoduchým a kompatibilním — ale za cenu hloubky detailu.

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

Řádek nabídek → Export → 3D Formats → SOG.

## TECHNICKY

Zapisuje Gaussian Cloud ve formátu SOG. SOG („Self-Organizing Gaussian“) je PlayCanvas formát s layoutem texture atlas a WebP kompresí kvantizovaných dat. Škáluje s 15–20× lepším poměrem velikosti než PLY. Export interně volá `cwebp` jako externí nástroj — proto v sandboxové variantě (App Store) potenciálně omezeno.

## JEDNODUŠE ŘEČENO

Velmi malý formát pro PlayCanvas workflow. Asi 15–20krát menší než PLY, protože data jsou zabalena do layoutu texture atlas a WebP komprimována. Pokud nemáš PlayCanvas workflow, je obvykle lepší volbou SPZ nebo Compressed PLY. Export interně volá `cwebp` jako externí nástroj — v App Store verzi (sandbox) může být tento krok omezen.

**M31** Export > Media > Export Orbit Video...

Řádek nabídek → Export → Media → Orbit Video.

## TECHNICKY

Renderuje 360° orbit kolem středu scény a enkóduje ho jako MP4 (H.264) nebo MOV (HEVC, podle systémového default). Na rozdíl od M18 (živé nahrávání) je dráha zde pevně předem daná — délka se volí v nastavení resp. v kroku Export Simple Mode.

## JEDNODUŠE ŘEČENO

Automaticky vytváří otáčivé video kolem tvé scény. Není potřeba ručního pohybu. Dobré pro sociální sítě nebo rychlé demo. Pokud chceš kameru řídit sám, použij místo toho Record Turntable Video (M18). Dráha je pevná: plný 360° orbit kolem středu scény, délku volíš v nastavení nebo v kroku Export Simple Mode. Video se podle systému vydává jako H.264 MP4 nebo HEVC MOV.

**M32** Export > Media > Export Web Viewer...

Řádek nabídek → Export → Media → Web Viewer.

**TECHNICKY**

Balí samostatný HTML prohlížeč (založený na gsplat.js) plus Gaussian data base64 kódovaná do jediného `.html` souboru. Tento soubor běží offline v každém moderním prohlížeči — žádné závislosti na serveru, žádné externí URL. Velikost souboru je přibližně faktor 1.3 větší než SPZ varianta (kvůli base64 režii).

**JEDNODUŠE ŘEČENO**

Uloží tvou scénu jako samostatně spustitelnou webovou stránku. Dvojklik na HTML soubor → prohlížeč se otevře → hotová interaktivní 3D scéna. Funguje bez internetu, lze poslat e-mailem, je to nejjednodušší způsob, jak sdílet výsledek s přáteli nebo zákazníky. Soubor obsahuje kompletní gsplat.js prohlížeč a Gaussian data v jediném dokumentu — nic se nedotahuje z webu. Velikost souboru je asi o třetinu větší než SPZ export, zato příjemce nepotřebuje žádný další software.

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

Řádek nabídek → Export → Photogrammetry → Export SfM (transforms.json).

**TECHNICKY**

Vlastní exportní cesta (ne přes sdílenou pomocnou rutinu), protože se neexportuje Gaussian Cloud, ale výsledek SfM. Otevírá dialog Save s `transforms.json` jako default a content type `json`. Při OK se zapíše nerfstudio kompatibilní `transforms.json` s intrinsics kamer, pozicemi (jako 4x4 matice v NeRF konvenci) a cestami framů. Helpový text v UI upozorňuje, že tréninkové obrázky musí být zkopírovány jako sousední složka `images/`. Aktivní jen tehdy, pokud existuje výsledek SfM a plná verze je odemčena.

**JEDNODUŠE ŘEČENO**

Pokud chceš výsledek SfM dále použít v jiném softwaru jako nerfstudio, Brush, gsplat nebo OpenSplat, exportuješ zde pozice kamer. Tvé tréninkové obrázky navíc dej do složky `images/` vedle souboru `transforms.json` — jinak cílový program nemůže obrázky přiřadit. Položka je zasedlá, dokud výsledek SfM ještě neexistuje, a ve free trial verzi uzamčena. Pro COLMAP workspace workflow vezmi místo toho M34.

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



Řádek nabídek → Export → Photogrammetry → Export SfM (COLMAP Workspace).

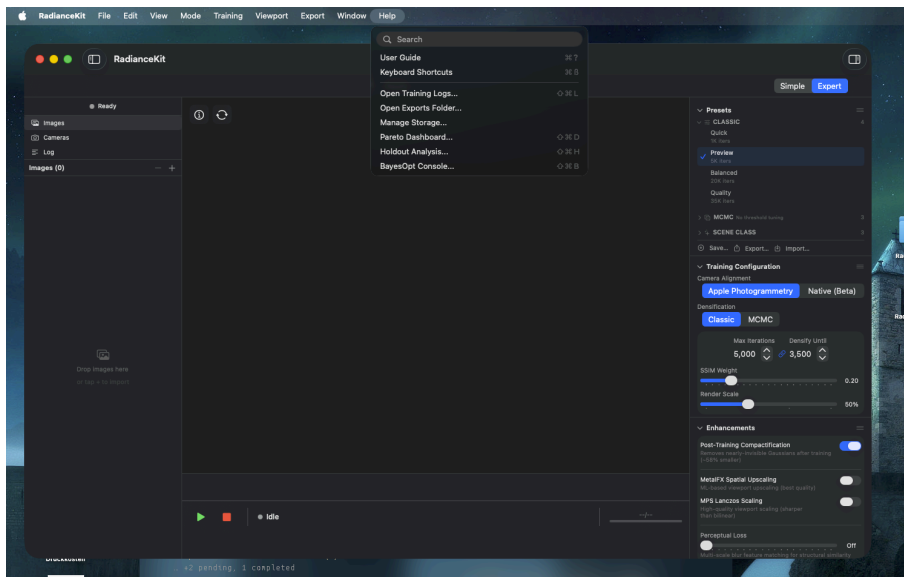


Otevírá dialog Save s předvyplněným názvem `colmap-workspace` (bez přípony, protože je to složka). Zapisuje standardní COLMAP workspace s `sparse/0/cameras.bin`, `images.bin`, `points3D.bin`. Umožňuje otevřít rekonstrukci SfM spočítanou nebo importovanou v RadianceKitu v jiných nástrojích jako Postshot, Nerfstudio nebo Meshroom, nebo při A/B re-run znovu načíst do samotného RadianceKitu (přes M5) jako už spočítaný vstup — šetří čas výpočtu. Aktivní jen tehdy, pokud existuje výsledek SfM a plná verze je odemčena.

#### JEDNODUŠE ŘEČENO

Jako M33, ale ve formátu COLMAP místo nerfstudio. Pokud používáš Postshot, Meshroom, Nerfstudio nebo jiný nástroj s COLMAP workflow, je to tvůj export. Praktický vedlejší efekt: tuto složku můžeš později přes M5 načíst zpět do RadianceKitu a ušetřit si čas výpočtu SfM při dalším běhu — zejména u velkých scén úspora hodin. Jako M33 dostupné jen tehdy, pokud existuje výsledek SfM, a ve free trial verzi uzamčeno.

## Menu Help



Obrázek 6: Menu Help s dokumentačními, složkovými a analytickými položkami

Sedm položek: dvě dokumentační okna (User Guide, Keyboard Shortcuts), tři odkazy na složky (Training Logs, Exports, Storage) a tři analytická okna (Pareto Dashboard, Holdout Analysis, BayesOpt Console). Typicky pro Apple stojí menu Help úplně vpravo. Standardní menu Help je úplně nahrazeno vlastní variantou RadianceKitu.

**M35 Help > User Guide**

Řádek nabídek → Help → User Guide (⌘?).



Otevírá okno User Guide. Ukazuje navigaci s tematickým sidebarem a oblastí scroll detailu při výchozí velikosti 860×640. Obsahy jsou staticky uložené (ne parsované z Markdownu).

**JEDNODUŠE ŘEČENO**

Otevírá průvodce vestavěného v aplikaci. Pokud nechceš všechno dohledávat v tomto manuálu, najdeš tam nejdůležitější kroky přímo v programu. Průvodce je vystavěn jako samostatné okno s tematickým sidebarem — můžeš tedy cíleně skákat na jednotlivá témata. Obsahy jsou kratší než tato příručka a soustředí se na nejčastější workflows.

**M36 Help > Keyboard Shortcuts**

Řádek nabídek → Help → Keyboard Shortcuts (⌘/).



Otevírá okno Keyboard Shortcuts — jednoduchý scroll layout se všemi klávesovými zkratkami aplikace, seskupený podle top-level menu. Výchozí velikost 440×560. Obsahy jsou rovněž staticky uloženy.

**JEDNODUŠE ŘEČENO**

Otevírá okno s kompletním seznamem všech klávesových zkratk. Pokud si např. nemůžeš zapamatovat, kterou klávesou se spouští trénink, podíváš se tam. Souhrn stojí i na konci této kapitoly. Seznam je seskupený podle top-level menu, takže rychle skočíš do správné oblasti. Užitečné, pokud se zrovna přepínáš ze stylu myš na klávesnici.

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

Řádek nabídek → Help → Open Training Logs... (⇧⌘L).



Počítá složku logu jako ~/Documents/RadianceKit/Logs, zakládá ji v případě potřeby a otevírá ji ve Finderu. Každý tréninkový běh tam zapisuje vlastní JSONL soubor `training_YYYY-MM-DD_HHmms.jsonl`.

**JEDNODUŠE ŘEČENO**

Otevírá ve Finderu složku se všemi dosavadními tréninkovými protokoly. Pokud se něco pokazilo nebo chceš nahlédnout, kdy přesně trénink konvergoval na jakou hodnotu, najdeš to zde v JSONL souborech. Na tréninkový běh se zakládá přesně jeden soubor s časovým razítkem — to můžeš také načíst do jiných nástrojů nebo poslat e-mailem do supportu. Pokud chceš grafické vyhodnocení, je Pareto Dashboard (M40) lepší vstup.

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

Řádek nabídek → Help → Open Exports Folder...



Analogicky k M37, ale s `~/Documents/RadianceKit/Exports`. Zakládá se při prvním autotest běhu nebo prvním kliknutí; poté tam přistávají standardní cesty všech autotest exportů (např. `autotest_<timestamp>.ply`). Manuálně přes dialog Save vybrané exporty NEMUSÍ nutně skončit zde, ale tam, kam uživatel uloží — proto je tato složka zajímavá především pro autotesty.

**JEDNODUŠE ŘEČENO**

Otevírá složku, ve které aplikace ukládá své vlastní exporty (především autotest běhy). Pokud jsi export manuálně přes dialog Save uložil jinak, je tam a ne v této složce. Praktické pro úklid nebo pro zjištění, kolik místa zabírají starší testovací exporty. Pokud chceš kompletní přehled včetně logů a scénových balíčků, vezmi místo toho Manage Storage (M39).

**M39 Help > Manage Storage...**

Řádek nabídek → Help → Manage Storage...



Otevírá Storage Browser (viz Kapitulu 4 Auxiliary Windows, ID W7–W12). Listuje všechny persistované scény, tréninkové logy, exporty a cache ve složce `~/Documents/RadianceKit/` s velikostí, umožňuje `reveal-in-Finder` a `move-to-trash` na položku.

**JEDNODUŠE ŘEČENO**

Otevírá oknový browser, který ti ukazuje, kolik místa RadianceKit zabírá na disku — na scénu, log a export. Můžeš přímo smazat jednotlivé věci, aniž bys musel jít do Finderu. Praktické po delším používání, když je disk plný — starší logy a autotest exporty se mohou sčítat na několik gigabajtů. Přes `reveal-in-Finder` se kdykoli dostaneš i ke klasickému pohledu.

**M40 Help > Pareto Dashboard...**

Řádek nabídek → Help → Pareto Dashboard...  
(⇧⌘D).



Otevírá Pareto Dashboard (viz Kapitulu 4, ID W13–W22). Dashboard načítá všechny JSONL tréninkové logy z `~/Documents/RadianceKit/Logs/`, řadí je podle scény a předvolby a vykresluje Pareto scatter plot (standard: loss vs Gaussiany, volitelně loss vs wall-clock nebo PSNR vs iterace).

**JEDNODUŠE ŘEČENO**

Otevírá přehled všech dosavadních tréninkových běhů jako diagram. Okamžitě vidíš, který běh dodal nejlepší rovnováhu kvality a velikosti. Praktické, pokud chceš porovnat různé předvolby. Standardně diagram ukazuje loss proti počtu Gaussianů — můžeš ale přepnout i na wallclock čas nebo PSNR. Data pocházejí z JSONL tréninkových logů (M37); čím více běhů máš, tím vypovídavější vyhodnocení bude.

**M41 Help > Holdout Analysis...**

Řádek nabídek → Help → Holdout Analysis...  
(⇧⌘H).



Otevírá okno Holdout Analysis (viz Kapitulu 4, ID W23–W29). Načítá `transforms.json`, vykresluje kamery jako 3D globus a umožňuje train/test fold splity (úhlové nebo lineární, 2–8 foldů). Výstupem je `fold-assignment.json`, kterou trénink v příslušných tréninkových konfiguracích může používat jako test set.

**JEDNODUŠE ŘEČENO**

Pomáhá ti rozložit tvé kameraové záběry do tréninkových a testovacích setů — abys objektivně mohl měřit, jak dobrá tvá scéna je (na obrázcích, které trénink neviděl). Spíše výzkumný a benchmark nástroj. Kamery jsou zobrazeny jako 3D globus; můžeš zvolit 2 až 8 foldů, buď rovnoměrně v úhlu, nebo lineárně přes pořadí. Výsledkem je malý JSON soubor, který trénink poté používá jako test set.

**M42 Help > BayesOpt Console...**

Řádek nabídek → Help → BayesOpt Console...  
(⇧⌘B).



Otevírá BayesOpt konzoli (viz Kapitulu 4, ID W30–W39). Načítá předdefinované vyhledávací prostory (např. „MCMC scale-reg + opacity-reg + ssim„), spouští Bayesian Optimization trials asynchronně a živě zobrazuje konvergenční křivku a trial log.

**JEDNODUŠE ŘEČENO**

Vestavěná auto tuner konzole. Místo aby ses ručně proklíkával různými parametry, může aplikace přes noc běžet sama a na konci ti navrhnout nejlepší hodnoty pro tvou scénu. Velmi pokročilý nástroj — pro většinu workflow je dostačující dobrá předvolba (viz Kapitulu 7). Volíš předdefinovaný vyhledávací prostor (např. „MCMC scale-reg + opacity-reg + ssim„), a živě vidíš konvergenční křivku a trial log. Plánuj podle setupu několik hodin až dní.

**Poznámka: Cmd-Z v menu Edit**

Od května 2026 podporuje Project Navigator v Expert Mode mazání importovaných obrázků přes tlačítko minus nebo klávesu Backspace a vrácení akce přes `Cmd-Z`. Tato akce `Cmd-Z` se objevuje v macOS menu Edit (které poskytuje SwiftUI) jako „Undo Remove Image“, dokud je smazaný obrázek ještě obnovitelný. Registruje se přes standardně-kompatibilní systém, ne v ; proto neexistuje vlastní položka M-ID v inventáři.

## Přehled klávesových zkratk

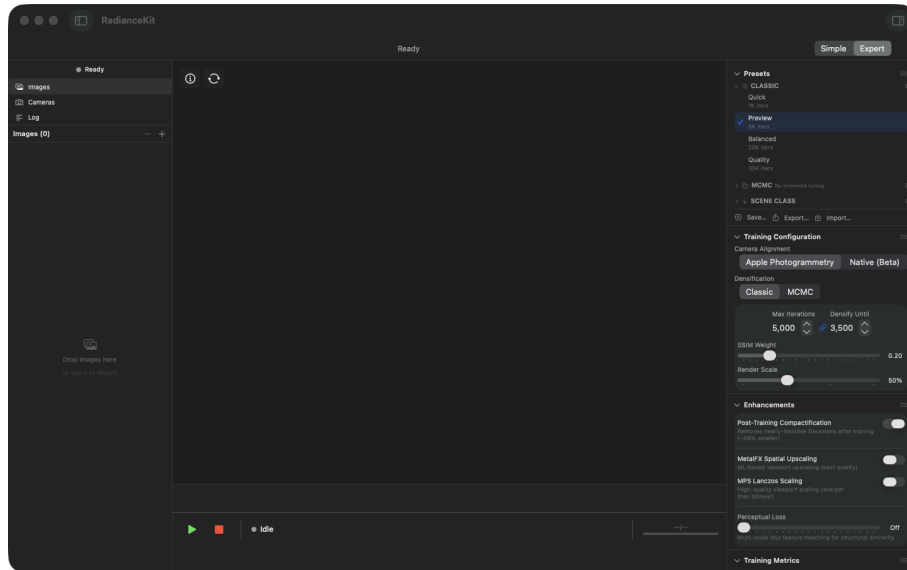
Položka menu	Zkratka
File > Open Scene...	⌘O
File > Save Scene...	⌘S
File > Import COLMAP / Metashape Workspace...	⇧⌘I
File > New Project	⇧⌘N
Mode > Simple Mode	⌘1
Mode > Expert Mode	⌘2
Training > Start Training	⇧⌘T
Viewport > Enter/Exit Edit Mode	⇧⌘E
Viewport > Toggle Auto-Rotation	⌘⌥T
Viewport > Save 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 (poskytované systémem, v Expert Mode při aktivním výběru Project Navigatoru):

Akce	Zkratka
Undo Remove Image	⌘Z
Remove Selected Image	Backspace / Delete

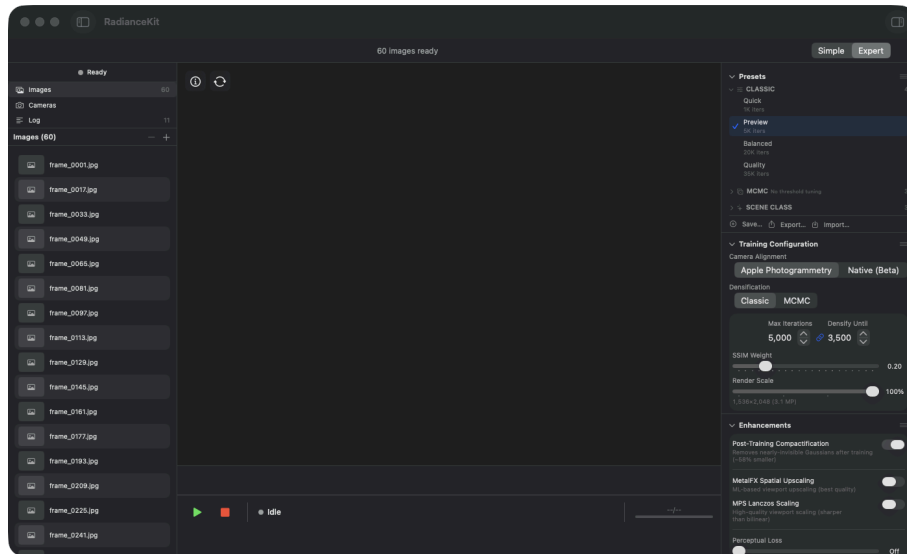
## KAPITOLA

## Kapitola 2 — Inspektor (Expert View)



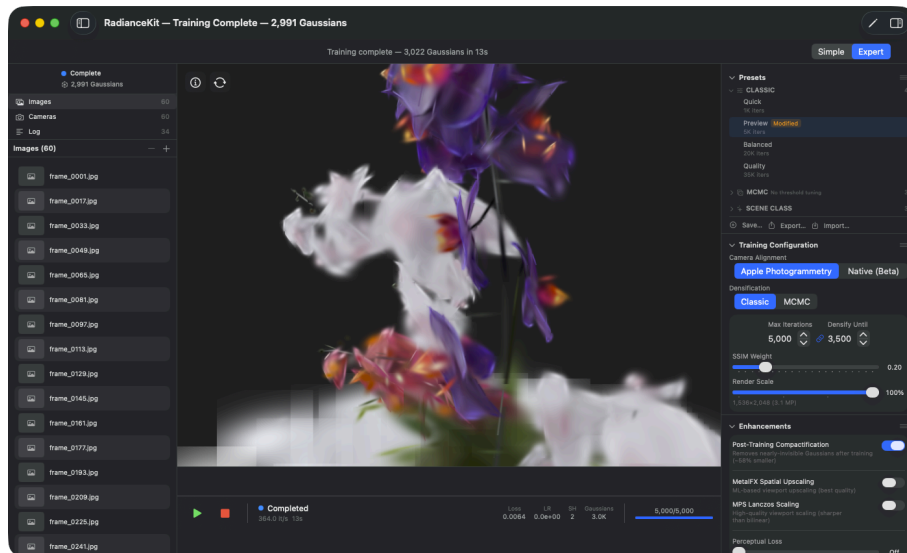
Obrázek 7: Expertní režim prázdný — Project Navigator vlevo (Images 0, Cameras, Log), prázdný náhled uprostřed, Inspektor vpravo se sekcemi Presets/Training Configuration/Enhancements/Training Metrics

**Prázdný Inspektor před importem:** Levý sidebar ukazuje Images counter 0 a drop hint „Drop images here / or tap + to import“. Inspektor vpravo je plně funkční, ale předvolby jsou pouze informativní (žádný aktivní trénink). Výchozí předvolba „Preview“ (5K iter) je označená. Camera Alignment na Apple Photogrammetry, densifikace Classic, SSIM Weight 0.20, Render Scale 50 %. Prázdné stavy v Training Metrics („Start training to see live metrics“) a Loss History („Loss curve will appear during training“).



Obrázek 8: Inspektor s 60 načtenými obrázky — Image sidebar ukazuje první názvy souborů `frame_0001.jpg` ff, hlavička „60 images ready,“

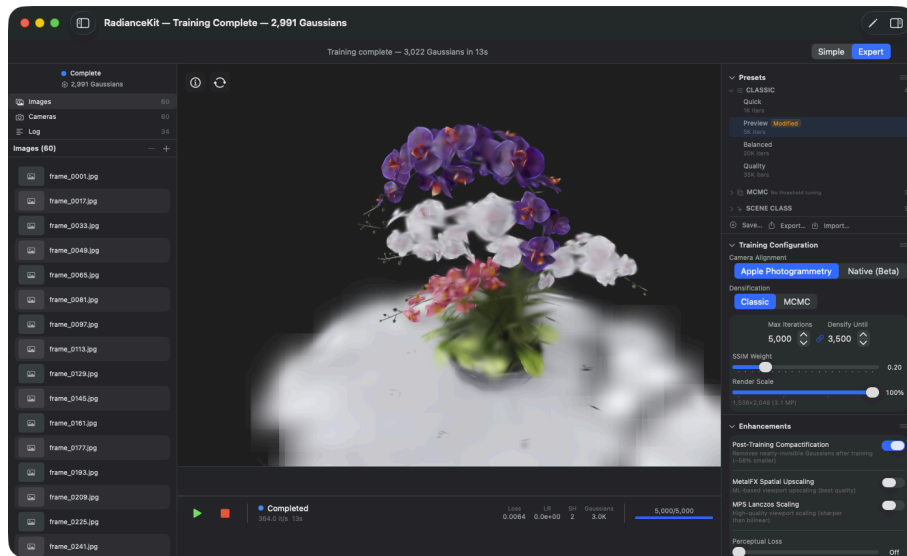
**Inspektor po importu:** Status v hlavičce „60 images ready,“. Image sidebar listuje všech 60 importovaných framů ( `frame_0001.jpg` až `frame_0945.jpg` , každý 16. frame datasetu jako podmnožina pro rychlé iterace). Logika auto render scale kontroluje rozlišení obrazu ( $1536 \times 2048 = 3.1$  MP) a tomu odpovídajícím způsobem upravuje Render Scale. Play tlačítko (zelené, vlevo dole) je nyní aktivní a spouští trénink s aktivní předvolbou.



Obrázek 9: Inspektor uprostřed tréninku — Live náhled ukazuje rekonstrukci, metric bar dole (Loss / LR / Gaussian Count / Iterace), karta předvolby „Preview,“ s „Modified“ odznakem pokud byly parametry upraveny

**Inspektor během tréninku:** Titulní lišta ukazuje globální postup „RadianceKit — Training NN %,“. Náhled renderuje běžící Gaussian rekonstrukci v reálném čase (každých 50 iterací aktualizováno — interval Live Preview lze nastavit v Settings → General → Training → Live Preview). Metric bar pod náhledem: aktuální Loss, Learning Rate, Gaussian Count a counter iterací (např. 1,600/5,000 u Preview předvolby). Karta předvolby

Inspektoru „Preview“ nese odznak „Modified“, jakmile se jakýkoli parametr odchýlí od built-in defaultu. Sidebar „Log“ sbírá události fáze SfM a tréninku.



Obrázek 10: Inspektor po dokončení tréninku — Náhled ukazuje hotovou rekonstrukci (2 991 Gaussianů po 5K iteracích za 13 s), titulní lišta „Training Complete — 2 991 Gaussians“

**Inspektor po tréninku:** Titulní lišta ukazuje finální počet Gaussianů (zde 2 991 — velmi kompaktní, protože syntetická scéna na světlém pozadí má jednoduchou geometrii). Náhled ukazuje hotový mrak bodů — orbitální drag navigace aktivní (otáčí kolem středu scény). Sekce tréninkových metrik je nyní naplněna finálními hodnotami, Loss History chart ukazuje průběh celých 5 000 iterací. Sekce exportu dole je nyní aktivní (všechna tlačítka formátů enabled).

Inspektor je pravý postranní panel v Expert Mode (§2). Sdružuje všechny tréninkové relevantní parametry v sedmi rozbalovatelných sekcích. Výchozí pořadí shora dolů při prvním spuštění je: Look, Předvolby, Konfigurace tréninku, Metriky, Loss Diagram, Enhancements a Export. Sekce „Look“, (úpravy obrazu po tréninku) je reálné UI přejmenování dřívější sekce „Finishing“ — její interní enum `rawValue` zůstává z důvodů persistence „Finishing“, zobrazená hlavička se jmenuje „Look“. Každou sekci lze kliknutím na hlavičku sbalit, pořadí přeuspořádat drag-and-drop (`InspectorView.swift:81-97`). **Při prvním spuštění jsou všechny sedmé sekce sbalené** (`InspectorCollapsedSections` defaultuje na `Set(InspectorSection.allCases)`); App-State poté ukládá preference sbalení a pořadí přes restarty aplikace.

Řada ovládacích prvků z Inspektoru se v téměř identické formě objevuje i v Nastavení (Kapitola 3) — typicky SfM backend, Sky Masking a podobné defaulty. Rozdělení je záměrné: Nastavení dodává globální šablonu aplikace pro nově založené projekty, Inspektor tyto hodnoty přepisuje pro aktuálně otevřený projekt. Kdo zná logiku ovládání jedné strany, může druhou používat poslepu.

Levý sloupec v Expert Mode — Project Navigator — nepatří k Inspektoru, ale je jeho přímým sousedem. Tam lze vybírat importované obrázky kliknutím, prohlížet je mezer-níkem v Quick Look a mazat přes tlačítko minus nebo klávesu Delete (s `Cmd-Z` pro vrácení akce). Inspektor sleduje aktuální výběr v sidebaru kontextově specifickými detailními informacemi, ale sedm hlavních sekcí zůstává vždy dostupných.

## Sekce Look (L1–L5)

Sekce Look (interní `rawValue` nadále „Finishing„) je nejvyšší sekce Inspektoru a sdružuje **úpravy obrazu po tréninku** na jednom místě. Všechny posuvníky pracují **nedestruktivně**: každý slider znovu aplikuje `FinishingPass` na nezměněný `pristine-snapshot` (původní DC-barva, -opacity, -škálování) — úprava je tím **idempotentní**, nikoli kumulativní. Výsledek se zobrazí **živě v náhledu** (WYSIWYG, přesně tak jako pozdější export) a **zapéká se do každého exportu**. Sekce je dostupná teprve **po dokončení tréninkového běhu** (předtím stojí „Available after a training run completes.“); její hodnoty se **při každém novém tréninku resetují**. Dokud běží export, jsou všechny posuvníky **uzamčeny** — objeví se lock-hint „Locked while exporting — the file uses the current settings.“, a `GroupBox` je disabled.

### L1 Slider Saturation



Inspektor → Sekce Look → `GroupBox` → Saturation.



Slider 0.5–1.2, zobrazení na dvě desetinná místa (např. „1.00“). Škáluje SH-DC chroma každého `Splat` kolem hodnoty luminance: 1.0 = beze změny, < 1.0 = odbarveno (barva tažena ke stupni šedi), > 1.0 = sytější. Matematicky se DC-barva přepočítává z `pristine-snapshot` (`desaturateDC`), takže opakované posouvání se nesčítá. Validováno na materiálu z DJI dronu (viadukt Pensford), který má sklon k přesycení — dronový default leží na 0.82. Působí pouze na barevný základ (SH-stupeň 0), vyšší SH koeficienty zůstávají nedotčeny.

#### JEDNODUŠE ŘEČENO

Jak syté jsou barvy hotového `Splatu`. 1.00 nechá vše tak, jak se to natrénovalo, hodnoty pod tím táhnou barvu směrem k šedi — dobré pro dronový nebo video materiál, který často vyjde přesycený. Hodnoty nad 1.0 ho dělají sytější. Můžeš posouvat sem a tam, jak chceš, aniž by se něco „nabalovalo“, protože aplikace vždy počítá znovu z nezměněného originálního stavu. Živě viditelné v náhledu a přesně tak v exportu.

## L2 Slider Splat length



Inspektor → Sekce Look → GroupBox → Splat length.



Slider 0.3–1.0, zobrazení na dvě desetinná místa. Táhne tři škálovací osy každého Gaussianu v log-prostoru k jejich průměru ( `shortenScale` , faktor `alpha` ): 1.0 = beze změny, menší hodnoty dělají protáhlé „jehlové“, Splaty kulatějšími, 0 by byly čisté koule. Útočí na jehlovité, přetažené Splaty, aniž by měnil celkovou velikost, a redukuje tím typické „konfetové“ artefakty. Aplikováno z pristine-snapshotu (původní log-škálování), proto idempotentní. Komutuje se Splat size (L3), protože obojí pracuje v log-prostoru.

### JEDNODUŠE ŘEČENO

Dělá příliš dlouhé, roztržitěné Splaty kulatějšími. 1.00 nechá tvar tak, jak se natrénoval, nižší hodnoty stlačují protáhlé „jehly“ do kulatějších kleksů — to zklidní zrnité rekonstrukce sužované konfetovými artefakty. Celková velikost zůstává stejná, jde jen o protáhlost. Lze bez obav kombinovat se Splat size (L3).

## L3 Slider Splat size



Inspektor → Sekce Look → GroupBox → Splat size.



Slider 0.5–2.0, zobrazení na dvě desetinná místa. Škáluje každý Gaussian uniformně na **všech** třech osách ( `sizeScale` ): 1.0 = beze změny,  $< 1.0$  = menší/hustší/ostřejší,  $> 1.0$  = větší/„nadýchanější“ (vyplňuje mezery mezi Splaty). Protože škálování leží v log-prostoru, je násobení realizováno jako aditivní offset `log(factor)` — to komutuje se Splat length (L2), protože konstantní offset ponechává odchylku-od-průměru nedotčenou. Z pristine-snapshotu, tedy idempotentní. Nové v této verzi.

### JEDNODUŠE ŘEČENO

Škáluje všechny Splaty rovnoměrně větší nebo menší. 1.00 je natrénovaný stav, hodnoty pod tím dělají mrak bodů hustší a ostřejší, hodnoty nad ním zakrývají mezery mezi Splaty (působí měkčeji/„nadýchaněji“). Praktické pro opticky uzavření dřavé rekonstrukce nebo naopak pro odhalení více detailu. Snese se bez problémů se Splat length (L2) — oba posuvníky se navzájem neovlivňují.

#### L4 Fade far region (se sub-slidery)



Inspektor → Sekce Look → GroupBox → Toggle „Fade far region“, plus sub-slidery „Fade start xradius“ a „Fade floor“.



Toggle, který aktivuje radiální pokles opacity se vzdáleností od těžiště kamer — slabě pozorovaná „far-konfeti“, na pozadí se vyblednou. **Pouze pro orbitální záběry:** toggle je disabled, když je `finishingContext.fadeEligible false` (lineární přelety, příliš málo nebo degenerované kamery); pak se místo sub-slidery objeví hint „Far-fade applies only to orbit captures (not this scene)“. Vhodnost se zjišťuje přes azimutální pokrytí pozic kamer (orbit obkrouží těžiště a vyplní mnoho sektorů kompasu, lineární přelet jen ~2). Dva sub-slidery řídí geometrii: **Fade start xradius** (1.0–3.0) nastavuje vnitřní poloměr jako násobek orbitálního poloměru, uvnitř kterého platí plná opacity; **Fade floor** (0.0–1.0) je faktor opacity daleko za fade-poloměrem. Důležité: fade **přeskakuje oblast Sky Dome** (frozen Gaussiany indexů [0, frozenCount]), aby se záměrná kopule pozadí nezdimovala.

#### JEDNODUŠE ŘEČENO

Vybledává rozmazané zbytky na vnějším okraji scény — přesně ty „far-konfetové“, chuchvalce, které u kruhových záběrů visí daleko vzadu. Funguje pouze u skutečných orbitálních / kruhových přeletů nebo příliš málo kamer je spínač zašedlý a hint vysvětluje proč. Když je aktivní, přidají se dva jemné regulátory: „Fade start xradius“ určuje, od jaké vzdálenosti (jako násobek poloměru kruhu) začíná vybledávání, „Fade floor“, jak silně zůstávají vzdálené Splaty nakonec ještě viditelné (0 = úplně pryč, 1 = beze změny). Záměrně rekonstruovaný Sky Dome (I44) se přitom nikdy nedotkne — obloha zůstává zachována.

#### L5 Tlačítko Reset finishing



Inspektor → Sekce Look → GroupBox → „Reset finishing“, (dole, malé tlačítko).



Resetuje všechna nastavení Look na defaulty (`FinishingPass.Settings() = Saturation 1.0, Fade vypnut, Splat length 1.0, Splat size 1.0`) a okamžitě spustí nový finishing, takže náhled skočí zpět na nezměněný natrénovaný stav. `controlSize(.small)`. Protože celý Look-stack počítá idempotentně z pristine-snapshotu, je „zpět na default“, přesně původní tréninkový výstup — žádná ztráta kvality opakovaným sem a tam. Jako všechny posuvníky sekce je během běžícího exportu uzamčeno.

#### JEDNODUŠE ŘEČENO

Jedním kliknutím vrátí všechny Look-regulátory na standard (Saturation 1.00, Fade vypnut, oba Splat slidery na 1.00) — náhled poté opět ukazuje přesně čerstvě natrénovaný výsledek. Praktické, když ses zahrál a chceš čistě začít znovu. Protože aplikace vždy počítá z originálního stavu, nedojde přitom k žádné ztrátě kvality. Dokud běží export, je tlačítko (jako slidery) uzamčeno.

## Sekce Předvolby (I1–I11)

Sekce Předvolby je nejrychlejší cesta, jak použít ověřenou konfiguraci. Built-in předvolby (Capture Class, Classic, MCMC, Hybrid) dodávají reprodukovatelné výchozí body z 560+ dokumentovaných experimentů; vlastní předvolby lze ukládat, exportovat, importovat a sdílet. Seznam je seskupený podle kategorií (Capture Class, Classic, MCMC, Hybrid, Custom) a více než jedna kategorie může být rozbalená současně. Přes mechanismus kontextového menu (pravý klik na řádek) jsou dostupné Export, Duplikace a — u vlastních předvoleb — Mazání.

### I1 Tlačítko Save...



Inspektor → Sekce předvoleb → Tlačítko Save... (akční lišta dole).



Otevírá popover s textovým polem a tlačítka Save/Cancel. Aktuální stav TrainingConfig se persistuje jako nová uživatelská předvolba (JSON kódovaná, ukládaná napříč aplikací). Save proces kopíruje všech 81 tréninkových parametrů plus aktuální strategii densifikace. Předvolba automaticky končí v kategorii Custom, bez ohledu na to, z jaké built-in předvolby byla odvozena. Prázdné názvy a vstupy z čisté bílé znaků se zahazují. Již existující názvy nejsou odmítány — každá předvolba má vlastní interní ID, duplicitní názvy jsou technicky povoleny, ale prakticky matoucí.

#### JEDNODUŠE ŘEČENO

Uloží tvou aktuální konfiguraci jako znovupoužitelnou předvolbu. Stiskni tlačítko, zadej v popoveru název a klikni Save — všech 81 parametrů včetně strategie densifikace přistane pod zvoleným názvem v kategorii Custom. Potřebuješ, když sis dal práci a nechceš na dalším projektu znovu od začátku ručkovat. Obzvláště praktické pro opakující se setupy jako „Dron 4K„ nebo „Indoor rychle“. Duplicitní názvy jsou technicky povoleny, ale prakticky matoucí — vezmi raději něco výmluvného.

## I2 Textové pole Preset Name



Save popover → Textové pole „Preset Name„.



Jednoduché textové pole se zaokrouhleným rámem, široká forma. Hodnota se při kliknutí na Save tlačítko přebírá jako název předvolby. Žádné omezení délky v UI, ale uložený název musí být JSON kódovatelný a zobrazitelný v UI seznamech — emoji a diakritika fungují. Obsah se při otevření popoveru automaticky resetuje na prázdný řetězec. Save tlačítko zůstává disabled, dokud je pole po trimu prázdné. Žádné auto-suggest a žádné předvyplnění názvem aktuálně aktivní předvolby.

### JEDNODUŠE ŘEČENO

Zde napíšeš název pro svou předvolbu. Vyber něco výmluvného jako „Dron 4K 30fps„ nebo „Interiér rychle“ — to ti později pomůže najít to v kategorii Custom. Emoji a diakritika jsou povoleny, tvrdé omezení délky neexistuje. Dokud je pole prázdné nebo obsahuje jen mezery, Save tlačítko zůstává zašedlé. Při opětovném otevření popoveru je pole opět prázdné — žádné předvyplnění aktivním názvem předvolby.

## I3 Tlačítko Cancel (Save dialog)



Save popover → Tlačítko Cancel (vlevo).



Zavírá popover bez uložení. Zahazuje obsah textového pole — při příštím otevření se opět logikou Save tlačítka (I1) resetuje na prázdné. Standardní styl tlačítka, žádné potvrzovací dialogy, žádné hotkey. Aktuální TrainingConfig zůstává nezměněna, protože cesta Save se vůbec neprovedla.

### JEDNODUŠE ŘEČENO

Zavírá Save popover, aniž by se cokoli ukládalo. Pokud sis to rozmyslel, překlepl ses nebo jsi dialog otevřel omylem — prostě klikni Cancel. Tvá aktuální tréninková konfigurace zůstává nezměněna, protože ještě nic nebylo zapsáno. Při příštím otevření popoveru pole názvu opět startuje prázdné. Žádný bezpečnostní dotaz, žádný hotkey — jen klik a pryč.

## I4 Tlačítko Save (Save dialog)



Save popover → Tlačítko Save (vpravo, prominentní styl).



Spouští samotnou persistenci. Validuje znovu ne-prázdný název (obránná kontrola) a poté zapisuje aktuální TrainingConfig jako JSON do úložiště aplikace. Následně zavírá popover. Zvýrazněno modře, zašedlé, dokud je textové pole prázdné. Pokud selže ukládání (např. protože je úložiště aplikace plné — velmi nepravděpodobné), neexistuje aktuálně viditelný chybový dialog; předvolba by se pak při příštím spuštění aplikace prostě neobjevila.

### JEDNODUŠE ŘEČENO

Kliknutím na Save přebíráš název a zapisuješ svůj aktuální setup jako novou předvolbu. Popover se zavře, předvolba se okamžitě objeví v kategorii Custom seznamu předvoleb a od nynějška ji lze aktivovat kliknutím. Tlačítko je modře zvýrazněno (`borderedProminent`) a zůstává zašedlé, dokud je pole názvu prázdné. Pokud selže ukládání (např. plně UserDefaults), není viditelný chybový dialog — předvolba by pak při příštím spuštění aplikace prostě chyběla.

## I5 Tlačítko Export...



Inspektor → Sekce předvoleb → Akční lišta → Tlačítko Export....



Exportuje aktuálně vybranou předvolbu jako `.radiancepreset` soubor (interně JSON). Disabled, pokud není vybrána žádná předvolba. Při kliknutí aplikace otevírá Save dialog s předvyplněným názvem souboru (název předvolby + přípona `.radiancepreset`). Uložený formát obsahuje kompletní TrainingConfig plus metadata (název, kategorie, ID, built-in flag). Dvojklik ve Finderu otevírá aplikaci — ale **ne** automaticky import; uživatel musí použít Import tlačítko (I6).

### JEDNODUŠE ŘEČENO

Vyber předvolbu v seznamu a klikni Export — pak ji můžeš uložit jako `.radiancepreset` soubor a poslat např. kolegovi nebo přenést na druhý Mac. Příjemce ji načte tlačítkem Import... (I6). Funguje stejně pro built-ins i tvé vlastní Custom předvolby. Tlačítko je zašedlé, dokud není v seznamu nic zakliknuto. Tip: Přes kontextové menu (I8) je to ještě rychlejší — nemusíš předvolbu nejprve selektovat.

**I6 Tlačítko Import...**

Inspektor → Sekce předvoleb → Akční lišta → Tlačítko Import....

**TECHNICKY**

Otevírá souborový dialog, který umožňuje pouze `.radiancepreset` soubory (vícenásobný výběr deaktivován). Při výběru se JSON soubor načte, validuje a vloží do kategorie Custom — s novým interním ID, aby nedošlo ke kolizím s built-ins. Import automaticky nastavuje kategorii na Custom, i když exportovaná předvolba původně byla např. built-in. Poškozené nebo nekompatibilní soubory se starší verzí schématu jsou tiše odmítány bez chybového dialogu (Console log ale informuje).

**JEDNODUŠE ŘEČENO**

Načte `.radiancepreset` soubor z disku. Užitečné, pokud ti někdo pošle ověřený setup nebo pokud chceš sám své oblíbené předvolby udržovat synchronizované přes několik Maců. Importované předvolby vždy přistávají v kategorii Custom — i když byly původně exportovány z built-ins. Poškozené nebo zastaralé soubory se tiše ignorují; v konzolovém logu pak stojí důvod. Vícenásobný výběr v dialogu je deaktivován, takže jeden soubor na kliknutí.

**I7 Řádek předvolby (aktivace kliknutím)**

Inspektor → Sekce předvoleb → každý řádek předvolby v každé kategorii.

**TECHNICKY**

Kliknutí na řádek předvolby nahrazuje všechna pole TrainingConfig hodnotami z předvolby, pamatuje si ID aktivní předvolby a resetuje Modified status. Aktivní zaškrtnutí před řádkem se objevuje pouze tehdy, pokud je předvolba vybrána a je nemodifikovaná. Jakmile se změní hodnota v TrainingConfig (slider, stepper, toggle v jiných sekcích Inspektoru), objeví se za názvem oranžový odznak „Modified„. Vestavěné předvolby nelze přepsat — při modifikaci se musí přes Save tlačítko (I1) založit vlastní kopie.

**JEDNODUŠE ŘEČENO**

Kliknutí na řádek aktivuje předvolbu a přebírá všechny v ní uložené hodnoty do aktuálních tréninkových nastavení. Zaškrtnutí před názvem ukazuje, která předvolba je právě aktivní. Jakmile poté přesuneš jakýkoli slider, stepper nebo toggle v jiných sekcích, objeví se za názvem oranžový odznak „Modified„ — protože tvůj setup se nyní od předvolby odchyluje. Built-in předvolby nelze přepsat; pokud chceš změny zachovat, založ přes tlačítko Save... (I1) vlastní kopii nebo duplikuj předvolbu (I9).

## I8 Kontextové menu „Export...“



Pravý klik na řádek předvolby → první položka „Export...“.



Identická funkcionality jako I5 (Export tlačítko), ale pohodlněji dostupná — aniž by musela být předvolba předtím vybraná. Exportuje přímo předvolbu, na kterou se v řádku kliklo. Funguje pro všechny kategorie předvoleb stejně (built-in nebo Custom), žádné omezení. Export obsahuje built-in flag a originální kategorii, ale při re-importu se kategorie podle popisu v I6 mapuje na Custom.

### JEDNODUŠE ŘEČENO

Rychlá cesta k exportu — pravý klik na požadovanou předvolbu a zvol „Export...“. Šetří obížděku přes předchozí zakliknutí a stisk Export tlačítka. Funguje stejně pro všechny kategorie, i pro built-ins. Vytvořený `.radiancepreset` soubor je identický s tím z I5; při pozdějším re-importu automaticky přistává v kategorii Custom.

## I9 Kontextové menu „Duplicate“



Pravý klik na řádek předvolby → druhá položka „Duplicate“.



Klonuje předvolbu do kategorie Custom. Generuje nové interní ID, připojuje „Copy“, k názvu a ukládá kopii. Funguje i pro built-in předvolby — klon je pak editovatelný. Originál zůstává nedotčen. TrainingConfig se kopíruje hodnota po hodnotě (JSON roundtrip), takže mezi originálem a kopií neexistují žádné referenční vazby.

### JEDNODUŠE ŘEČENO

Vytvoří editovatelnou kopii předvolby v kategorii Custom. Praktické, pokud např. chceš built-in „Quality“, předvolbu jako výchozí bod a pak chceš jen lehce posunout SSIM slider. Workflow: duplikovat, přejmenovat (kontextové menu nebo nový Save běh), upravit, hotovo. Originál zůstává nedotčen — můžeš se k němu kdykoli vrátit. Funguje i pro built-ins, což je jediná cesta, jak jejich hodnoty převzít jako základ a současně je udělat editovatelnými.

## I10 Kontextové menu „Delete„



Pravý klik na vlastní řádky předvoleb → poslední položka „Delete„ (červené, destruktivní).



Viditelné pouze u Custom předvoleb. Built-ins nelze smazat. Položka je označena jako destruktivní, v kontextovém menu se objevuje červeně a je oddělena dividerem, aby se na ni omylem nekliklo. Neexistuje **žádný** potvrzovací dialog — jedním kliknutím se předvolba okamžitě smaže. Smazaná předvolba není obnovitelná (Cmd-Z zde nefunguje — Undo v aktuálním buildu existuje pouze pro seznam obrázků, ne pro operace s předvolbami). Pokud byla smazaná předvolba právě aktivní, zůstává aktuální Training-Config nezměněna, jen aktivní výběr předvolby se vynuluje.

### JEDNODUŠE ŘEČENO

Mazání vlastních předvoleb. U built-ins (Quick, Preview, Balanced, Quality, Ultra Detail, Drone / Aerial, 360° Walkaround, Photo / Object atd.) „Delete„ vůbec není viditelné — nemůžeš je omylem zabít. Pozor: žádný bezpečnostní dotaz a žádné undo, jeden klik a předvolba je pryč. Pokud si nejsi jistý, předtím přes Export... (I5/I8) stáhni bezpečnostní kopii na disk — tu můžeš kdykoli znovu importovat. Pokud byla předvolba právě aktivní, tvá Training-Config zůstává nezměněna, jen zaškrtnutí zmizí.

## I11 Hlavička kategorie (rozbalit/sbalit)



Inspektor → Sekce předvoleb → každá hlavička kategorie (Capture Class, Classic, MCMC, Hybrid, Custom).

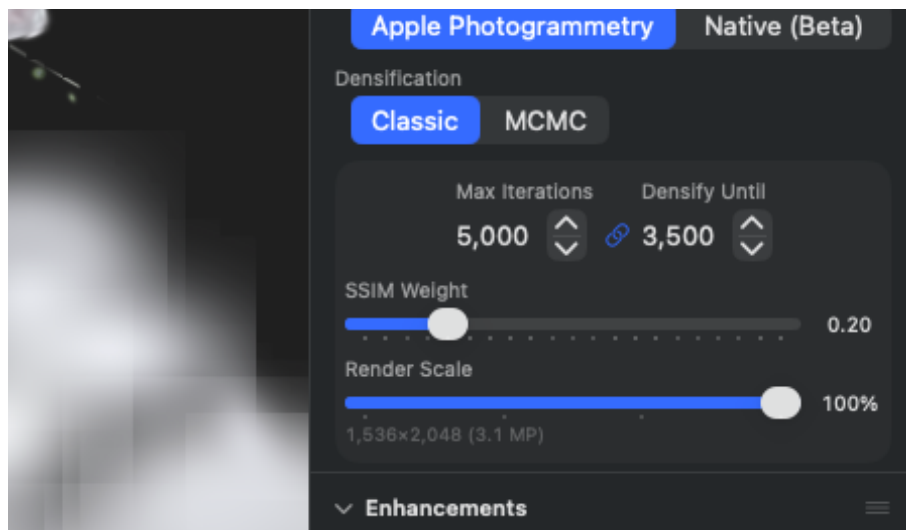


Stav sbalení per kategorie s různým defaultem: kurátorovaná skupina Capture Class startuje **rozbalená**, Classic, MCMC, Hybrid a Custom startují **sbalené**. Stav se nepersistuje — při restartu aplikace jsou všechny kategorie opět ve výchozím stavu. Šipka chevron rotuje animovaně. Číslo vpravo v hlavičce ukazuje počet předvoleb v dané kategorii. Click hit area zahrnuje celou hlavičkovou oblast.

### JEDNODUŠE ŘEČENO

Rozbalování a sbalování kategorií pro přehlednost seznamu předvoleb. Při startu aplikace je skupina Capture Class otevřená, Classic, MCMC, Hybrid a Custom jsou zavřené. Klik na hlavičku (kompletní oblast je klikatelná) a seznam se s krátkou chevron animací rozbalí nebo sbalí. Malé číslo vpravo ukazuje, kolik předvoleb leží v kategorii. Po restartu aplikace je opět výchozí stav — aplikace toto nastavení sbalení záměrně neukládá.

## Sekce Konfigurace tréninku (I12–I22)



Obrázek 11: Crop pouze sekce Konfigurace tréninku — Camera Alignment (Apple Photogrammetry aktivní, Native (Beta) neaktivní), Densifikace (Classic aktivní), Max Iterations 5,000 / Densify Until 3,500 s ikonou link, slider SSIM Weight 0.20, slider Render Scale na 100 % (1,536×2,048 = 3.1 MP)

Zde přistávají centrální páky: které SfM backend má počítat, jak pracuje densifikace, kolik iterací, jak velká SSIM váha. U MCMC strategie se objevují dva další toggler („MCMC Quality“ a „Auto-scale by scene“), které jsou v Classic režimu skryty. U Native SfM backendu se přidává pole FOV override, které je potřeba pouze pro video framy bez EXIF ohniska.

### I12 Picker Camera Alignment



Inspektor → Konfigurace tréninku → Camera Alignment (segmentovaný picker nahoře).

#### TECHNICKY

Segmentovaný picker se dvěma možnostmi: Apple Photogrammetry a Native (Beta). Výběr určuje použitý SfM backend při další rekonstrukci kamer. Zároveň ovlivňuje, které další prvky Inspektoru jsou viditelné: Native navíc ukazuje FOV override (I13), který je potřeba pouze u video framů bez EXIF. Poznámka: pro velmi velké outdoorové záběry můžeš výsledek externího nástroje (Metashape nebo COLMAP) zavést přes Workspace Import — viz Kapitola 1 (M5) a Kapitola 9 (Q3, Q6).

#### JEDNODUŠE ŘEČENO

Zde volíš, jak budou rekonstruovány pozice kamer — nejdůležitější přepínač pro koncovou kvalitu. Apple Photogrammetry je rychlý standard a pro většinu skenů objektů úplně stačí. Native (Beta) je App-Store kompatibilní vlastní vývoj, dobrý pro orbity a turntable scény, a u video framů bez EXIF potřebuje FOV override (I13). U velmi velkých outdoorových setů můžeš kamery alternativně spočítat v Metashape nebo COLMAP a výsledek načíst přes Workspace Import. Detaily a doporučení pro typ scény najdeš v Kapitole 9.

**I13 Pole FOV Override (Native SfM)**

KDE

Inspektor → Konfigurace tréninku → FOV Override (viditelné pouze při Camera Alignment = Native).

## TECHNICKY

Numerické textové pole (rozsah 0-170°), default 0 = automatické určení z EXIF nebo heuristiky. Ruční zadání je potřeba, pokud jsou vstupní obrázky extrahovány z videa, které neobsahuje metadata ohniska. Typické hodnoty: iPhone Wide ≈ 73°, DJI Mavic Wide-Crop ≈ 70°, dron s plnoformátovým senzorem ≈ 84°. Hodnota je clampována na [0, 170] — hodnoty mimo se přímo srážejí zpět. Působí pouze na nativní SfM pipeline (Q4/Q5); Apple Photogrammetry tuto hodnotu kompletně ignoruje.

## JEDNODUŠE ŘEČENO

Pokud tvé obrázky nemají EXIF (typické u extrahovaných video framů), zaznamenáš zde horizontální zorné pole kamery ve stupních. Orientační hodnoty: iPhone Wide ≈ 73°, DJI Mavic Wide-Crop ≈ 70°, dron s plnoformátovým senzorem ≈ 84°. 0 nechá aplikaci sama hádat — to často jde dobře, ale u vzácných objektivů to může selhat. Hodnoty nad 170° se automaticky srazí zpět. Pole je viditelné a účinné pouze tehdy, pokud jsi zvolil Native jako Camera Alignment (I12) — Apple Photogrammetry ho kompletně ignoruje.

**I15 Picker Densifikace**

KDE

Inspektor → Konfigurace tréninku → Densifikace (segmentovaný picker, vždy viditelný).

## TECHNICKY

Přepíná mezi dvěma strategiemi densifikace: Classic (originální 3DGS postup s Clone/Split/Prune a gradient thresholdem) a MCMC (Stochastic Gradient Langevin Dynamics s relokací, NeurIPS 2024). Při přepnutí z Classic na MCMC aplikace automaticky nastavuje MCMC specifická pole na ověřené default hodnoty (Reg-Weights = 0, MCMC cap multiplier 3.0, sample/noise schedule). Bez této automatické inicializace by relace se starými předvolbami trpěly bugem MCMC collapse v 1.4.4 (460K → 5 Gaussianů, watchdog kill). Picker výběr navíc určuje, které prvky Inspektoru jsou viditelné — u MCMC se objevují I16/I17. Detailní účinek polí v Kapitole 6, T11–T16 (Classic) a T61–T73 (MCMC).

## JEDNODUŠE ŘEČENO

Centrální volba strategie pro růst počtu Gaussianů. Classic je dobře vyladěný z 459 experimentů, generuje rychlé a kvalitní výsledky a nepotřebuje znát MCMC pole. MCMC je novější přístup (NeurIPS 2024), reprodukovatelnější a obejde se bez ručního ladění thresholdu — za to počítá asi 6× déle při srovnatelné kvalitě. Při přepnutí na MCMC aplikace automaticky nastavuje bezpečné defaulty, aby trénink neběžel do 1.4.4 collapse. Detaily ke strategickým polím najdeš v Kapitole 6 (T11–T16 Classic, T61–T73 MCMC).

**I16 Toggle MCMC Quality**

KDE

Inspektor → Konfigurace tréninku → MCMC Quality (pouze u Densifikace = MCMC).

## TECHNICKY

Přepíná gradient accumulation na 2 kroky (aktivní) resp. 1 krok (neaktivní). Akumuluje gradienty ze dvou po sobě následujících kameraových pohledů, než se provede optimizer step. Empiricky (Session 33, V544a) to redukuje finální L1 chybu asi o 6 % (0.0246 s Quality vs 0.0261 bez, při průměru 3 trialů na Horse-Full-MCMC). Cena: zdvojnásobený tréninkový čas. U velmi dlouhých tréninků (200K iterací) to vede k dalším 10+ minutám čekání — vyplatí se tedy pouze tehdy, pokud jsou opravdu potřeba poslední procenta kvality. Působí pouze na trénink, ne na export formát nebo zobrazení v náhledu.

## JEDNODUŠE ŘEČENO

Quality režim pro MCMC s akumulací gradientu přes dva pohledy. Empiricky zlepšuje konečný výsledek asi o 6 % (L1 0.0246 místo 0.0261 v Horse testu), za to stojí dvakrát déle. Pokud už beztak jedeš 200K MCMC trénink (klidně i 2 hodiny), přijde ještě skoupá hodina navíc. Vyplatí se u finálních showcase renderingů nebo na konci quality sweep relace, v denním workflow spíše ne. Viditelné pouze tehdy, pokud je Densifikace na MCMC (I15).

**I17 Toggle Auto-scale by scene**

KDE

Inspektor → Konfigurace tréninku → Auto-scale by scene (pouze u MCMC).

## TECHNICKY

Pokud aktivní, škáluje efektivní horní limit Max Gaussianů s SfM init point count × MCMC cap multiplier (default 3.0). Příklad: SfM dodává 250K init bodů, základní cap = 150K, multiplier 3.0 → efektivní horní limit =  $\max(150K, 750K) = 750K$ . Pokud deaktivováno, platí striktně jen základ. Bylo zavedeno pro v1.4.5, protože velké outdoorové záběry s více než 1000 framy a odpovídajícím hustým SfM bodovým polem hladověly densifikaci s rigidním 150K cap defaultem — nadbytečné body zůstávaly, nové nesměly vzniknout. Default OFF v Custom předvolbách, ON v MCMC built-ins. Působí pouze v tréninkovém čase, ne v exportu.

## JEDNODUŠE ŘEČENO

Nechá maximální počet Gaussianů růst s velikostí scény (přesněji: s počtem SfM init bodů). U malých scén sotva poznáš rozdíl, u velkých outdoorových scén je to často rozhodující pro kvalitu — jinak se trénink „udusí“, protože defaultní horní limit 150K je pro scénu příliš nízký. Bylo extra zavedeno pro v1.4.5, poté co velmi velké outdoorové sety (přes 1000 framů) viditelně viseli na capu. U MCMC built-in předvoleb už předem zapnuté; ve vlastních předvolbách defaultně vypnuté.

**I18 Stepper Max Iterations**

Inspektor → Konfigurace tréninku → GroupBox → Max Iterations.



Stepper s rozsahem 1 000–100 000, krok 1 000. Určuje celkový počet iterací optimizéru. Lineárně koreluje s tréninkovým časem (poloviční = asi 50 % času). Empirické sweet spots: 20K (Classic Balanced,  $L1 \approx 0.028$ ), 40K (Classic Quality,  $L1 \approx 0.023$ ), 200K (MCMC Full,  $L1 \approx 0.0246$ ). Nad 40K u Classic v průměru přináší sotva zlepšení — diminishing returns. Při změně se, pokud je funkce link (I19) aktivní, Densify Until proporcionálně táhne s sebou (default poměr: 0.5, tj.  $\text{Densify-Until} = \text{Max}/2$ ).

**JEDNODUŠE ŘEČENO**

Kolik tréninkových kroků se jede — více je lepší, ale stojí to také lineárně více času. Pravidlo: 20 000 pro dobrou kvalitu, 40 000 pro optimum u Classic strategie (nad to už v průměru sotva něco přinese). MCMC potřebuje výrazně víc, 200 000 je tu standard. Zdvojnásobení iterací zhruba zdvojnásobí tréninkový čas. Při aktivním Link tlačítku (I19) je Densify Until proporcionálně tažen — prakticky vždy to, co chceš.

**I19 Tlačítko Link/Unlink (Densify ↔ Iterations)**

Inspektor → Konfigurace tréninku → GroupBox → malé link tlačítko mezi Max Iterations a Densify Until.



Toggle tlačítko, které zmrazí poměr Densify Until k Max Iterations. Při aktivním (link icon zvýrazněn) se při každé změně Max Iterations Densify Until proporcionálně dotáhne. Při unlink (link-plus icon) hodnoty zůstávají nezávislé. Default je linked, protože to odráží typickou korelaci — pokud trénink protáhneš na dvojnásobné iterace, většinou chceš nechat běžet densifikaci proporcionálně déle. Poměr se při nastavení link tlačítka počítá z aktuální hodnoty; typický poměr je 0.5 ( $\text{Densify-Until} = \text{poloviční počet iterací}$ ).

**JEDNODUŠE ŘEČENO**

Malé spínací tlačítko mezi Max Iterations a Densify Until. Pokud je aktivní (link icon zvýrazněn), tyto dvě hodnoty putují společně — zdvojnásobí iterace, zdvojnásobí se i Densify Until ve stejném poměru. Pokud ne (`link.badge.plus icon`), můžeš je nastavit nezávisle. Standard je linkováno, protože to odráží typickou korelaci — delší trénink chce většinou i delší densifikační fázi. Pro 99 % případů ho nech zaháknuté.

## I20 Stepper Densify Until



Inspektor → Konfigurace tréninku → GroupBox → Densify Until.

### TECHNICKY

Stepper s rozsahem 500–50 000, krok 500. Určuje iterační index, od kterého už nepřibývají nové Gaussiány skrze Clone/Split (Classic) nebo Relokaci (MCMC). Po dosažení se pouze zjemňuje pozice a barva. Vyšší hodnoty = více Gaussianů = větší soubor, delší per-iteration time (+30–60 % GPU time na krok). Typické hodnoty: 15K (pro 30K max-iter), 20K (pro 40K), 100K (pro 200K MCMC). Při aktivním link (I19) automaticky škáluje. Působí jinak u Classic vs MCMC: Classic úplně zastavuje růst, MCMC zastavuje relokační logiku, ale sample/noise adaptace běží dál.

### JEDNODUŠE ŘEČENO

Do které iterace smí přibývat nové Gaussiány — u Classic přes Clone/Split, u MCMC přes Relokaci. Poté už jde jen o zjemnění barvy a tvaru existujících bodů. Vyšší = více detailu, ale také větší soubor a +30–60 % GPU času na krok. Typické hodnoty: 15K (pro 30K max-iter), 20K (pro 40K), 100K (pro 200K MCMC). Visí normálně přes Link (I19) na Max Iterations — zřídka má smysl to ručně oddělit.

## I21 Slider SSIM Weight



Inspektor → Konfigurace tréninku → GroupBox → SSIM Weight.

### TECHNICKY

Slider 0.0–1.0 v krocích 0.05, zobrazení jako „0.20“. Mísí L1 loss (0.0) a SSIM loss (1.0). L1 napíná jas na pixel, SSIM strukturální podobnost (hrany, lokální statistiky). Default 0.2 je hodnota z originálního 3DGS paperu (Kerbl 2023) a reverse-engineered jako robustní kompromis v řadě sezení. Vyšší hodnoty (0.5+) preferují zachování detailu, ale mohou ignorovat lokální jasové chyby. Nižší hodnoty (< 0.1) vedou ke ztrátě detailu na ostrých hranách. Výpočet SSIM běží v shaderu s 11×11 Gaussovským oknem. Výkon: při 0.0 (pouze L1) je trénink asi o 8–12 % rychlejší, protože SSIM výpočet v shaderu se přeskakuje.

### JEDNODUŠE ŘEČENO

Jak silně se váží strukturální podobnost obrazu (hrany, lokální vzory) vůči čistému porovnání jasu. 0.2 je standard z originálního 3DGS paperu a stačí pro téměř všechny scény. Vyšší (0.5+) u jemných struktur jako vlasy, srst nebo vegetace — tam pomáhá víc strukturální váhy. Nižší (0.0) udělá trénink asi o 8–12 % rychlejší, protože SSIM výpočet v shaderu se přeskakuje, ale stojí detail na ostrých hranách. Kdo nemá dobrý důvod pro změnu, nechá 0.2.

**I22 Slider Render Scale**

Inspektor → Konfigurace tréninku → GroupBox → Render Scale.

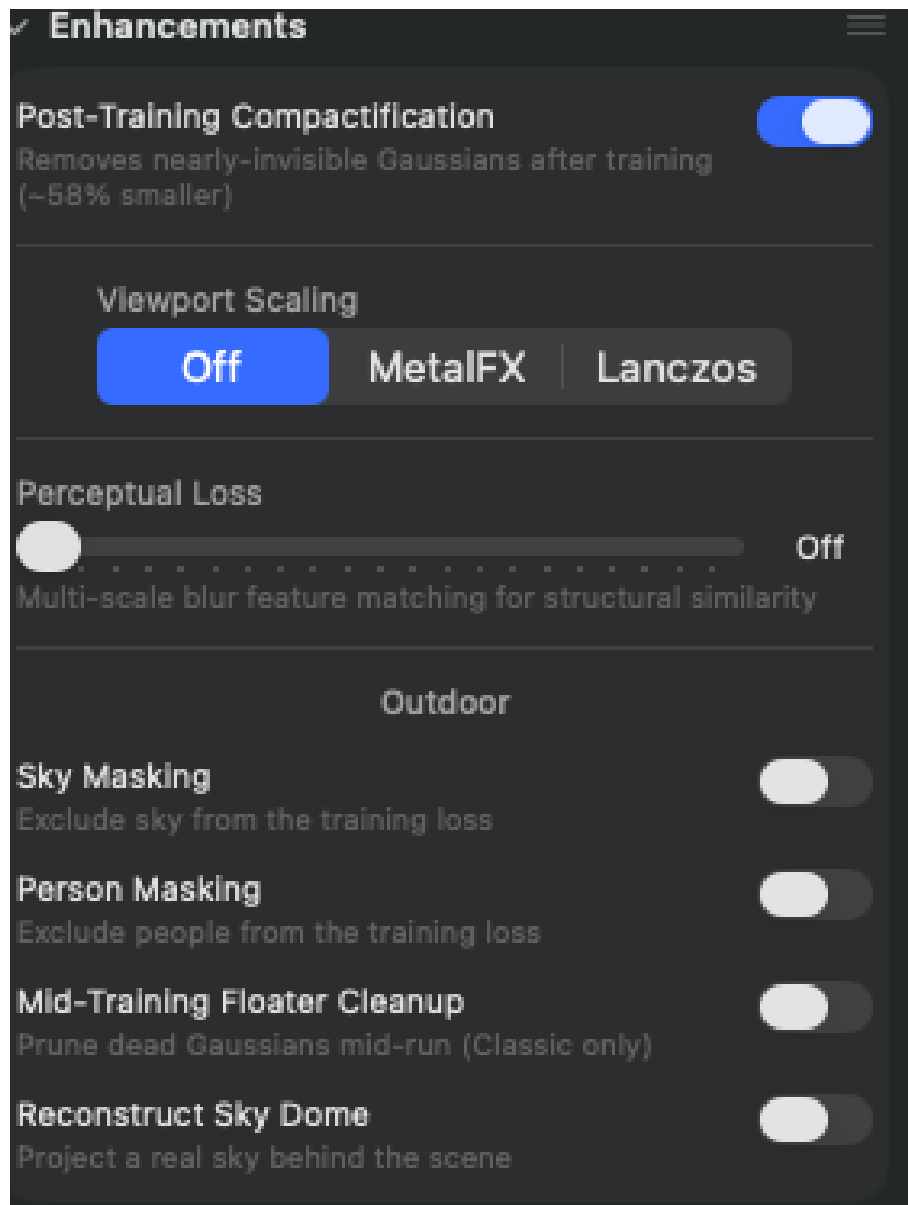
**TECHNICKY**

Slider 0.25–1.0 v krocích 0.25, zobrazení jako „100 %“. Škáluje tréninkové render rozlišení relativně k velikosti zdrojového obrazu. Největší páka na výkon: 50 % redukuje GPU čas asi o 75 % (protože 4× méně pixelů), 25 % asi o 94 %. Gradient threshold se automaticky škáluje s ním. Pod sliderem se objevuje live zobrazení rozlišení v MP (např. „2304×1296 (3.0 MP)“). Pokud se aktuální hodnota odchyluje od doporučené, oranžovým písmem se vkládá „— recommended: 50 %“. Doporučení cílí na ~3 MP efektivní rozlišení — oblast nejefektivněji zpracovávaná Apple Silicon GPU. 4K zdrojové obrázky dostávají např. automaticky doporučeno 25 %, FullHD obrázky 100 %. Změna navíc spouští realokaci bufferu.

**JEDNODUŠE ŘEČENO**

S jakým rozlišením trénink renderuje — jedna z největších pák výkonu. Plné (100 %) dává nejlepší kvalitu, ale u velkých obrázků stojí hodně GPU času. Poloviční (50 %) šetří zhruba 75 % GPU času, protože se počítá čtyřikrát méně pixelů — perfektní pro 4K zdroje. Pod sliderem vidíš efektivní rozlišení v megapixelech; aplikace cílí na asi 3 MP, protože to na Apple Silicon běží nejefektivněji. Pokud se tvá hodnota od toho odchyluje, aplikace zobrazí oranžový „recommended“ hint — většinou se vyplatí ho následovat.

## Sekce Enhancements (I26–I29, I42–I44)



Obrázek 12: Crop pouze sekce Enhancements — tři řádky: Post-Training Compactification (toggle zapnut), Viewport Scaling (segmentovaný picker Off/MetalFX/Lanczos), Perceptual Loss (slider na „Off“). Každý řádek se subtitle vysvětlující funkcí

Sekce Enhancements sdružuje tři funkce, které vylepšují kvalitu obrazu, aniž by měnily samotné jádro tréninkové smyčky. První dvě (I26–I27) jsou **post-training** resp. **viewport stupně**: Compactification po skončení tréninku uklízí, Viewport Scaling je čistý viewport renderer, který neovlivňuje běžící trénink. Perceptual Loss (I29) je přes svou sekční příslušnost součástí tréninku — aktivuje se během tréninku jako dodatečný loss term, proto oddělení od viewport regulátorů přes divider. Počínaje verzí v1.6 má sekce navíc skupinu Outdoor (I42–I44: Sky Masking, Mid-Training Floater Cleanup, Reconstruct Sky Dome) — tréninkové volby proti floaterům v obloze, které dříve sídlily v okně Nastavení a nyní sedí zde per projekt.

**I26 Toggle Post-Training Compactification**

Inspektor → Enhancements → Post-Training Compactification.

**TECHNICKY**

Aktivuje V443 post-processing: po skončení tréninkových iterací se Gaussiany s opacity pod 0.01 (1 % viditelnost) mažou. Empiricky to redukuje velikost souboru asi o 55–58 % bez viditelné ztráty kvality — protože tyto Gaussiany vizuálně stejně nepřispívají. Compactification běží jako GPU compact pass a trvá podle počtu Gaussianů zlomky sekund až několik sekund. Neovlivňuje výkon tréninku. Pokud je tento toggle vypnut, exportují se i neviditelné Gaussiany — relevantní pouze tehdy, pokud chceš formát použít pro další tréninkovou fázi (Continue Training), jinak plýtvání pamětí.

**JEDNODUŠE ŘEČENO**

Po tréninku uklízí Gaussiany, které stejně nevidíš (opacity pod 1 %). Dělá exportní soubory asi o polovinu menší (55–58 % redukce velikosti) bez viditelné ztráty kvality. Běží jako krátký GPU pass po poslední iteraci, trvá jen zlomky sekund až několik sekund. Mělo by být prakticky vždy zapnuté — jediný důvod to vypnout je, pokud chceš trénink později pokračovat přes Continue Training a musíš si nechat i neviditelné Gaussiany. U normálních exportních workflow prostě nech zapnuté.

**I27 Picker Viewport Scaling**

Inspektor → Enhancements → Viewport Scaling  
(segmentovaný picker se třemi možnostmi: Off, MetalFX, Lanczos).

**TECHNICKY**

Jediný segmentovaný picker, který vybírá viewport upscaler — tři možnosti se **navzájem vylučují**. Pokud je tréninkové rozlišení (přes I22 Render Scale) nižší než velikost viewportu, zvolený mód škáluje renderovaný frame nahoru na velikost zobrazení. **Off** = prosté bilineární natažení. **MetalFX** = Apple ML-based Spatial Upscaler, nejostřejší možnost (ML model je optimalizován na ostré hrany), režie asi 1-2 ms na frame na M3 GPU. **Lanczos** = Apple Metal Performance Shaders s 8-tap sinc resamplingem, klasicky bez ML, minimální režie (< 0.5 ms), kvalita pod MetalFX, ale bez ML typického „uhlazování“, jemných liniových struktur. Renderer pipeline se při přepnutí živě překonfiguruje — viditelné okamžitě, bez restartu. **Pozadí:** dříve to byly dva oddělené toggly (MetalFX + Lanczos), které mohly být zapnuté současně — rozporuplný stav, ve kterém MetalFX tiše přebíjel Lanczos. Picker tento stav odstraňuje; případně z starších sessions zděděný „oba-zapnuté“ stav se při příštím přepnutí sám zhojí na MetalFX. Působí **pouze** na live viewport, ne na renderované exporty (orbit video, screenshoty) — ty se renderují v plném zdrojovém rozlišení.

**JEDNODUŠE ŘEČENO**

Ostří live obraz v viewportu — obzvláště užitečné, pokud pracuješ s redukováním tréninkovým rozlišením (Render Scale 50 %, viz I22). Tři stupně, z nichž je vždy aktivní jen jeden: „Off“, prostě natáhne pixely, „MetalFX“ používá Apple machine learning pro nejostřejší hrany (prakticky vždy nejlepší volba), „Lanczos“, je klasický filtr bez ML — vezmi ho jako fallback, pokud ti MetalFX v některé scéně hladí linie nebo ukazuje artefakty. Působí živě, bez restartu. Působí pouze v live viewportu, ne na exportovaná orbit videa nebo screenshoty — ty se renderují v plném zdrojovém rozlišení. Na rozdíl od dřívějšíka už nemůžeš omylem zvolit dva módy současně.

**I29 Slider Perceptual Loss**

Inspektor → Enhancements → Perceptual Loss.

**TECHNICKY**

Slider 0.0–0.2 v krocích 0.01, zobrazení při 0.0 jako „Off“, jinak jako „0.05“ atd. Aktivuje dodatečný loss term, který porovnává multi-škálované Gaussian blur renderingu s ground truth obrazem (3 blur škály). Zachycuje strukturální rozdíly, které L1+SSIM samo o sobě nepoznává. V460 implementace. Empiricky hodnota 0.05–0.1 zlepšuje L1 skóre v relacích o pár procent, ale stojí ~5 % tréninkového času (dodatečný forward pass přes blur kernely). Nad 0.15 se trénink stává nestabilním a L1 se opět zhoršuje (loss term dominuje optimalizaci). Působí **během** tréninku, ne v post-processingu — přes pozici v sekci „Enhancements“, tedy není čisté zlepšení následně.

**JEDNODUŠE ŘEČENO**

Dodatečný podíl lossu, který kontroluje strukturální podobnost obrazu přes tři různé úrovně rozostření. Pomáhá obzvláště u scén s jemnými strukturami jako vlasy, látka nebo vegetace, protože zachycuje vzory, které L1+SSIM samo o sobě nevidí. Menší hodnoty jsou bezpečnější — 0.05 až 0.1 je sweet spot, nad 0.15 se trénink stává nestabilní a loss se opět zhoršuje. Na 0 (Off) je funkce kompletně vypnutá a nic nestojí; aktivní polkne asi 5 % tréninkového času na extra forward pass přes blur kernely. Působí navzdory sekci „Enhancements“, přímo během tréninku, ne v post-processingu.

**I42 Sky Masking**

Inspektor → Enhancements (skupina Outdoor) → Toggle „Sky Masking“. Bound: `AppState.trainingConfig.skyMaskingEnabled` (per projekt, `@DefaultFalse` ). Default: `false` .

**TECHNICKY**

Aktivuje pre-training segmentaci pixelů oblohy přes Apple Vision. Před startem tréninku se pro každou vstupní kameru extrahuje oblast oblohy přes Apple Vision foreground mask (obloha = pozadí) a přiřadí jako per-pixel maska k dané kameře. Během tréninku se příspěvek lossu na pixel násobí komplementem masky oblohy — pixely oblohy přispívají 0 ke gradientu, takže Gaussians, které se projektují do oblohy, nedostávají optimalizační signály a tudíž se nestávají „hustšími“, nebo „světlejšími“. Významně redukuje floatery (tmavé chuchvalce v obloze) u outdoorových / dronových scén. Stojí ~3 % L1 regrese u klasického 40K tréninku (viz `memory/dev_outdoor-floater-reduction.md`). Smysluplné pouze u outdoorových scén s jasně rozeznatelnou oblohou; u interiérových scén nebo bílého pozadí identifikuje segmentace oblohy nesprávné oblasti a blokuje validní loss signály. Hodnota se nyní ukládá per projekt (už ne globálně v aplikaci) a řídí se předvolbou / souborem scény.

**JEDNODUŠE ŘEČENO**

U outdoorových záběrů s oblohou v obraze často vznikají černé nebo barevné chuchvalce v obloze — takzvané „floatery“. Tato volba automaticky pozná, kde je obloha, a řekne tréninku: „Nech oblohu na pokoji.“ Funguje velmi dobře u dronových letů a krajinných scén. U interiérů nebo tmavých pozadí to může obraz zhoršit — takže zapínat jen pokud je vidět skutečná obloha. Detaily: `memory/dev_outdoor-floater-reduction.md`.

## I43 Mid-Training Floater Cleanup



Inspektor → Enhancements (skupina Outdoor) → Toggle „Mid-Training Floater Cleanup“. Bound: `AppState.trainingConfig.floaterCleanupEnabled` (per projekt, `@DefaultFalse`). Default: `false`.

### TECHNICKY

U Classic 40K tréninku (předvolba „P4 Quality“) zapíná dva další density-control průchody: u iterace 20,000 a u iterace 30,000. Oba průchody prohledávají všechny Gaussiany podle tří kritérií: (a) velmi nízká opacity (standardně 0.005), (b) nepatrná velikost ve screen space, (c) žádné příspěvky k lossu v posledních 1000 iteracích. Gaussiany splňující všechny tři podmínky jsou prořezány. Efekt: ~5–15 % méně Gaussianů na konci tréninku, viditelně méně tmavých chuchvalců v obloze u dronových / outdoorových scén. Stojí ~1–3 % L1 regrese u blízkých interiérových scén, proto není aktivováno defaultně. Hodnota se pamatuje přes restarty (na rozdíl od S7). Dvě iterace cleanupu (20K, 30K) jsou napevno definovány a nelze je aktuálně měnit přes UI; u kratších tréninků (např. P2 Preview 5K) toggle nemá efekt, protože iterační značky nedosáhne. **Nové:** Toggle je nyní ovladatelný pouze tehdy, když aktivní předvolba používá **Classic** densifier (`densificationStrategy == .classic`). Při MCMC nebo Hybrid se **disabled** a objeví se inline-hint, protože tyto strategie si mrtvé Gaussiany ošetřují stejně samy (MCMC přes relokaci, Hybrid přes kombinovanou reloc-/noise-logiku) — manuální cleanup průchody by tam byly bezúčelné, resp. kontraproduktivní. Code-reference: `RadianceKitApp.swift, General-Tab`. Detaily: `memory/dev_outdoor-floater-reduction.md`.

### JEDNODUŠE ŘEČENO

Během tréninku občas vznikají „mrtvé“, Gaussovy body, které už k obrazu nepřispívají, ale zabírají paměť. Tato volba dvakrát během dlouhého tréninku (u 20K a 30K iterací) uklidí a odstraní tyto pozůstatky. U outdoorových scén s oblohou je to obzvláště užitečné, protože tam se sbírá nejvíce floaterů. U malých tréninků nebo detailů nábytku spíše není potřeba. Spínač lze zapnout pouze tehdy, když tvá předvolba používá **Classic** densifier — u MCMC nebo Hybrid předvoleb je zašedlý (i s krátkým vysvětlením), protože ty si své mrtvé body uklízejí samy.

**I44 Reconstruct Sky Dome**

Inspektor → Enhancements (skupina Outdoor)  
→ Toggle „Reconstruct Sky Dome„. Bound:  
`AppState.trainingConfig.skyDomeEnabled` (per projekt, `@DefaultFalse` ). Default: `false` .

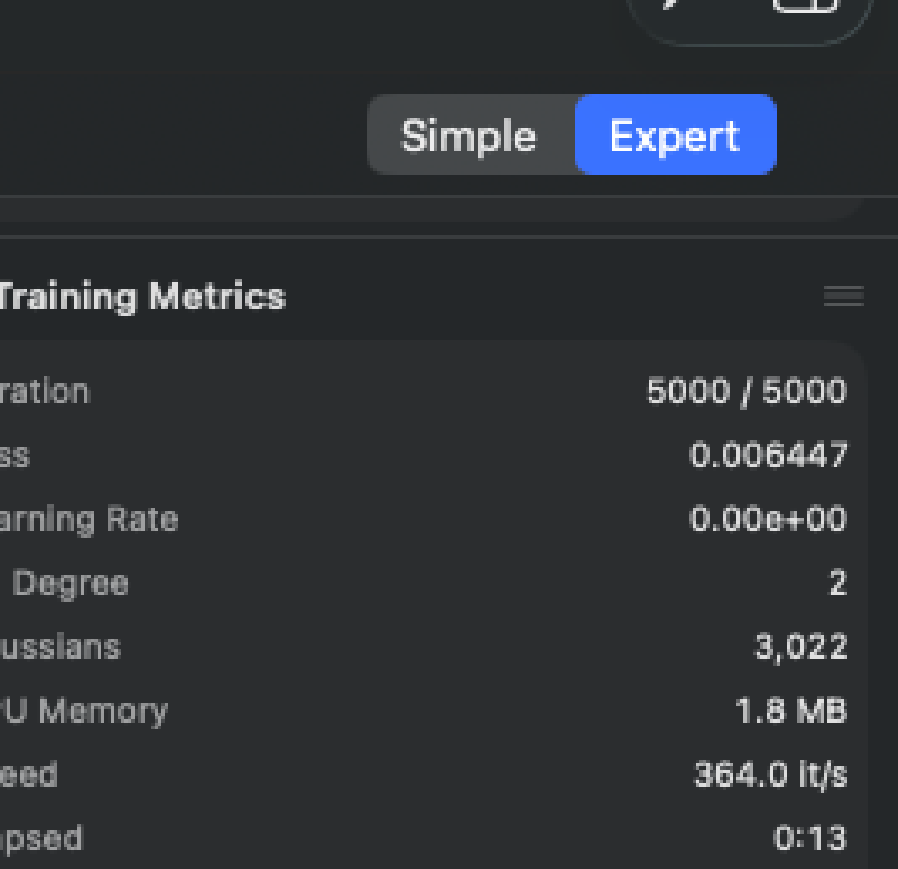
**TECHNICKY**

Aktivuje pre-training projekci Sky Dome (V549e MVP). Po SfM a před startem tréninku se pro každou vstupní kameru extrahuje maska oblohy Apple Vision sdílená s S7, pixely oblohy se s pomocí kameraových intrinsik ent-projektují na virtuální povrch koule (standardní poloměr 8x poloměr scény). Na této kouli se inicializuje ~5000 nových Gaussianů s průměry barev z projektovaných pixelů oblohy, velmi velkou škálou (1.0 v jednotkách scény) a počáteční opacity 0.95. Těchto 5000 Gaussianů není sky mask v klasickém slova smyslu — trénují se jako všechny ostatní, ale díky vysoké počáteční opacitě zůstávají v tenké slupce. Výsledek: u 360° novel-view u outdoorových / dronových scén se místo tmavých konfeti chuchvalců objevují skutečná barva oblohy a struktury mraků. Hodnota se pamatuje přes restarty. Smysluplné pouze u outdoorových scén s alespoň 360° pokrytím kamerou; u čistých object capture bez výhledu na oblohu nemá efekt. Status: experimentální, širší A/B validace přes další outdoorové sety teprve čeká.

**JEDNODUŠE ŘEČENO**

Místo aby trénink zkoušel oblohu „uhádnout“, z několika viditelných pixelů (což vede k floatérům), RadianceKit projektuje pixely oblohy přímo na virtuální kouli kolem scéně ještě před startem tréninku. Když pak hotovou scénu otáčíš v 360°, vidíš skutečnou oblohu místo černých chuchvalců. Funguje pouze u outdoorových záběrů, kde je opravdu obloha v obraze. U scénů obýváku nebo studiových záběrů nic nepřináší.

## Sekce Metriky (I30–I38)



The screenshot shows the 'Expert' view of the RadianceKit interface. At the top, there are two buttons: 'Simple' (disabled) and 'Expert' (active). Below this is a section titled 'Training Metrics' with a hamburger menu icon on the right. The metrics are displayed in a table format:

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

Obrázek 13: Crop pouze sekce Training Metrics po dokončeném tréninku (5K iterací, 2 991 Gaussianů final) — Tabulka s tréninkovými metrikami (Iteration, Loss, SSIM Loss, Combined Loss, Gaussian Count, Learning Rate, Elapsed, ETA)

Zatímco trénink běží, ukazuje sekce Metriky devět live hodnot z tréninkové smyčky. Před startem tréninku je sekce prázdná („Start training to see live metrics,“). Všechny hodnoty se aktualizují každých ~30 iterací (frekvence aktualizace streamu). Sekce je read-only — žádný prvek není klikatelný nebo změnitelný. Pro hlubší analýzu použij JSONL tréninkové logy pod `~/Documents/RadianceKit/Logs/` (skript `python3 scripts/analyze_logs.py best 5`).

**I30 Iterace**

Inspektor → Metriky → Iteration. Read-only.

**TECHNICKY**

Zobrazení ve formátu „4523 / 40000“ — aktuální iterace přes celkové plánované iterace. Počítá synchronně s tréninkovou smyčkou, která pushuje hodnoty každých ~30 iter. Druhé číslo odpovídá hodnotě Max Iterations v okamžiku startu; už se nemění, ani když uživatel poté přesune stepper — běžící běh používá svoji vlastní snapshot kopii. Pokud aplikace přes menu Training přidává iterace (Continue Training +5K/+10K/+20K), jmenovatel se zvyšuje.

**JEDNODUŠE ŘEČENO**

Kde trénink právě stojí. „4523 / 40000“ znamená: 4523 ze 40 000 kroků je hotových, tedy asi 11 % hotovo. Levé číslo se počítá v sekundovém taktu; pokud zůstane stát minuty, trénink visí — většinou hint na GPU throttling nebo konkurující aplikaci. Pravé číslo odpovídá hodnotě Max Iterations (I18) při startu tréninku a už se nemění, ani když stepper později přesuneš. Při Continue Training (+5K/+10K/+20K) roste s dodatečnými kroky.

**I31 Loss**

Inspektor → Metriky → Loss. Read-only.

**TECHNICKY**

Float hodnota se šesti desetinnými místy (např. „0.024385“). Měří kombinovaný L1+SSIM loss (mix kontrolovaný přes I21 SSIM Weight) plus volitelně Perceptual Loss (I29) a jiné regularizéry. Škála není absolutní, ale závislá na scéně — vyžaduje pro většinu srovnání stejný dataset. Typické koncové hodnoty u dobrých konfigurací: - 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 (geometricky horší) - Indoor apartmány: 0.018–0.025

Hodnoty nad 0.10 po 5K iterací naznačují SfM problémy (špatné pozice kamer) — přerušit a SfM přepočítat.

**JEDNODUŠE ŘEČENO**

Jak daleko je renderovaný obraz od originálu — kombinováno z L1, SSIM a popř. Perceptual Loss. Menší je lepší. Pod 0.03 je obvykle opravdu dobré, pod 0.05 ještě OK, outdoorové scény leží geometricky spíše na 0.03-0.06. Nad 0.10 po několika tisících iterací je varovný signál — většinou to leží v rekonstrukci kamer (SfM neprošla čistě). Škála není absolutní, ale závislá na scéně; srovnání dělat pouze v rámci stejného datasetu. Pokud číslo náhle skočí nahoru, většinou došlo k události gradient explosion.

### I32 Learning Rate



Inspektor → Metriky → Learning Rate. Read-only.

#### TECHNICKY

Zobrazení vědeckým zápisem (např. „1.60e-04“). Aktuální learning rate pro position parametry (3DGS má šest nezávislých LR pro Position, SH-DC, SH-Rest, Opacity, Scale, Rotation — zobrazena zde je position LR jako reprezentativní velikost). Default počáteční hodnota 1.6e-4, která přes exponential decay klesá na ~1.6e-6 do konce tréninku. Pokles je nastavitelný přes LR schedule pole v konfiguraci tréninku (T pole v Kap. 6). Pokud LR zůstává neobvykle vysoké (např. 1e-3 nebo více po 10K iterací), může to naznačovat chybně načtenou konfiguraci.

#### JEDNODUŠE ŘEČENO

Jak velké jsou aktuálně optimační kroky — konkrétně learning rate pro pozice Gaussiánů. Startuje na 1.60e-04 a klesá exponenciálně na asi 1.60e-06 ke konci tréninku („1.60e-06“ = 0.0000016). Průběh běží automaticky, tady nemusíš nic ladit. Pokud je hodnota po 10 000+ iteracích stále větší než 1e-3, byla pravděpodobně načtena chybná konfigurace — trénink přerušit a vybrat předvolbu znovu. Interně má 3DGS šest nezávislých learning rate (Position, SH-DC, SH-Rest, Opacity, Scale, Rotation); zde vidíš jen position LR jako zástupce.

### I33 SH Degree



Inspektor → Metriky → SH Degree. Read-only.

#### TECHNICKY

Celé číslo 0–3. Spherical Harmonics stupeň pro barevnou reprezentaci. Začíná na 0 (pouze DC komponenta, tj. směrově nezávislá barva na Gaussian — tedy jen RGB konstanta) a v průběhu tréninku progresivně roste na 3. Standardní schedule zvedá stupeň o 1 při iteracích 1000/2000/3000. SH-3 odpovídá 48 barevným koeficientům na Gaussian (3 RGB kanály × 16 SH básových funkcí). Vyšší SH stupeň znamená více směrově závislých odrazů (lesklé povrchy vypadají správně různě pod různými úhly pohledu), ale také více paměti a pomalejší trénink.

#### JEDNODUŠE ŘEČENO

Jak komplexní je momentálně reprezentace barvy na Gaussian. Startuje na 0 (pouze jedna směrově nezávislá barva na bod) a stupňovitě se zvyšuje na 3 — typicky u iterace 1000, 2000 a 3000. Stupeň 3 znamená 48 barevných koeficientů na Gaussian a umožňuje směrově závislé odrazy, tedy že lesklé povrchy z různých úhlů pohledu vypadají správně různě. Aktivně se toho nemusíš dotýkat, schedule běží automaticky. Vyšší stupeň stojí více paměti a lehce zpomaluje trénink — ale to je cena za realistická lesklá místa.

## I34 Gaussians



Inspektor → Metriky → Gaussians. Read-only.

## TECHNICKY

Aktuální počet Gaussianů v modelu, formátováno s locale separátorem (např. „524.318“). Růst: - Classic: startuje od SfM init bodů (typicky 50K-300K), roste skrze Clone/Split až krátce před Densify Until, poté statický až do konce tréninku (modulo pruning) - MCMC: sample body se přidávají až do MCMC capu, poté už jen reloace

Healthy koncové hodnoty: - Classic Quality: 400K-700K (Horse 524K, Garden 800K) - MCMC Full: přesně na capu (default 150K, s auto scale multiplier × SfM count podle scény 500K-1.5M)

U MCMC číslo klesá na < 60 % capu → anomálie (collapse indikátor, naznačuje příliš agresivní regularizéry).

## JEDNODUŠE ŘEČENO

Kolik Gaussian bodů má 3D model právě teď. Roste během tréninku, dokud není dosaženo Densify Until (I20); poté číslo zůstává prakticky konstantní. Více bodů = více detailu, ale také větší soubor a pomalejší rendering v viewportu. 500 000 Gaussianů je typická středová hodnota pro Classic Quality na střední scéně; MCMC Full přistává podle Auto-Scale (I17) na 500K až 1.5M. Pokud číslo u MCMC náhle klesne pod 60 % capu, je to collapse indikátor — většinou příliš agresivní regularizéry.

## I35 GPU Memory



Inspektor → Metriky → GPU Memory. Read-only.

## TECHNICKY

Odhad spotřeby paměti Gaussian bufferu jako Gaussian count × 616 bajtů (formátováno v memory stylu). 616 bajtů je empirická velikost plně vybaveného Gaussianu (pozice, scale, rotation, opacity, SH koeficienty stupeň 3, gradient akumulátor). Zobrazení **nezachycuje** renderer overhead (tile buffer, sort buffer, backward buffer) — reálná spotřeba GPU paměti leží typicky 2-3× nad touto hodnotou. U 500K Gaussianů: zobrazeno ~290 MB, reálně ~700 MB. U 1.5M Gaussianů: zobrazeno ~880 MB, reálně ~2.5 GB. Na M3 Max s 64+ GB Unified Memory nekritické, na M3 Pro s 18 GB už limit.

## JEDNODUŠE ŘEČENO

Odhad, kolik GPU paměti zabírají samotné Gaussiany — zhruba 616 bajtů na bod. Skutečná spotřeba GPU je 2-3× vyšší než zobrazeno, protože renderer si ještě přidává vlastní tile, sort a backward buffery. U MacBooku s 16-18 GB Unified Memory bys měl zůstat pod 500K Gaussianů; s M3 Max nebo Studiem (64+ GB) můžeš v klidu jet 1.5M a víc. Pokud trénink náhle padá nebo systém swapuje, většinou je tady dosažen limit — Render Scale (I22) dolů nebo Densify Until (I20) redukovat.

**I36 Speed**

Inspektor → Metriky → Speed. Read-only.

**TECHNICKY**

Iterace za sekundu s jedním desetinným místem („24.3 it/s“). Vypočítáno trainerem jako klouzavý průměr přes posledních ~100 iterací. Typické hodnoty: - Quick předvolba (1K iter): 80-120 it/s (krátké, žádný steady-state) - Classic 20K @ 1.0 Render Scale (Truck scéna, 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 (pomalejší kvůli relo-kaci) - U 1M+ Gaussianů a plného rozlišení: < 10 it/s  
Klesající Speed v průběhu tréninku je normální — víc Gaussianů = víc compute na iteraci. Náhlé propady (např. ze 30 → 5 it/s) naznačují GPU thermal throttling nebo konkurující aplikace.

**JEDNODUŠE ŘEČENO**

Jak rychle trénink běží, v iteracích za sekundu. Stojí typicky na 20-50 it/s, při redukovaném Render Scale (50 %) a malých scénách občas 80-120 it/s. Klesá v průběhu tréninku zcela přirozeně, protože víc Gaussianů = víc práce na iteraci. Náhlé propady (např. 30 → 5 it/s) naznačují GPU thermal throttling nebo konkurující aplikace — prohlížečové záložky s videem, Time Machine backup, indexování Photos. Pomáhá držet aplikaci v popředí a zavřít programy v pozadí. U 1M+ Gaussianů a plného rozlišení jsou pod 10 it/s normální.

**I37 Elapsed**

Inspektor → Metriky → Elapsed. Read-only.

**TECHNICKY**

Již uplynulý čas jako „4:23“ (m:ss) nebo „1:23:45“ (h:mm:ss). Přepínání formátu od 1 hodiny. Měří pouze čistý tréninkový čas, ne předcházející fáze (SfM výpočet, import obrázků). Při pauze/resume hodiny běží dál — je to tedy wall-clock, ne CPU čas.

**JEDNODUŠE ŘEČENO**

Jak dlouho trénink už běží, jako čistá stopka (wall-clock čas). Formát je „m:ss“, do jedné hodiny, poté „h:mm:ss“. Ne „CPU čas“, ale „jak dlouho už čekáme“ — tedy i pauzy se počítají. Měří pouze čistou tréninkovou fázi, ne předcházející SfM výpočet nebo import obrázků. Užitečné pro porovnání s ETA (I38) — pokud Elapsed ztlačně přestřelí původní ETA, trénink se někde stal pomalejším, než bylo plánováno.

## I38 ETA



Inspektor → Metriky → ETA. Read-only.

## TECHNICKY

Odhad zbývajícího času jako „17:42“ nebo „1:12:35“. Výpočet:  $(\text{Max Iterations} - \text{aktuální iterace}) / \text{iterace za sekundu}$ . Ukazuje „-“, pokud je Speed momentálně nula (úplně na začátku nebo při pauze). Odhad se **nepřizpůsobuje** typickému zpomalení ke konci tréninku — právě u MCMC a Classic s velkými Densify Until hodnotami má trénink tendenci zpomalovat, protože ke konci přicházejí do obrazu stále další Gaussiány. Reálně zůstává typicky 10-20 % nad počáteční ETA.

## JEDNODUŠE ŘEČENO

Jak dlouho ještě pravděpodobně čekat — vypočítáno ze zbývajících iterací a aktuální Speed (I36). Hrubý odhad: pokud Mac náhle zpomalí (více Gaussianů od densifikační fáze, thermal throttling, jiné aplikace), může to trvat déle než zobrazeno. Aplikace nepočítá typické zpomalení ke konci tréninku, proto skutečný konec přistává obvykle 10-20 % nad počáteční ETA. Připočti 15 %, pak to obvykle sedí. Ukazuje „-“, pokud je Speed momentálně 0 (začátek tréninku nebo pauza).

## Sekce Loss Diagram (I39–I41)



Obrázek 14: Crop pouze sekce Loss History po dokončeném tréninku — Current 0.0064, Min 0.0035 (zeleně), modrý průběh od 0.027 (Iterace 1) na 0.0035 (Iterace 5K) s charakteristickým zlomem kolem Iter 200, pod tím chart Gaussian Count oranžově

Sekce Loss Diagram vizualizuje tréninkový průběh přes čas. Sestává ze dvou chartů: Loss Curve chartu (velký, nahoře, modrý) a Gaussian Count chartu (menší, dole, oranžový). Oba se staví živě během tréninku a persistují do dalšího startu tréninku. Před prvním tréninkem je oblast prázdná („Loss curve will appear during training“). Charty jsou čisté SwiftUI path drawingy (žádný framework Swift Charts), aby plyně renderovaly i při 100K+ bodech.

### I39 Current Loss (zobrazení)



KDE

Inspektor → Loss Diagram → levá label oblast „Current: 0.0287“. Read-only.



TECHNICKY

Float hodnota posledního loss sample bodu, formátovaná se čtyřmi desetinnými místy. Identické s I31 (Loss v sekci Metrics), jen zde kompaktněji formátováno. Zdrojem je Loss History — seznam, který každých ~30 iterací dostane záznam. Pouze konečné hodnoty se do seznamu přidávají — NaN/Infinity (velmi vzácné, v případě bugu gradient explosion) jsou filtrovány.

#### JEDNODUŠE ŘEČENO

Aktuální hodnota lossu v kratším zápisu než v sekci Metrics (čtyři desetinná místa). Obsahově identické s I31, ale tady sedí zobrazení přímo u loss chartu a dává ti při sledování křivky přesnou číselnou hodnotu. Aktualizováno jako všechny live metriky každých ~30 iterací. NaN nebo Infinity hodnoty (extrémně vzácné u bugů gradient explosion) aplikace automaticky filtruje. Užitečné, abys při pohledu na diagram nemusel skákat do druhé sekce.

### I40 Min Loss (zobrazení)



KDE

Inspektor → Loss Diagram → pravá label oblast „Min: 0.0245“ (zeleně). Read-only.



TECHNICKY

Minimum všech kdy viděných hodnot loss aktuálního tréninkového běhu. Přepočítává se živě z Loss History — žádná samostatná persistence. Zobrazuje se zeleným písmem, protože „Min“, znamená „Best so far“. Zelená přerušovaná čára na dolním okraji chartu značí tuto Y pozici vizuálně. Při Continue Training relacích sledování minima začíná znovu — stará history se v UI nahrazuje novou (ne připojuje). Pokud aktuální trénink běží hůř než předchozí, může být Min zobrazení tedy větší než předchozí konečný výsledek.

#### JEDNODUŠE ŘEČENO

Nejnižší hodnota lossu, kterou tento trénink zatím viděl — zobrazeno zeleně, protože „best so far“. Zelená přerušovaná čára na dolním okraji chartu značí tuto pozici i vizuálně. Pokud aktuální křivka leží zrovna znatelně nad ní, je se štěstím ještě zlepšení; většinou je ale Min indikátor konečného výsledku, který tě později zajímá. Při Continue Training relacích sledování Min začíná znovu, protože stará history se v UI nahrazuje novou — Min hodnota tím může vypadat hůř než předchozí konečný výsledek.

## I41 Gaussian Count Chart



Inspektor → Loss Diagram → druhý chart pod tím (oranžový). Read-only.



Liniový diagram počtu Gaussianů přes tréninkové iterace. Zdroj: Gaussian Count History (seznam dvojic (Iter, Count), naplňovaný trainerem každých ~30 iter). Y škála dynamická mezi minimem a maximem history. U Classic strategie křivka typicky vypadá takto: stále rostoucí až do Densify Until, poté plochá (s malými pruningovými výkyvy). U MCMC: strmý nárůst až k capu, poté horizontální čára (relokace drží počet konstantní). Pokud křivka **klesá** navzdory aktivnímu tréninku, densifikace příliš agresivně prune — indikátor špatných defaultů nebo známý MCMC collapse bug (téma hotfixu v1.4.4).

### JEDNODUŠE ŘEČENO

Jak se počet Gaussianů vyvíjí přes tréninkový čas — menší oranžový chart pod loss křivkou. U Classic strategie linie stoupá stabilně, dokud není dosaženo Densify Until (I20), poté zůstává plochá s malými pruningovými výkyvy. U MCMC vyrazí strmě nahoru na cap a poté zůstává horizontální, protože relokace drží počet konstantní. Pokud křivka navzdory aktivnímu tréninku náhle ohne dolů, densifikace pruning příliš agresivně — klasický příznak MCMC collapse bugu z v1.4.4. Pak pomáhá update aplikace nebo přechod zpět na Classic.

### Jak číst loss křivku?

Loss chart je nejdůležitější diagnostický nástroj v Inspektoru — žádný jiný indikátor neukazuje tak přímo, zda trénink užitečně postupuje nebo visí. Typický zdravý tvar je rychlý pokles v prvních 1000-3000 iteracích (z ~0.15 na ~0.05), následovaný pomalým rovnoměrným poklesem až do konce tréninku (na 0.020-0.030). Logaritmicky působí křivka jako hladká diagonála.

**Co znamená plató u lossu?** Pokud křivka zůstává přes několik tisíc iterací plochá, jsou dvě možná čtení: (a) trénink je „konvergoval“, — loss už nemůže významně klesat, protože model je tak dobrý, jak může s danými daty a nastavením být. To je žádoucí; to je „hotovo“. (b) trénink „visí“, — loss by ještě mohl klesat, ale optimalizace stagnuje (lokální minimum, learning rate příliš malá, densifikace vypnutá). Rozlišování: pokud hodnota lossu leží v typicky dobrém rozsahu (0.020-0.030 u indoor/object, 0.040-0.060 u outdoor) a křivka je už 5K iterací plochá, je to konvergováno. Pokud je hodnota výrazně vyšší než u srovnatelných scén (např. 0.08), visí.

**Pozor Gaussian plató ≠ Loss plató.** Plató v počtu Gaussianů **neznamená** „trénink je hotový“. Znamená pouze, že densifikace přestala přidávat nové body — buď protože je dosaženo (Classic), nebo protože je MCMC cap plný. Trénink běží poté dál a zjemňuje pouze existující body. Skutečný signál „hotovo“ čteš z loss křivky a z iteračního zobrazení (I30), ne zde.

**Pravidlo pro přerušení:** pokud loss křivka po 5000+ iteracích leží nad 0.08 a sotva ještě klesá, s velkou pravděpodobností je SfM rekonstrukce špatně. Trénink přerušit, v Kapitole 9 nahlédnout, zda zvolený SfM backend odpovídá scéně, popř. přepnout na COLMAP/Native, pak znovu spustit. Raději investovat 10 minut do lepšího SfM než 2 hodiny tréninku se špatným zarovnáním kamer.

## Kdy sáhnout po Inspektoru?

Rychlá reference: která sekce + které ovládací prvky pro který typický use case?



## KAPITOLA

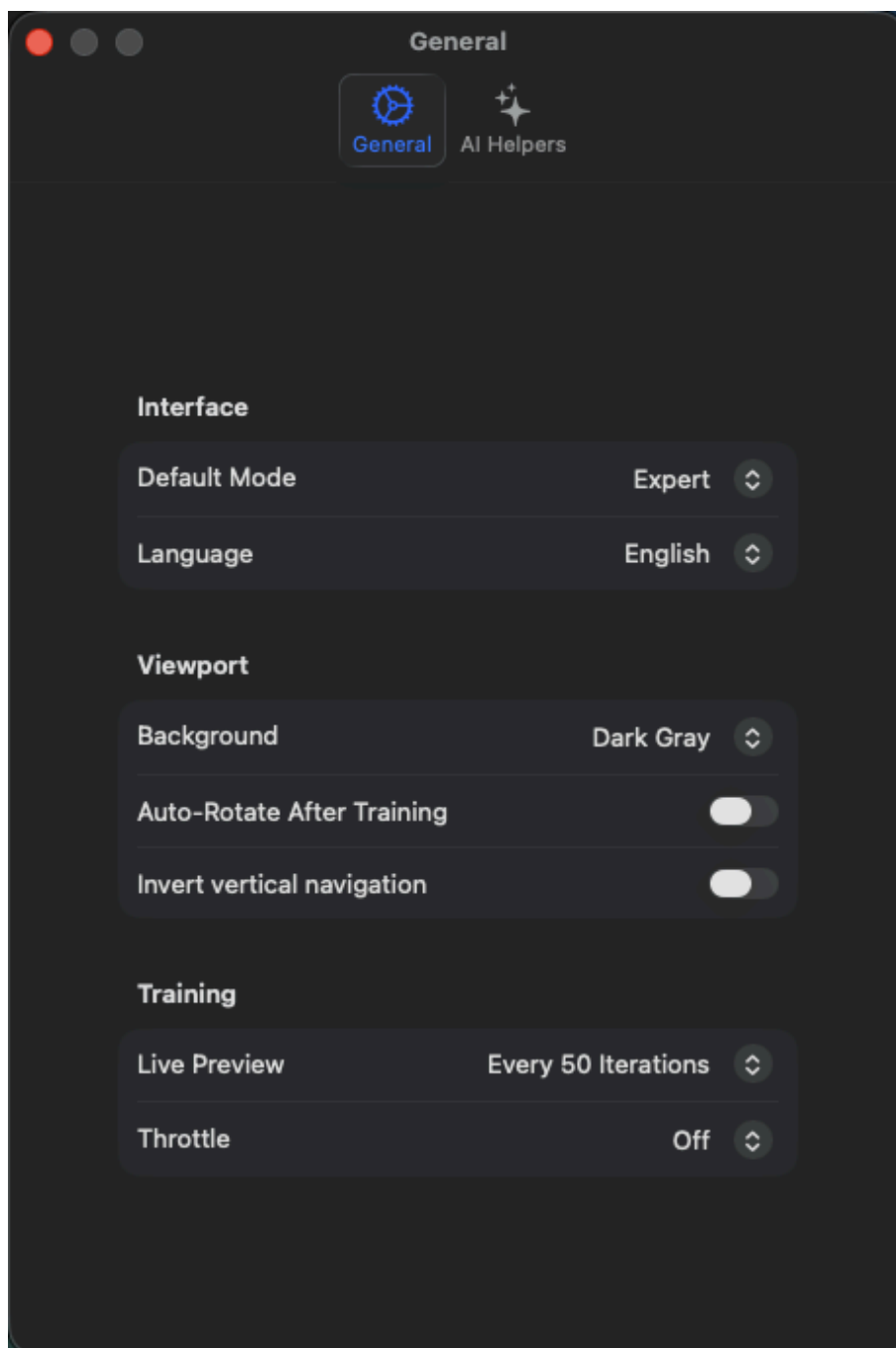
## Kapitola 3 — Nastavení

---

Okno Nastavení se otevírá přes RadianceKit → Nastavení... nebo standardní klávesovou zkratkou `⌘, .` Obsahuje dvě záložky: **General** a **AI Helpers**. Na rozdíl od hodnot Inspektoru z Kapitoly 2 působí nastavení z tohoto okna **globálně v aplikaci** (napříč všemi projekty) — ukládají se a přežívají restarty aplikace. Záložka General sdružuje tři obsahové oddíly: Rozhraní, Náhled a Trénink. Záložka AI Helpers zapíná on-device strojové učení (Vision, CoreML) pro předzpracování SfM a tréninku. (Tři přepínače pro redukci outdoorových floaterů — Sky Masking, Mid-Training Floater Cleanup, Reconstruct Sky Dome —, které dříve sídlily zde, se počínaje verzí v1.6 přesunuly do sekce Enhancements v Expert-Inspektoru, kde se nyní ukládají per projekt; viz Kapitola 2, 142–144.)

Dřívější ovládací prvky pro hromadnou aktivaci nebo deaktivaci všech AI Helperů v aktuální verzi již neexistují — odpovídajícím způsobem zde nejsou dokumentovány. Také dřívější oblast „Coming Soon“, pro ještě nedodané helpery byla odstraněna a zde se na ni neodkazuje.

## Záložka General



Obrázek 15: Nastavení → Záložka General s Rozhraním, Náhledem, Tréninkem a experimentální sekci

## S1 Default Mode



Nastavení → General → Rozhraní → Default Mode picker. Bound: Default: `.simple`.



Řídí, ve kterém ze dvou UI režimů se aplikace otevře po příštím spuštění. „Simple Mode„ je řízený wizardový workflow ve 4 krocích (Import → Processing → Preview → Export, dokumentováno v Kapitole 10 pod Z1–Z4), „Expert Mode“ klasické tří-panelové rozložení s Navigátorem, 3D náhledem a Expert-Inspectorem z Kapitoly 2. Hodnota se pamatuje přes restarty. Identický účinek jako menu Mode → Simple Mode (⌘1) / Mode → Expert Mode (⌘2), jen že menu přepíná běžící relaci, zatímco tento picker nastavuje default pro budoucí relace. Oba režimy přistupují ke stejnému projektovému stavu — projekty, kamery a konfigurace tréninku zůstávají při změně režimu zachovány. Toolbar tlačítka specifická pro daný režim se okamžitě překreslí.

### JEDNODUŠE ŘEČENO

Zde vybíráš, s jakým rozhraním RadianceKit nastartuje při příštím spuštění. „Simple Mode„ je režim pro začátečníky: čtyři jasné kroky, předem nastavené hodnoty, minimum voleb. „Expert Mode“ je plná sada nástrojů se všemi posuvníky, které vidíš v Kapitole 2. Kdykoli můžeš přes menu „Mode„ přepínat sem a tam, aniž by se ztratily obrázky nebo postup tréninku.

## S2 Language



Nastavení → General → Rozhraní → Language picker. Bound: Default: `.system` (sleduje jazyk macOS).



Vybírá jazyk zobrazení celého UI aplikace, nezávisle na systémovém jazyce macOS. RadianceKit je lokalizován do 17 jazyků (`de`, `en`, `pl`, `en-AU`, `ar-SA`, plus 12 dalších). Při „System„ aplikace sleduje jazyk macOS. Při explicitním výběru se nastavení jazyka pamatuje přes restarty; úplného účinku obvykle dosáhne až po restartu aplikace, protože lokalizační bundly se načítají pouze při startu. Všechny 298 dokumentovaných lokalizačních klíčů v projektu se zohledňuje, včetně všech textů v sub-views a toltipech.

### JEDNODUŠE ŘEČENO

Pokud tvůj Mac běží v angličtině, ale ty bys raději chtěl české rozhraní RadianceKitu (nebo naopak), nastavíš to zde. Většina textů se vymění okamžitě. Některé dialogy se v novém jazyce objeví až po restartu aplikace.

### S3 Viewport Background



KDE

Nastavení → General → Náhled → Background picker. Bound.: Default: `.darkGray` (RGB 0.1, 0.1, 0.1).



TECHNICKY

Nastavuje výchozí barvu pozadí pro 3D náhled. Tři možnosti: „Dark Gray“, (RGB 0.1, 0.1, 0.1 — default), „Black“ (0, 0, 0) a „White“, (1, 1, 1). Nastavení uchová default pro nové projekty a relace přes restarty a současně okamžitě aktualizuje běžící Metal renderer. Identické možnosti najdeš v menu Viewport → Background (M21, M22, M23), ale Settings picker nastavuje default, zatímco menu přepíná běžící zobrazení. Důležité pro snímky obrazovky a demo videa: bílá pozadí silněji zvýrazňují zelené/modré floatery, tmavá pozadí jsou lepší pro čisté renderovací záběry.

#### JEDNODUŠE ŘEČENO

Barva za tvými 3D modely v náhledovém okně. Tmavě šedá je standard a hodí se pro většinu scén. Bílá je dobrá pro snímky obrazovky, černá působí elegantněji u renderovacích záběrů. Barvu můžeš kdykoli přepnout přes menu „Viewport → Background“, pro běžící scénu — toto nastavení pouze určuje, která barva má být aktivní při příštím otevření.

### S4 Auto-Rotate After Training



KDE

Nastavení → General → Náhled → Toggle „Auto-Rotate After Training“, Bound.: Default: `false`.



TECHNICKY

Bezprostředně po konci tréninku spustí spojitou otáčivou rotaci kamery náhledu kolem těžiště scény (standardní rychlost rotace ~0.3 rad/s). Praktické pro demo relace, A/B porovnání a pro přímé posouzení z 360° pohledu, zda vznikly „floatery“, na okraji scény. Vizualně je efekt shodný s menu Viewport → Toggle Auto-Rotation (M16,  $\text{⌘} \backslash T$ ), jen že tento toggle spouští chování automaticky po konci tréninku místo ručně. Dá se kdykoli později přerušit přes menu nebo kliknutím do náhledu (což rotaci pauzuje). Nemá vliv na výkon tréninku — rotace běží až poté, co je trénink hotový.

#### JEDNODUŠE ŘEČENO

Pokud je zapnuté, 3D scéna se automaticky otočí, jakmile je trénink hotový — jako kolotoč. Hezké, pokud při nočním tréninku ráno hned vidíš výsledek v pohybu, aniž bys musel sám klikat. U dlouhých relací, kdy trénink jen hlídáš, ho raději nech vypnutý.

**S5 Live Preview Interval**

Nastavení → General → Trénink → Live Preview picker. Bound: `AppState.trainingConfig.livePreviewInterval`. Default: 0 (Off).

 TECHNICKY

Určuje, v jakém intervalu iterací se aktuální snapshot tréninku vykresluje do 3D náhledu. Čtyři diskrétní hodnoty: 0 („Off“), 50, 250, 1000 iterací. Při aktivním Live Preview kopíruje trainer Gaussian buffer z GPU do samostatného render bufferu a spouští překreslení náhledu. Při „Off“ se náhled aktualizuje až po dokončení tréninku. Nákladnost na výkon: každých 50 iterací ~5–10 % pomalejší na M3 Ultra, každých 250 iterací ~1–2 % pomalejší, každých 1000 iterací neměřitelně. Paměťový overhead je konstantní ~2 GB pro snapshot buffer, nezávisle na intervalu. Hodnota slouží jako default pro nové tréninky; po spuštění tréninku ukazuje trénink-inspektor reálnou live hodnotu daného tréninku. Při intervalu 50 je vizuální dojem plynulého „vyrůstání“ mraku bodů, při 1000 to působí trhaně.

 JEDNODUŠE ŘEČENO

Zatímco trénink běží, můžeš zvolit, jak často se 3D pohled aktualizuje. „Off“ znamená: žádná aktualizace během tréninku (nejrychlejší). „Every 50 Iterations“ ukazuje téměř v reálném čase, jak scéna vzniká (trochu pomalejší). Pro pohodlné sledování malých tréninků je „Every 250“ dobrý kompromis.

## S6 Throttle Delay



Nastavení → General → Trénink → Throttle picker.  
Bound: `AppState.trainingConfig.throttleDelayMs`.  
Default: 0 (Off).

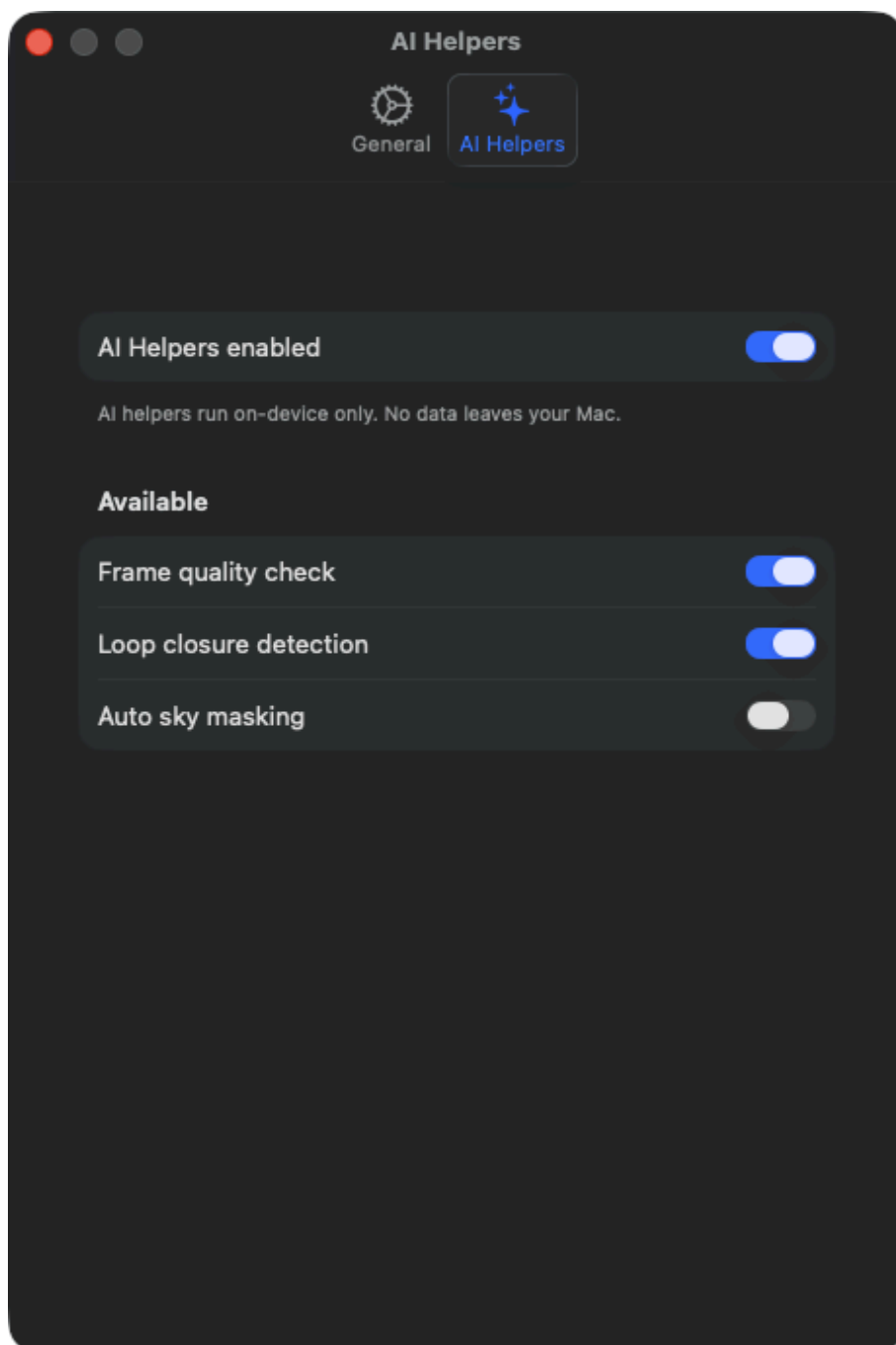


Vkládá mezi iterace tréninku umělé zpoždění v milisekundách. Čtyři diskrétní hodnoty: 0 („Off“), 2 („Light“), 5 („Moderate“), 10 („Eco“). Smysl: u delších tréninků (několik hodin) je GPU jinak vytížena na 100 %, což vede ke zdatelně pomalejšímu systémovému UI (kurzor myši trhá, jiné aplikace se zpomalí). Throttle zpoždění dává GPU pauzy, ve kterých mohou běžet jiné úkoly. Náklady na výkon jsou značné: při 5 ms throttle trvá typický 40K trénink asi 50–80 % déle než bez throttle. V režimu Performance „Eco“ (10 ms) je zpoždění na iteraci delší než iterace sama — faktor 2–3× pomalejší. Při aktivním throttle se pod pickerem objeví poznámka: „Throttle is on. Training will be slower than usual.“ Sama aplikace nereaguje zdatelně lépe — profitují pouze jiné aplikace.

### JEDNODUŠE ŘEČENO

Pokud se ti Mac během dlouhého tréninku přehřívá nebo jiné programy začínají být pomalé, zapni zde brzdu. „Off“ dává GPU plný plyn (nejrychlejší). „Light“ dělá malou pauzu mezi každým krokem (trochu pomalejší, ale systém reaguje lépe). „Eco“ je nejsilnější brzda — dobré pro noční tréninky na MacBooku, které se nemají moc zahřát.

## Záložka AI Helpers



Obrázek 16: Nastavení → Záložka AI Helpers s hlavním přepínačem a sub-toggly

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

Nastavení → AI Helpers → první sekce → Toggle „AI Helpers enabled“, Bound: Default: `true`.

 TECHNICKY

Master přepínač přes všechny funkce AI Helpers v pipeline. Pokud je vypnutý, import a SfM pipeline přeskakuje všechny ML předzpracovací fáze úplně — žádné volání Apple Vision, žádné načítání CoreML modelu, žádné probouzení NPU. Pokud je zapnutý, konzultují se individuální sub-toggly (S12–S13). Hodnota se pamatuje přes restarty. Působí na následující fáze: (a) Frame quality pre-check před SfM (S12), (b) detekce uzavření smyček (S13). Důležité: při vypnutém jsou dva sub-toggly deaktivovány a vizuálně zašedlé. Poznámka v patičce zdůrazňuje, že všechny AI Helpers běží striktně on-device — žádný upload obrázků, žádné cloudové zpracování. Záruka soukromí pochází z výhradního použití frameworku Apple Vision (lokálně na Neural Engine) a CoreML modelů, které leží přímo v balíku aplikace.

 JEDNODUŠE ŘEČENO

Hlavní přepínač pro všechny funkce, které interně používají AI / strojové učení. Default je „on“, protože helpery šetří hodně času, aniž by tvé obrázky opustily Mac. Pokud je chceš mít úplně vypnuté (např. kvůli šetření energie nebo protože tvůj Mac nemá NPU), vypni je zde — pak dvě pod-volby níže automaticky zesřednou a nic nedělají.

**S12** Frame quality check

Nastavení → AI Helpers → sekce Available → Toggle „Frame quality check„. Bound: . Default: `true` .

 TECHNICKY

Aktivuje Frame Quality Screener (fáze 3.11), který před voláním SfM analyzuje každý importovaný frame. Kroky pipeline na frame: (a) Laplacian Variance filtr z Apple Vision (detekce rozmazání — threshold ~150), (b) histogramová kontrola over/under-exposure (threshold: >5 % pixelů u 0 nebo 255), (c) detekce prázdného frame (směrodatná odchylka < 5 přes všechny pixely). Frame, který projde všemi třemi kontrolami, jde přímo dál. Frame, který selže v alespoň jedné kontrole, vyvolá modální potvrzovací dialog, který listuje každý problematický frame s thumbnailem a důvodem a ptá se, zda má být odstraněn. Důležité: žádné automatické mazání — dialog je vždy vyžadován, uživatel si ponechává poslední rozhodnutí. Výkon: ~50 ms na frame na M3 Ultra, běží paralelně. Při vypnutém se všechny framy předávají do SfM bez kontroly. Při deaktivovaném masteru (S11) je toggle vizuálně zašedlý a bez účinku. Dodaný status dle paměti: SHIPPED 2026-05-23.

 JEDNODUŠE ŘEČENO

Před samotným tréninkem se aplikace dívá na každou fotografii: je rozhýbaná? úplně tmavá nebo bílá? prázdná? Pokud ano, zeptá se tě, jestli chceš obrázek vyhodit — nikdy nemaže automaticky. Šetří to později mnoho hodin, protože jediný úplně rozhýbaný obrázek může občas trénink celý pokazit. Default je „on“, protože náklad je téměř nulový a užitek velký.

**S13 Loop closure detection**

KDE

Nastavení → AI Helpers → sekce Available → Toggle „Loop closure detection“,. Bound:. Default: `true` .



TECHNICKY

Aktivuje detekci uzavření smyčky založenou na Apple Vision Feature Print. Pro každý importovaný frame se počítá ~768 dimenzionální feature vektor, který představuje neuronový embedding obsahu obrazu. Následně se všechny Feature Prints porovnávají párovým způsobem přes cosine similarity. Páry s similarity  $> 0.85$  a vzdáleností v indexu framů  $> 50$  (tedy nesousední framy) jsou identifikovány jako „loop closure kandidáti“, a zapsány do sidecar JSONL souboru v projektové složce. Pouze informační — importovaná sekvence obrázků se nemění. Smysl: dává SfM solveru (zejména COLMAP) nápovědu, že tyto framy patří k sobě v 3D prostoru. Pro nativní SfM je sidecar informace aktuálně pouze dokumentární; COLMAP používá hinty interně přes custom matches-file (ruční integrace možná, není automaticky propojena). Výkon: ~200 ms na frame na M3 Ultra, běží paralelně. Při vypnutém se Feature Prints negenerují. Při deaktivovaném masteru (S11) vizuálně zašedlé.

**JEDNODUŠE ŘEČENO**

Když při fotografování obcházíš objekt a na konci přistaneš zpátky na výchozím bodě, pomáhá to počítači obrovsky to vědět. Tato volba automaticky pozná, které z tvých fotografií byly pořizeny „téměř ze stejného místa“, a zapíše to do malého pomocného souboru. SfM nástroje (především COLMAP) tuto informaci umí použít k čistší 3D rekonstrukci. Default je „on“, protože to běží bez tvého zásahu a nic na tvých obrázcích nemění.

**Settings zrcadlené z Inspektoru**

Ostatní záznamy nastavení (S17–S33) z inventární tabulky jsou zrcadlené z Expert-Inspektoru a jsou dokumentovány v Kapitole 2 (Inspector Controls I12–I29). Neobjevují se fyzicky v okně Nastavení, v inventáři byly uvedeny pouze proto, že běží přes `TrainingConfig` properties, které jsou persistovány a tudíž mají formálně charakter nastavení. Pro obsahová vysvětlení viz tam.

## Kdy co?

Nastavení	Rozsah platnosti	Persistence
S1 Default Mode	Globální	Restart aplikace
S2 Language	Globální	Restart aplikace
S3 Viewport Back-ground	Globální (Default) + Runtime	Restart aplikace
S4 Auto-Rotate After Training	Globální	Restart aplikace
S5 Live Preview Interval	Default pro nové tréninky	Restart aplikace
S6 Throttle Delay	Default pro nové tréninky	Restart aplikace
S11 AI Helpers Master	Globální	Restart aplikace
S12 Frame quality check	Globální	Restart aplikace
S13 Loop closure detection	Globální	Restart aplikace

Globální = působí na všechny projekty. Default pro nové tréninky = působí pouze na příští založený trénink, běžící relace zůstávají nezměněny. Aktuální trénink = působí okamžitě na běžící konfiguraci tréninku, ale nepersistuje bez explicitního reimportu.

## KAPITOLA

## Kapitola 4 — Pomocná okna

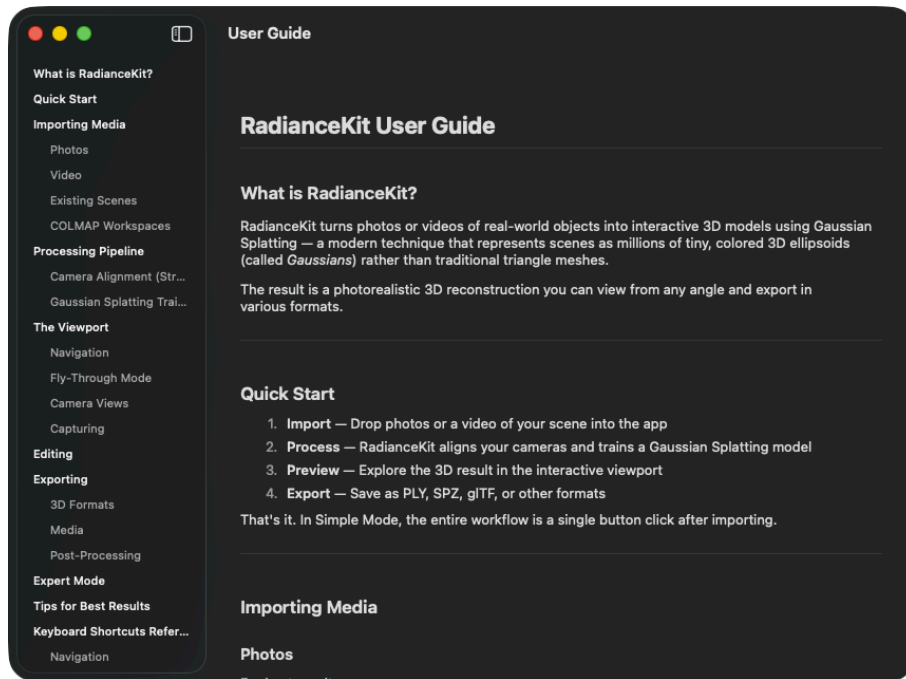
---

Vedle hlavního okna (3D náhled plus Inspektor) RadianceKit spravuje sedm dalších oken, která se všechna otevírají přes menu Help. Seznam shora dolů: User Guide (⌘?), Keyboard Shortcuts (⌘/), Open Training Logs... (neotevívá okno aplikace, ale Finder; proto zde dále neřešeno), Manage Storage..., Pareto Dashboard... (⌘⌘D), Holdout Analysis... (⌘⌘H), BayesOpt Console... (⌘⌘B). Tři z nich — Dashboard, Holdout, BayesOpt — jsou samostatné analytické nástroje. Každý má svůj vlastní view model stack, čte nebo zapisuje JSON soubory na disk, a pro každý existuje CLI argument, kterým můžeš nechat otevřít okno hned při startu aplikace na konkrétní soubor ( `--dashboard-dir` , `--holdout-file` , `--bayesopt-autorun` ).

Čtyři jednoduchá okna (User Guide, Keyboard Shortcuts, Manage Storage, plus podpoložky Open Training Logs / Open Exports Folder) dostávají na ovládací prvek krátký záznam. Tři analytická okna jsou dokumentována podrobněji — vždy s úvodem, který vysvětluje, co v okně vidíš, kdy ho otevřít a jak interpretovat zobrazený obraz.

Na konci kapitoly je oddíl s odkazy na Inspektor hlavního okna: co v live loss chartu a v zobrazení Gaussian Count během běžícího tréninku můžeš smysluplně odečíst.

## User Guide (W1–W4)



Obrázek 17: Okno User Guide se sidebar vlevo a renderovaným Markdown obsahem vpravo

**Co to je:** Vestavěné okno nápovědy, které renderuje s aplikací dodávaný `guide_<jazyk>.md`. Jazyk se odvozuje z Nastavení (záložka General → Language) nebo, pokud tam stojí „System“, z jazykových preferencí macOS. Layout je klasický: vlevo sidebar se všemi nadpisy, vpravo plynulý text.

**KDY OTEVŘÍT** Když potřebuješ rychlou připomínku jednotlivého bodu — tedy jako náhradu klíčových slov. Podrobná reference je tato příručka; vestavěné okno nápovědy je spíše to, co by bylo `--help` v příkazové řádce. Aktualizuje se s každým releasem aplikace, ale obsahově zůstává povrchnější.

## W1 NavigationSplitView (Sidebar + Detail)



Help → User Guide (⌘?)..

### TECHNICKY

Dvouloupcový layout s úzkým sidebarem (minimálně 180 pt široký) pro strom obsahu a scrollovatelnou detailní oblastí pro samotný Markdown obsah. Okno má minimální velikost 700 × 500 pt. Při prvním otevření okno načítá příslušný `guide_<lang>.md` z balíku aplikace (fallback `guide_en.md`), parsuje jej do blokových záznamů (nadpisy H1–H4, odstavce, seznamy, tabulky, oddělovače) a samostatně extrahuje strukturu nadpisů pro sidebar. Inline formátování (bold, italic, code span) se renderuje přes vestavěný markdown engine. Jazyk se čte z nastavení aplikace, se speciálním případem čínštiny (`zh-Hans`) a brazilské portugalštiny (`pt-BR`), které se zachovávají jako plné locale tagy, protože se tyto varianty odlišují od `zh` resp. `pt`.

### JEDNODUŠE ŘEČENO

Vestavěný text nápovědy, vlevo seznam témat, vpravo obsah. Jazyk se nastavuje automaticky podle systémových nastavení. Funguje offline, ale je záměrně jen zkrácenou verzí — úplnou referencí je tato příručka.

## W2 List (Heading sidebar)



Levý sloupec v okně User Guide..

### TECHNICKY

Seznam přes všechny nadpisy H2 a H3 aktuálního Markdown dokumentu. H2 položky se objevují bez odsazení s medium váhou písma, H3 položky s 16 pt odsazením vlevo a redukováným foreground stylem. H4 a hlubší jsou ignorovány, protože hloubka by jinak udělala sidebar nepřehledným. Kotvící ID se generují z textu nadpisu sluggifikací (lowercase + mezery na pomlčky + filtrování na písmena/čísla/pomlčky — stejný algoritmus, který GitHub používá pro své Markdown kotvy, takže by i externí URL k dokumentaci potenciálně přistály na stejné kotvě). Seznam používá nativní macOS styl.

### JEDNODUŠE ŘEČENO

Navigační lišta na levé straně. Klikni na položku a skočíš na sekci.

### W3 Button (Heading → skok na kotvu)



Na řádek sidebaru jedno tlačítko..

#### TECHNICKY

Každá položka sidebaru je tlačítko, které nastavuje aktuální kotvu, ale opticky vypadá jako položka seznamu. Pozorovací proměnná pak spouští scroll skok k odpovídající kotvě s měkkou animací přes 0,3 s. Po skoku se hodnota kotvy resetuje, aby další klik na stejnou kotvu znovu vystřelil (jinak by pozorovatel nestřelil znovu, protože hodnota se nezměnila).

#### JEDNODUŠE ŘEČENO

Kliknutí tě vede na příslušné místo v textu vpravo.

### W4 ScrollView (detail obsahu)



Pravý sloupec..

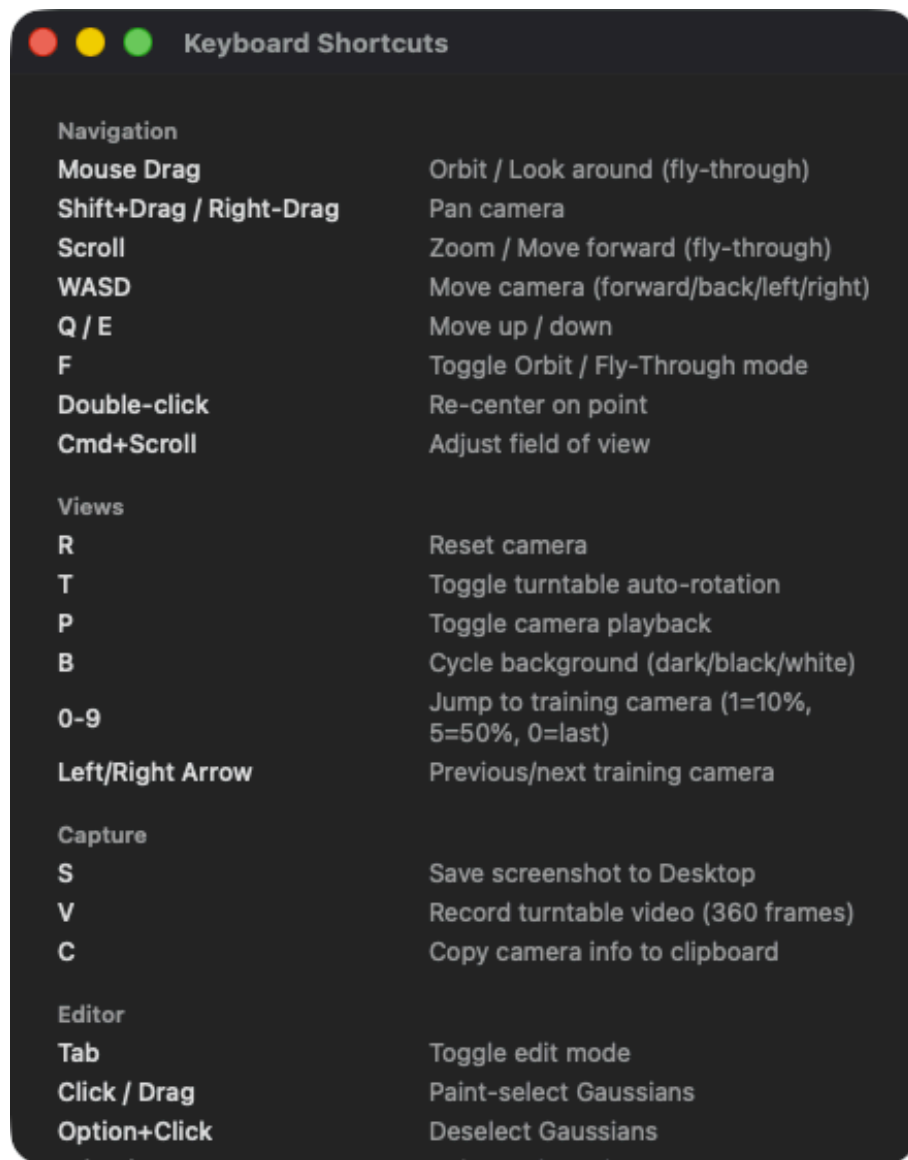
#### TECHNICKY

Scrollovatelná, vertikálně stohovací oblast obsahu s lazy renderováním, protože delší průvodci mohou snadno mít přes 200 Markdown bloků — ne-lazy varianta by všechny instancovala najednou. Každý blok dostává vlastní ID, buď heading kotvu (pro skočitelné H1–H3) nebo index placeholder. Maximální šířka je 720 pt, padding 32 horizontálně / 24 vertikálně, aby si dlouhé řádky zachovávaly dobře čitelný layout. Tabulky se renderují buňka po buňce s horizontálními stacks a oddělovacími čarami; inline kód přes vestavěný Markdown engine. Skutečné kódové bloky se aktuálně zachází jako s odstavcem — známé omezení okna nápovědy.

#### JEDNODUŠE ŘEČENO

Samotný text nápovědy. Scrollovatelný, dobře čitelná šířka, jasná typografie.

## Keyboard Shortcuts (W5–W6)



Obrázek 18: Okno Keyboard Shortcuts — pět skupin Navigation/Views/Capture/Editor/Training se sloupcem hotkey vlevo a popisem vpravo

**CO JE NA OBRÁZKU** Statický referenční seznam v pěti sekcích. **Navigation:** Mouse Drag (Orbit/Fly), Shift+Drag/ Right-Drag (Pan), Scroll (Zoom), WASD (fly-through pohyb), Q/ E (Up/Down), F (Toggle Orbit/Fly), Double-click (Re-center), Cmd+Scroll (úprava FoV). **Views:** R (Reset Camera), T (Auto-Rotation), P (Camera Playback), B (Background cycle), 0–9 (skok na training cam 1=10 %/5=50 %/0=last), Left/Right Arrow (Prev/Next Cam). **Capture:** S (Screenshot na desktop), V (Turntable video), C (Copy Camera Info). **Editor:** Tab (Edit režim), Click/Drag (paint select), Option+Click (deselect), X / Delete (smazat výběr), Cmd-Z (vrátit poslední mazání), [ / ] (zmenšit/zvětšit štetec), Esc (zrušit výběr). **Training:** Start, Pause/Resume, Cancel, Continue +5K/+10K/+20K přes shortcuty menu v M9–M14.

**Co to je:** Jednoduchý statický přehled všech klávesových zkratk — Navigation, Views, Capture, Editor, Training. Obsah je napevno zakódován, žádné Markdown načítání.

**KDY OTEVŘÍT** Když hledáš nejrychlejší cestu, jak něco udělat v náhledu. WASD fly through, R pro reset kamery, B pro cyklování pozadí — všechny tu stojí.

### W5 ScrollView (oblast obsahu)

 KDE

Help → Keyboard Shortcuts (⌘/)..

 TECHNICKY

Jednoduchá scroll oblast s vertikálním seznamem uvnitř. Padding 20 dokola, žádný sidebar navigation tree (seznam je dostatečně krátký). Obsah je seskupený do pěti sekcí (Navigation, Views, Capture, Editor, Training). Na kombinaci kláves jeden řádek s přeložitelným textem v obou sloupcích. Levý sloupec (kód kláves) fixovaný na 180 pt šířku, aby popisy vpravo zůstaly vertikálně zarovnané. Žádná interakce kromě scrollování — klik na řádek nic nepouští, klávesové zkratky jsou skutečné modifikátory v menu a na náhledu.

#### JEDNODUŠE ŘEČENO

Tabulka všech zkratk. Statický cheat-sheet pro rychlé nahlédnutí.

### W6 VStack (zkratkové sekce)

 KDE

Uvnitř ScrollView..

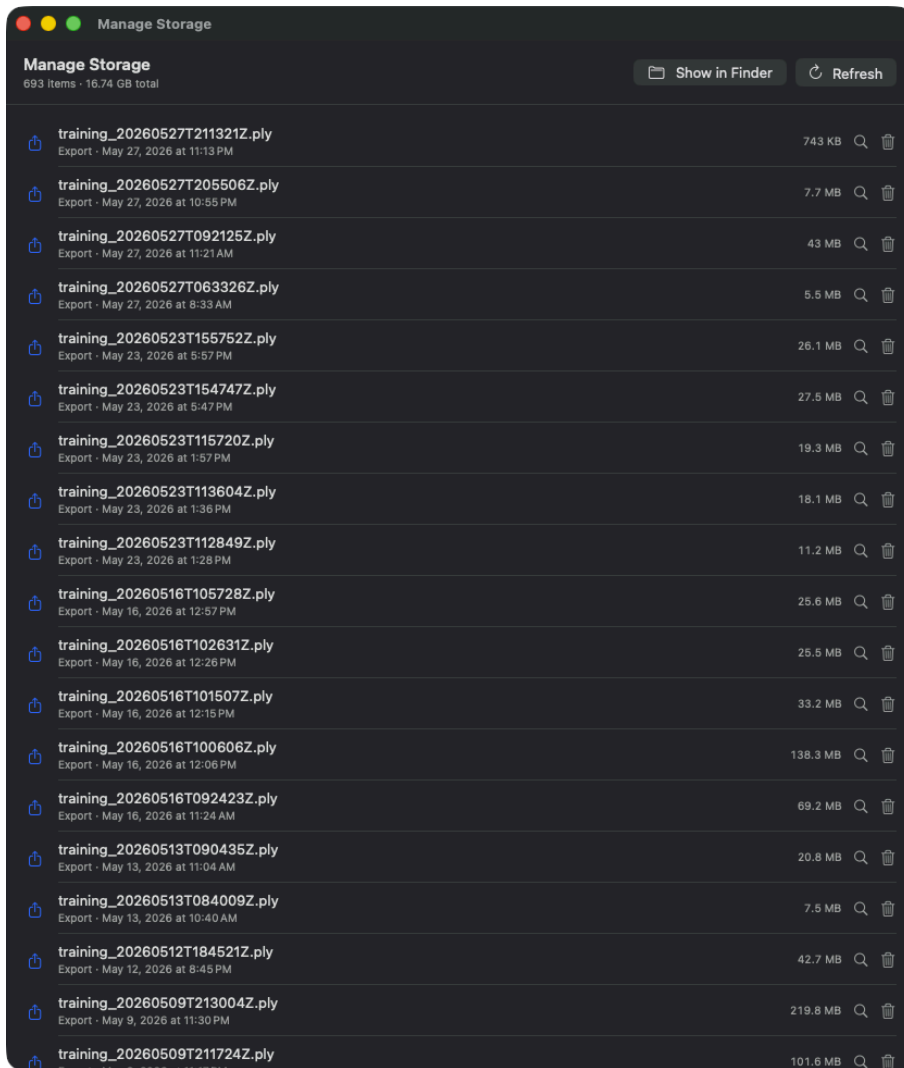
 TECHNICKY

Doleva zarovnané stohové sekce s 16 pt rozestupem. Uvnitř pěti sekcí vždy heading + sled řádků. Nadpisy používají sekundární subheadline styl — záměrně bez title formátu, protože sekce nemusí být navigovatelné. Obsah je záměrně plochý (žádné disclosure, žádný search, žádný filtr), aby komponenta na každé verzi macOS běžela nezměněně a soubor zůstal čitelný.

#### JEDNODUŠE ŘEČENO

Seskupení kláves podle funkce (Navigation, Views, Editor a tak dále).

## Manage Storage (W7–W12)



Obrázek 19: Okno Manage Storage — hlavička ukazuje „693 items · 16.74 GB total“, tabulka s export PLY soubory seřazená dle data, vždy format-pill + název souboru + velikost + datum

**CO JE NA OBRÁZKU** Tabulkový pohled všech RadianceKitem spravovaných souborů. Hlavička počítá 693 položek, celkovou velikost 16.74 GB. Toolbar nahoře: „Show in Finder“, + „Refresh“. Každý řádek: PLY ikona, název souboru (např. training\_20260527T211321Z.ply), datum exportu, velikost (variuje 7 KB až 218 MB), ikona lupy (reveal) a ikona koše (move to trash). Soubory jsou seřazené podle data, nejnovější nahoře. V této demo nahrávce dominují PLY exporty, protože se hodně pracovalo s `--benchmark`.

**Co to je:** Přehled využití disku pro vše, co RadianceKit ukládá pod `~/Documents/RadianceKit/` — logy, exporty, scény, capture bundly (z iOS doprovodu), importy (staging kopie vstupních obrázků). Na záznam velikost v bajtech a dvě tlačítka: „zobrazit ve Finderu“, a „přesunout do koše“. NENÍ to automatický úklid — aplikace sama nic nemaže; rozhoduješ se na záznam.

**KDY OTEVŘÍT** Když je disk plný. Především logy se sbírají (jeden JSONL na tréninkový pokus, plus `_qualityMetrics.json`); exporty samozřejmě také (PLY 100 % surová data,

jeden na export). Užitečné i po pádu, pokud staging adresář importů ještě má staré kopie vstupních obrázků (viz „Disk-pressure incident“, v `dev_v549f-needle-reduction.md`).

### W7 Tlačítko „Show in Finder,“



Hlavička vpravo nahoře v okně Storage Browser..



Otevírá celý RadianceKit adresář (`~/Documents/RadianceKit/`) ve Finderu, takže můžeš strukturu složek přímo vidět a také ji manipulovat ve Finderu samotném. Akce otevírá nové okno Finder a nepřepíná do App Sandbox containeru — `~/Documents/RadianceKit/` je doména Documents regulárně přístupná aplikacím, ne cesta sandboxed containeru.

#### JEDNODUŠE ŘEČENO

Otevírá adresář ve Finderu, abys sám mohl manipulovat se soubory.

### W8 Tlačítko „Refresh,“



Hlavička, vedle Finder tlačítka..



Spouští background scan, který běží na uživatelem iniciovaném asynchronním tasku, aby skenování velkých stromů adresářů neblokovalo UI. Samotné chození prochází každou známou podsložku (Logs, Exports, Scenes, Captures, Imports) a generuje storage záznam na přímé dítě. Na záznam se zjišťuje rekurzivní velikost — preferenčně skutečná spotřeba disku (včetně sdílení APFS hardlinks) s fallbackem na logickou velikost souboru.

#### JEDNODUŠE ŘEČENO

Načte seznam znovu, pokud jsi mezitím ve Finderu něco smazal nebo přidal.

**W9 List (storage záznamy)**

Hlavní obsah pod hlavičkou..

**TECHNICKY**

Seznam s tímto layoutem na řádek: kategoricky specifická SF symbol ikona (dokument pro logy, šipka nahoru pro exporty, krychle pro scény, tray pro importy), název + podtitul (kind label

1. formátované datum modifikace), counter bajtů vpravo

(zarovnáno vpravo, monospaced), reveal tlačítko (symbol lupy), trash tlačítko (koš). Třídění: primárně podle kindu (Scenes nejdřív, pak Exports, Logs, Captures, Imports, Other), sekundárně podle data modifikace sestupně (nejnovější nahoře). Pokud skenování ještě běží, ukazuje místo toho „Scanning...“, progress. Pokud se nic nenašlo, empty state s tray ikonou.

**JEDNODUŠE ŘEČENO**

Seznam všech tvých Radiance-Kit dat, seřazený podle typu a aktuálnosti. Na záznam vidíš velikost a můžeš přímo mazat.

**W10 Tlačítko řádku „Reveal in Finder„**

Na řádek, symbol lupy vpravo..

**TECHNICKY**

Otevírá Finder a označuje konkrétní item (soubor nebo složku). Rozdíl od W7: W7 otevírá root adresář; W10 značí přesně tento jeden záznam. Praktický workflow: identifikuj velký záznam, klikni na lupu, pak ho zkopíruj např. na externí volume.

**JEDNODUŠE ŘEČENO**

Skočí ve Finderu přímo na tento záznam, abys ho rychle našel.

**W11 Tlačítko řádku „Move to Trash„**

Na řádek, symbol koše vpravo vedle lupy..

**TECHNICKY**

Vyvolává potvrzovací dialog box (W12). Až po potvrzení běží macOS standardní operace „přesunout do koše„ (tedy reverzibilní, ne přímé mazání). Po úspěšném trashu se záznam odstraňuje ze seznamu a aktualizuje se counter celkových bajtů. Při chybách se zobrazuje modální chybový dialog.

**JEDNODUŠE ŘEČENO**

Přesouvá záznam do koše. Dialog se předtím zeptá.

**W12 ConfirmationDialog (potvrzení mazání)**

Vyvolán přes W11, zobrazen jako macOS sheet..

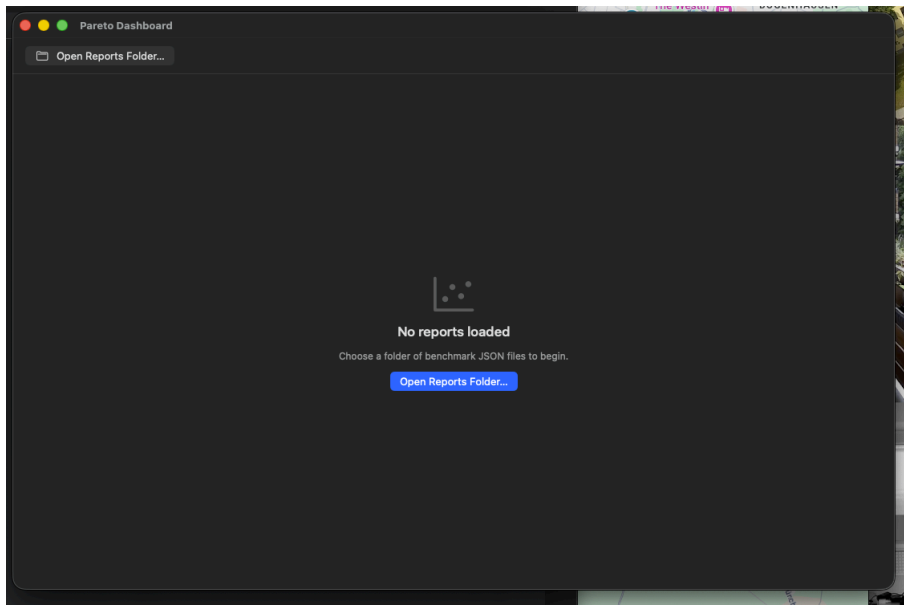
**TECHNICKY**

Standardní potvrzovací dialog s dynamickým titulem „Delete <name>?„ a řádkem zprávy, který explicitně upozorňuje, že záznam přistane v koši a je z něj obnovitelný (dokud se koš nevyprázdní). Dvě tlačítka: „Move to Trash“ jako destruktivní akce (zobrazená červeně) a „Cancel„ s automatickou Esc vazbou. Dialog je non-modal v tom smyslu, že blokuje pouze toto okno, ne celou aplikaci — to je macOS standard pro reverzibilní mazání.

**JEDNODUŠE ŘEČENO**

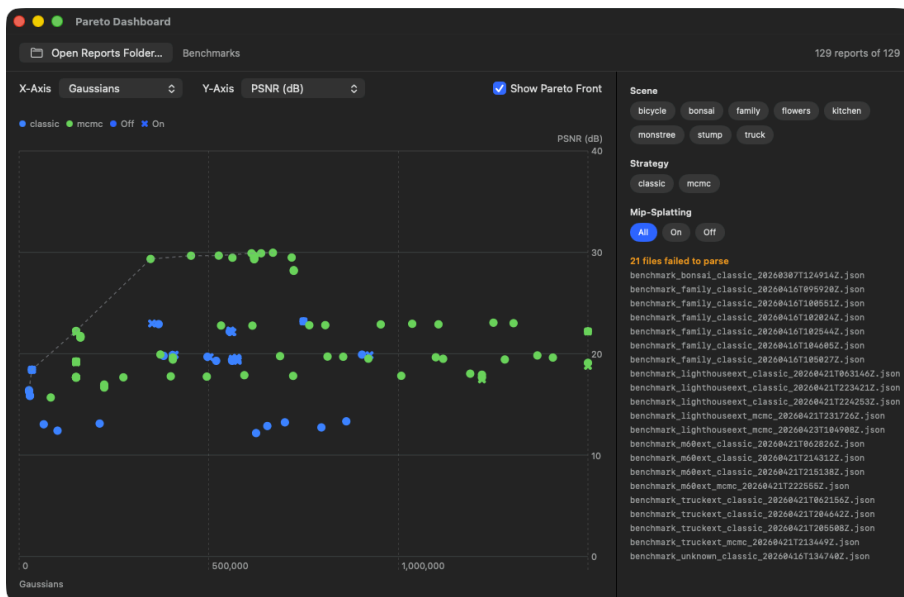
Bezpečnostní otázka před mazáním. „Move to Trash„ je reverzibilní — dokud koš není vyprázdněn.

## Pareto Dashboard (W13–W22)



Obrázek 20: Pareto Dashboard — prázdný stav před importem reportu

Prázdný stav (po prvním otevření) — empty state s call-to-action „Open Reports Folder...“. Datové body se objevují, jakmile jsou tréninkové reporty načteny, viz další snímek.



Obrázek 21: Pareto Dashboard se 129 načtenými benchmark reporty — Gaussians vs PSNR s Pareto frontou, filtry Scene/Strategy/Mip

**CO JE NA OBRÁZKU** Toolbar v hlavičce ukazuje „129 reports of 129“ (všechny reporty ve zvolené složce úspěšně sparsovány — 21 dalších souborů nebylo možné sparsovat kvůli staršímu formátu, viz seznam poznámek vpravo). Osy: X-Axis picker na Gaussians, Y-Axis picker na PSNR (dB). Scatter plot: zelené body = Classic-Strategy, modré body = MCMC. Přerušovaná Pareto front čára běží podél nejlepších dosažených PSNR hodnot


a plateauje kolem  $PSNR \approx 30$  dB od asi 500K Gaussianů. Filter chipy vpravo: 7 scén, 2 strategie (classic, mcmc), 3 možnosti Mip-Splatting (All, On, Off). Aktuálně jsou všechny filtry otevřené, proto hustý cluster bodů.

**Co to je:** Nástroj pro porovnání více běhů. V minulosti jsi trénoval více scén nebo stejnou scénu s různými předvolbami — každý z těchto tréninkových běhů produkuje (pokud jsi přidal `--benchmark` nebo zavolal přes `benchmark` funkci) JSON report soubor, který mimo jiné obsahuje finální PSNR, SSIM, LPIPS, Gaussian count a wallclock čas. Dashboard čte celou složku takových reportů najednou a plotuje je jako 2D scatter se selectovatelnými osami. Navíc je vykreslena Pareto fronta (množina nedominovaných bodů) jako přerušovaná čára.

**KDY OTEVŘÍT** Poté co jsi vytvořil alespoň tři nebo čtyři tréninkové reporty. S méně body fronta čára není výpovědní. Typický use case: zkoušel jsi rekonstruovat outdoorovou scénu a postupně jsi projel P3 Balanced (Classic), P4 Quality (Classic), P7 MCMC Quality a P9 Outdoor (tuned) — nyní chceš vědět, která konfigurace dodává nejlepší PSNR na sekundu tréninkového času nebo která potřebuje nejméně Gaussianů pro danou PSNR.

**JAK INTERPRETOVAT** Obě osy jsou volně volitelné (X osa: `psnr`, `ssim`, `lpips`, ...; Y osa stejně). Logika Pareto fronty v `ParetoFront2D.indices` ví pro každou metriku, zda „menší = lepší“, (např. LPIPS, Loss, Time) nebo „větší = lepší“ (PSNR, SSIM) — čára tedy podle volby os běží zleva dolů doprava nahoru nebo zleva nahoru doprava dolů, vždy podél nejlepší dosažené kombinace. Bod je Pareto-optimální, pokud ŽÁDNÝ jiný bod není v OBOU dimenzích alespoň stejně dobrý (tedy žádný jiný ho nedominuje). Pareto-optimální body leží na čáře, jiné body vpravo/výše (podle orientace os) od ní. Body NA čáře jsou skuteční kandidáti pro „nejlepší předvolba“; body DALEKO od čáry jsou promarněný tréninkový čas.

**FILTER CHIPY** Můžeš výběr omezit na konkrétní scénu (pokud chceš např. porovnat pouze outdoorové runy), na konkrétní strategii (Classic nebo MCMC), nebo na Mip-Splatting on/off (relevantní po fázi Q1.5, kde Mip zůstává jako opt-in advanced flag).

 Máš tři reporty pro „truck“ scénu 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). Nastav X osu na `trainingTime`, Y osu na `PSNR`. Run B leží vpravo nahoře, Run C ještě dál vpravo nahoře, Run A vlevo dole. Pareto fronta spojuje A a C — oba nedominované. Run B je „lost“ (C je lepší v Time I PSNR). Poznatek: pro „truck“ se MCMC default nevyplácí; buď rychle+OK (A) nebo dlouho+velmi dobře (C). Konfiguraci z C uložit jako vlastní předvolbu (Inspektor → I1 Save Preset).

**Další akce:** uložit nejlepší konfiguraci jako předvolbu. Konkrétně: podívej se na Pareto body (hover ukazuje PSNR/SSIM/LPIPS/Gs/Time v tooltipu), rozhodni se, který ti z `time-vs-quality` kompromisu sedí nejlépe, otevři příslušný report (název souboru obsahuje `run timestamp`), zkopíruj jeho tréninkovou konfiguraci do nového běhu nebo ji ulož po další tréninkové relaci jako předvolbu přes Inspektor.

**W13 Tlačítko „Open Reports Folder...“**

Toolbar nahoře vlevo..

**TECHNICKY**

Otevírá výběr složky s výzvou „Select a folder containing benchmark .json reports,“. Po potvrzení běží background task, který sekvenčně parsuje všechny .json soubory ve složce. Chybné reporty (poškozený JSON, špatné schema) se sbírají a dole v sidebaru se zobrazují jako „N file failed to parse“ — žádný pád. Pokud nastane druhý klik, zatímco první load ještě běží, předchozí task se přeruší, takže do stavu nezapisují dva výsledky současně.

Také přes CLI: `--dashboard-dir /cesta/k/reports` načte složku hned při startu aplikace.

**JEDNODUŠE ŘEČENO**

Vybírá složku, ve které leží tvé benchmark reporty. Výchozí cesta je `~/Documents/RadianceKit/Reports/`. Pak načte všechny JSONy najednou.

**W14 Picker „X-Axis“**

Nad chartem, vlevo..

**TECHNICKY**

Menu picker se všemi dostupnými metric osami modulu dashboardu (PSNR, SSIM, LPIPS, Gaussian count, tréninkový čas atd.). Default je Gaussian count. Při změně se zrušuje pohnutý hover bod, protože dosud zvýrazněná pozice ve starém souřadnicovém systému os po změně os už nemá smysl. Picker je omezen šířkou obsahu, aby netáhl přes celou šířku.

**JEDNODUŠE ŘEČENO**

Která metrika má stát na vodorovné ose. Obvykle „tréninkový čas,“ nebo „počet Gaussianů“, protože to jsou „náklady,“ které chceš porovnávat.

**W15 Picker „Y-Axis„**

KDE

Nad chartem, vedle X-Axis..



TECHNICKY

Identické s W14, jen že default je PSNR. Volba os se ukládá nezávisle, takže uživatel může volit i nesmyslné kombinace (X=PSNR, Y=PSNR — hodilo by všechny body na diagonálu). Takové kombinace ale nejsou odchyťávány; vědomé rozhodnutí, protože porovnání „SSIM vs PSNR„ je rozhodně zajímavé, aby se vidělo, jak konzistentně se metriky chovají.

**JEDNODUŠE ŘEČENO**

Co stojí na svislé ose. Normálně „PSNR„ nebo „SSIM“ jako míra kvality.

**W16 Toggle „Show Pareto Front„**

KDE

Vpravo vedle pickerů os..



TECHNICKY

Standardní macOS toggle. Pokud je aktivní, v Pareto chartu se navíc k mraku bodů kreslí čára s vypočítanou 2D Pareto frontou. Styl: přerušovaný (dash pattern 4–4), šedě poloprůhledný, síla čáry 1.5 pt. Výpočet Pareto fronty běží na hlavním vlákne — u typického počtu reportů ( $\leq \sim 50$ ) je to bez problémů rychlé. Pokud je toggle vypnutý, čára se vynechává, takže stojí jen holé body.

**JEDNODUŠE ŘEČENO**

Ukazuje čáru, která vede přes „dosud nejlepší„ body. Pokud ti čára překáží (např. protože chceš porovnat jen jednotlivé trade-offy), vypni ji.

**W17 Chipy „Scene„ filtru**

KDE

Pravý sidebar v okně Dashboard..



TECHNICKY

Filter chipy pro každou scénu vyskytující se v načtených reportech. Vlastní flow layout, který chipy automaticky balí do více řádků, jakmile je šířka vyčerpaná. Aktivní chipy dostávají akcentové pozadí, neaktivní neutrální standardní material pozadí. Vícenásobný výběr je možný (Set sémantika); pokud není zvolen žádný chip, platí všechny scény jako „propuštěné„ — tj. logika setu je „prázdný výběr = vše“, ne „prázdný výběr = nic„.

**JEDNODUŠE ŘEČENO**

Klik na název scény filtruje body pouze na tuto scénu. Vícenásobný výběr možný. Prázdné = všechny scény.

**W18** Chipy „Strategy„ filtru

KDE

Pod Scene filtrem v sidebaru..



TECHNICKY

Přesně jako W17, ale pro tréninkové strategie — typicky dvě hodnoty „classic“, a „mcmc“, odvozené z strategy pole benchmark report JSON. Užitečné, pokud máš reporty obou strategií smíchané a chceš vidět jen jeden druh (např. „ukázat jen MCMC runy, protože Classic jsem už vyloučil“).

**JEDNODUŠE ŘEČENO**

Filtr podle Classic nebo MCMC. Standardně jsou obě aktivní.

**W19** Chipy „Mip-Splatting„ filtru

KDE

Pod Strategy filtrem v sidebaru..



TECHNICKY

Trojhodnotový filtr (místo setu jako W17/W18): „All, / „On“ / „Off,“. Pozadí: Mip-Splatting byl ve fázi Q1.5 evaluován jako experimentální vylepšení multi-scale a finální verdikt byl „žádné spolehlivé vítězství; ponechat jako opt-in flag“. Pokud děláš mip-on/off porovnání, často chceš ostře oddělit. Proto dedikovaný ternární filtr se stavy „vše propustit,“, „pouze Mip on“, „pouze Mip off,“. Sekce sidebaru se zobrazuje pouze tehdy, pokud je v datech alespoň jeden Mip report A alespoň jeden ne-Mip report (jinak filtrování nedává smysl).

**JEDNODUŠE ŘEČENO**

Pokud chceš porovnat Mip-Splatting on/off, tady trojdílný filtr. Jinak ignoruj.

**W20** ChipButton (filter toggle, all/on/off)

KDE

Helper komponenta, používá se v W17/W18/W19..



TECHNICKY

Minimalistický button wrapper. Obsah: label text s caption fontem a paddingem 10 horizontálně / 5 vertikálně. Pozadí podmíněčné: pokud aktivní → akcentová barva aplikace s bílým textem; jinak neutrální standardní material pozadí s černým textem. Tvar je capsule (pilulkovitý). Plain button style, aby capsule material nebyl překryt systémovým borderem.

**JEDNODUŠE ŘEČENO**

Samotná kulatá filter tlačítka. Vypadají jako iOS tag.

**W21 Chart (Pareto scatter)**

Středová plocha dashboardu..



Swift Charts diagram se dvěma vrstvami: 1. jeden bod na report — pozice z vybraných X a Y metrik, barva podle Strategy, symbol podle Mip statusu. Velikost symbolu normálně 80, zvýrazněno 200 (pokud ID odpovídá aktuálně hovered reportu). 2. čára pro Pareto frontu, pouze pokud je toggle zapnut.

Chart overlay: průhledný obdélník registruje pohyb myši; na frame se vypočítává euklidovskými nejbližší pozice bodu v plot frame a aktualizuje se hover report, pokud je vzdálenost pod 24 px (jinak reset). Tak dostaneš tooltip bez klikání — stačí hover.

**JEDNODUŠE ŘEČENO**

Samotný scatter diagram. Každý bod je jeden tréninkový běh. Hover pro detail tooltipu.

**W22 Tooltip (hover detail)**

Pod chartem, zobrazený při hoveru..

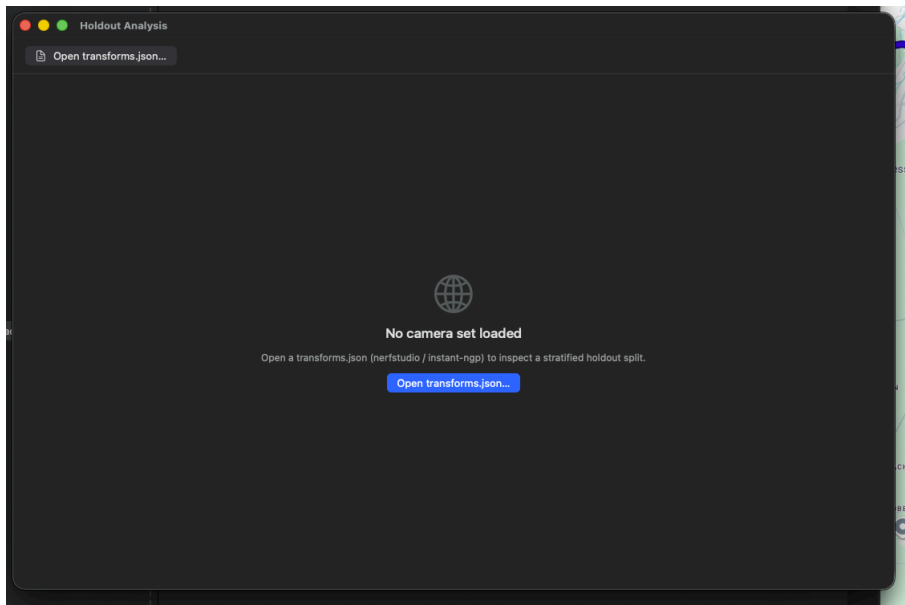


Horizontální stack: název scény (headline), strategy tag (caption), oddělovací čára, pak PSNR/SSIM/LPIPS/Gs/Time metriky vždy v malé vertikální skupině (label + monospaced hodnota). Pokud byl Mip aktivován, navíc „Mip“, capsule tag v akcentové barvě. Pozadí poloprůhledný blur, zaoblený obdélník s 8 pt radiusem. Zobrazuje se pouze tehdy, pokud je myš skutečně nad bodem. Automaticky mizí při opuštění.

**JEDNODUŠE ŘEČENO**

Detailní karta dole, když najedeš myši na bod. Ukazuje všechny kvalitativní metriky a konfiguraci běhu najednou.

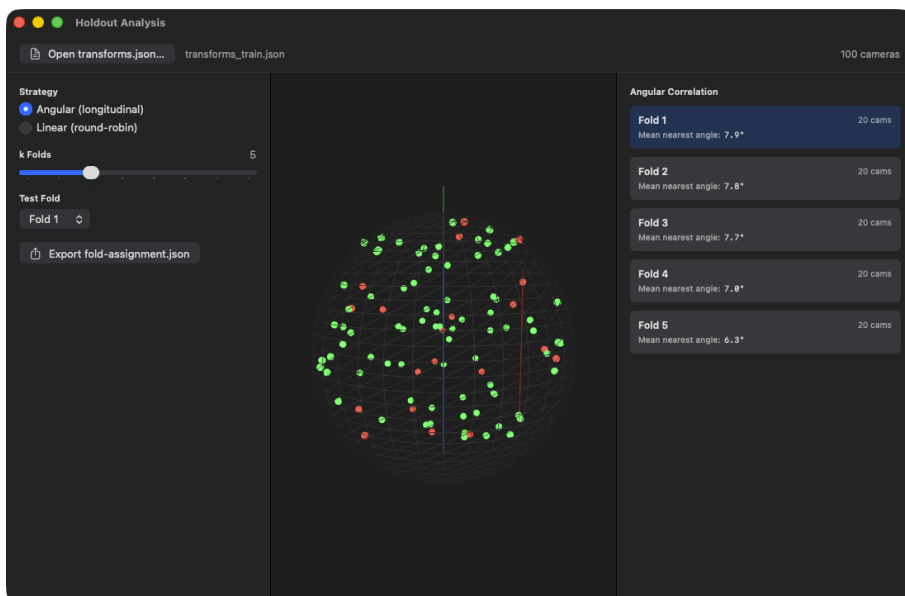
## Holdout Analysis (W23–W29)



Obrázek 22: Holdout Analysis — prázdný stav před načtením transforms.json

Prázdný stav s empty state a call-to-action „Open transforms.json...“. Akceptuje formát Nerfstudio a Instant-NGP.

Prázdný stav (po prvním otevření) — kamerové markery se objevují, jakmile je načten transforms.json, viz další snímek.



Obrázek 23: Holdout globe se 100 NeRF-Blender Mic kamerami, 5 foldů á 20 kamer, aktivní Angular strategie

**CO JE NA OBRÁZKU** Hlavička ukazuje načtený soubor (transforms\_train.json) a count kamer („100 cameras“). Levý sidebar: strategy picker se dvěma možnostmi — aktivní Angular (longitudinal) (řadí foldy podle longitudinálních/latitudinálních sektorů na sféře, aby každý test fold byl geometricky hustý) vs Linear (round-robin) (založeno na


pořadí, všechny k-té framy jako test set). Slider k-foldů stojí na 5, picker test foldu na Fold 1. Export tlačítko generuje `fold-assignment.json` pro Nerfstudio/Instant-NGP. Středový panel: 3D globus projekce všech 100 kamer — zelené body = train, červené body = aktuální test fold (Fold 1 s 20 kamerami). Pravý sidebar (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°) — menší hodnota znamená, že kamery uvnitř tohoto foldu leží těsně u sebe, tedy holdout split je prostorově koherentní.

**Co to je:** 3D vizualizér tvého kameraového uspořádání s logikou cross-validation. Načítáš `transforms.json` (standardní formát Nerfstudio / Instant-NGP pro pozice kamer), aplikace čte všechny kamery, projektuje jejich směry pohledu na jednotkovou kouli a ukazuje je jako malé sférické markery na virtuálním glóbu. Pak dělí kamery do  $k$  foldů (podle zvolené strategie: angular nebo linear), zeleně značí tréninkový podíl a červeně test podíl (holdout) a počítá na fold angular correlation score, který říká, jak daleko leží test fold od tréninkového foldu v prostoru úhlů pohledu.

**KDY OTEVŘÍT** Když chceš dělat holdout evaluaci — tedy: jak dobře generalizuje tvůj model na neviděné úhly pohledu? Standard v tréninku je „every-8th view jako holdout,“ (Mip-NeRF360 konvence), ale to je velmi lineární rozdělení. Pokud jsou tvé obrázky např. časově clusterované (nejprve jedna strana objektu, pak druhá), pak „every-8th“ není reprezentativní — náhodná sekvenční pozice přistane v testu, ale všichni její sousedé jsou v tréninku, to je moc jednoduché. S „angular,“ se místo toho stratifikuje přes prostor úhlů pohledu: každý fold obsahuje kamery ze všech oblastí orbitu, takže test skutečně testuje mezery v generalizaci.

**JAK INTERPRETOVAT** Angular vs Linear: - Angular (standard): dělí kamery podle longitudinálního úhlu ( $\phi$  souřadnice kolem osy Y) do  $k$  stejných sektorů. Fold 0 jsou kamery s  $\phi \in [0^\circ, 360/k^\circ)$ , Fold 1 další, a tak dále. Výhoda: každý fold pokrývá výřez orbitu; test fold je prostorově kompaktní, ale široce rozprostřený přes světový dataset. Dobré pro klasické orbitální záběry. - Linear (round-robin): fold index = (image\_index modulo k). To je jednoduchá „every-k-th,“ distribuce. Funguje, pokud pořadí obrázků NEMÁ prostorový bias (např. náhodně seřazené dronové záběry). Funguje špatně, pokud obrázky časově clusterují.

V 3D glóbu okamžitě vidíš: zelené body (training) a červené body (test). Pokud červené body všechny clusterují v jednom rohu, je holdout špatný (žádný dobrý test generalizace). Pokud leží rovnoměrně mezi zelenými, je dobrý. Angular correlation score na fold (pravý sidebar, ve stupních) navíc říká: menší hodnota = test je blízko tréninku (každá test kamera má blízkou tréninkovou kameru, snadný test); větší hodnota = test je daleko od tréninku (tvrdší generalizace).

 Pořídil jsi svou Truck scénu s 251 obrázky, exportuješ přes menu M33 (Export SfM transforms.json) nerfstudio soubor. Otevři Holdout okno (⇧⌘H), načti JSON přes „Open transforms.json...“, podívej se na globus.  $k=5$  (default) ti dává 5 foldů. Klik na „Fold 3“ — vidíš, zda jsou červené markery zhruba rovnoměrné. Pokud ano: „Export fold-assignment.json“, polož exportovaný soubor do reports složky a při dalším training runu s `--benchmark` (nebo odpovídajícími nastaveními Inspektoru) se použije přesně toto rozdělení foldů jako test holdout — místo standardu „every-8th“.

**W23** Tlačítko „Open transforms.json...“

Toolbar nahoře vlevo..



Otevírá souborový dialog omezený na JSON soubory. Po potvrzení Holdout modul načítá soubor. Loader parsuje jak formát nerfstudio (camera intrinsics plus seznam framů s cestou obrazu a transformaticí), tak formát instant-ngp (stejná stavba). Na frame se z transform matice extrahuje směr pohledu (z osa kamerové lokální báze) a ukládá. Pokud se parsování nezdaří, zobrazuje se chybové hlášení ve stavové oblasti.

Také přes CLI: `--holdout-file /cesta/k/transforms.json` startuje okno přímo s načteným souborem.

**JEDNODUŠE ŘEČENO**

Načte tvůj JSON s pozicemi kamer. Standard jsou exporty Nerfstudio a Instant-NGP. RadianceKit sám může transforms.json exportovat přes menu → Export → SfM.

**W24** Picker „Strategy„ (angular/linear)

Levý sidebar, nahoře..



Radio picker se dvěma možnostmi: Angular a Linear. Změna strategie automaticky spouští přepočítání. Směry pohledu jsou seznam 3D jednotkových vektorů na sféře; angular strategie je projektuje na longitudinální úhel  $\phi$  a třídí, linear strategie dělá prostě modulo distribuci přes frame index.

**JEDNODUŠE ŘEČENO**

Angular pro rovnoměrné orbitální záběry (standard, bezpečné), Linear jen pokud tvé obrázky neclusterují prostorově.

**W25 Slider „k Folds„**

KDE

Levý sidebar, uprostřed..



TECHNICKY

Slider od 3 do 10, krok 1. Při změně se výpočet foldů automaticky spouští znovu, takže seznam foldů, train/test indexy a per-fold score se okamžitě přepočítávají. Zvolená hodnota se zobrazuje jako monospaced digit text vpravo vedle labelu.

Pravidlo:  $k=5$  je standard (dává ti 20 % test na fold, to je obvyklé pro cross-validation).  $k=10$  pokud máš hodně dat a potřebuješ víc foldů pro statistickou výpovědní hodnotu.  $k=3$  pokud máš málo dat.

**JEDNODUŠE ŘEČENO**

Kolik foldů v rozdělení. 5 je standard a sedí téměř vždy.

**W26 Picker „Test Fold„**

KDE

Levý sidebar, pod k sliderem..



TECHNICKY

Menu picker. Možnosti jsou dynamicky  $0..<k$ , label „Fold 1“ až „Fold N“, (tedy 1-indexed v UI, 0-indexed interně). Pokud je předchozí zvolený index  $\geq k$  (např. protože jsi snížil  $k$  z 10 na 5), automaticky se resetuje na 0. Zvolený test fold se v globu zobrazuje červeně, všechny ostatní zeleně.

**JEDNODUŠE ŘEČENO**

Který fold je právě test fold. Můžeš proklikávat a vidíš, jak každý jednotlivý fold v globu vypadá.

**W27 Tlačítko „Export fold-assignment.json„**

KDE

Levý sidebar, dole..



TECHNICKY

Otevírá Save dialog s defaultním názvem `fold-assignment.json`. Po potvrzení Holdout modul kódu je aktuální distribuci do JSON schématu (per frame fold přiřazení plus strategy meta blok). Tento soubor lze pak při dalším tréninku s `--benchmark` přibalit, takže se pro finální vyhodnocení metrik používá stejný holdout. Chyby zápisu se zobrazují jako chybový text; úspěch v zeleném textu jako „Saved to (filename)“.

**JEDNODUŠE ŘEČENO**

Uloží aktuální train/test rozdělení jako JSON. Tento soubor pak můžeš při tréninku přímo přibalit, aby se použila stejná testovací sada.

**W28 SCNView (3D Camera Globe)**

KDE

Středový panel v okně Holdout..

**TECHNICKY**

SceneKit globus view. Scéna sestává z: wireframe koule (poloměr 1.0, 36 segmentů, tmavě šedá), tří barevných osových pahýlů (červený/zelený/modrý pro X/Y/Z, každý 1.2 dlouhý), a na kameru malá marker koule (poloměr 0.03) na odpovídající pozici pohledu na jednotkové kouli (lehce vně, aby nezmižela v wireframe kouli). Markery se při změně foldu NESTAVÍ znovu — rebuild je potřeba jen tehdy, pokud se mění seznam framů (tedy nová JSON je načtena). Místo toho běží na update in-place aktualizace materiálových barev: červená pro test indexy, zelená pro training, světle šedá pokud ani jedno. Tak zůstávají slider ticky výkonné i u  $N > 1000$  kamer.

Ovládání kamery je aktivováno — můžeš myší globus otáčet, zoomovat, posouvat. Osvětlení se stará, aby markery nevypadaly ploché. Pozadí je tmavě šedé.

**JEDNODUŠE ŘEČENO**

3D globus s kamerovými pozicemi. Zelená = training, červená = test, světle šedá = nepřiráženo (nevyskytuje se, všechny kamery někam patří). Myší můžeš globus rotovat a zoomovat.

**W29 FoldCard (klikni pro výběr foldu)**

KDE

Pravý sidebar, sekce „Angular Correlation,..“

**TECHNICKY**

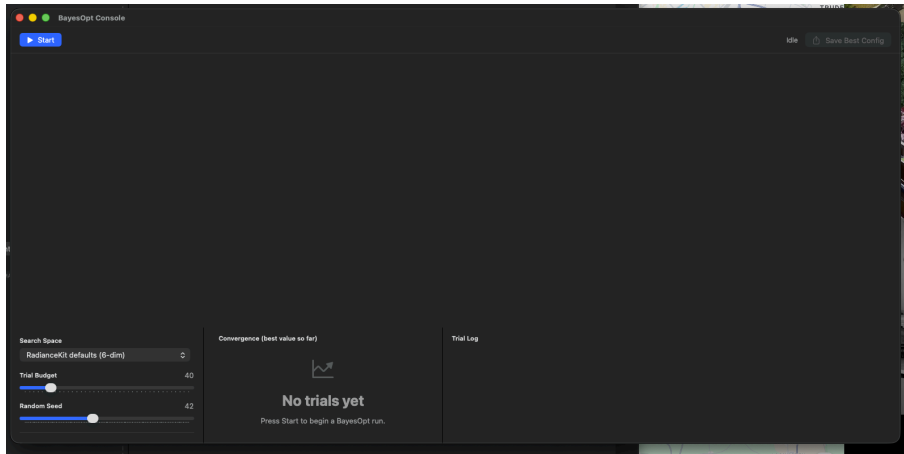
Na fold karta view — zaoblený obdélník s 6 pt radiusem, padding 10, vertikální layout se dvěma řádky (nahore „Fold N„ + počet kamer, dole „Mean nearest angle:“ + hodnota ve stupních). Barva pozadí podmíněčně: aktivní fold = akcentová barva polo-průhledně, neaktivní = neutrální standardní material. Klepnutí volí fold, a globus se živě překresluje barvami.

Score „Mean nearest angle„ je střední nejmenší úhel na test kameru k nejbližší tréninkové kameře (interně počítáno v radiánech, v UI zobrazeno ve stupních).

**JEDNODUŠE ŘEČENO**

Na fold malá karta vpravo s počtem kamer a průměrnou vzdáleností k nejbližší tréninkové kameře. Klik na ni volí tento fold jako test.

## BayesOpt Console (W30–W39)



Obrázek 24: BayesOpt konzole — prázdný stav před startem trialu

Prázdný stav se search space pickerem (RadianceKit defaults (6-dim)), slider Trial Budget (default 40), Random Seed (42) a tři prázdné panely pro convergence chart, trial log a seznam parametrů search space.

Prázdný stav (po prvním otevření) — convergence chart a trial tabulka se naplňují, jakmile byl spuštěn běh, viz další snímek.



Obrázek 25: BayesOpt konzole po 40 trialech — convergence chart strmě stoupá až k trialu 15, Best Value 0.9943, Trial log s init/bo/restart tagy


**CO JE NA OBRÁZKU** Status vpravo nahoře „Finished — best 0.9943 after 40 trials“. Levý sidebar: search space picker na RadianceKit defaults (6-dim), Trial Budget 40, Random Seed 42. Seznam parametrů ukazuje šest ladících hyperparametrů s jejich hodnotovými rozsahy: 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]. Střed: convergence chart (X = trial index 1–40, Y = objective value 0–1) — šedé body = initial samples (LHS), modré body = BayesOpt acquisition, oranžové body = restart trials (#22 a #31). Best value čára strmě stoupá k trialu ~7, pak už jen marginální zlepšení do trialu 15, od tam plochý plateau na 0.99+. Pravý sidebar: trial log #1–#34 se skórem + tagem (init/bo/restart). Tlačítko Save Best Config vpravo nahoře zapisuje bayesopt-best.json .

**Co to je:** Bayesovská optimalizační konzole pro hledání hyperparametrů. Bayes-Opt je automatická metoda, která se pokouší s co nejméně experimenty najít optimální bod neznámé funkce — typicky: „která kombinace z mcmcMaxGaussians, capMultiplier, ssi-mWeight a gradThreshold dodává nejlepší PSNR pro mou třídu scén?“. Místo gridu  $6^4 = 1296$  trialů Bayes-Opt zkusí asi 40–100 informovaných trialů a tím se blíží optimu.

**Důležité:** aktuální verze dodaná v aplikaci nevede optimalizaci proti skutečným tréninkovým runům (to by trvalo dny), ale proti syntetické demo objective — multimodální krajině s hill-climbing charakterem plus mírným šumem. To je záměrně tak: okno ti má ukázat chování optimizéru (konvergenční průběh, sample body, best-so-far) a nechat tě porozumět definicím search space. Pro skutečné tréninkem řízené BayesOpt runy (jak prováděno ve fázi Q7 pro Scene-Class předvolby) se používá samostatný offline CLI workflow; toto okno je live UI varianta.

**KDY OTEVŘÍT** Tři použití: 1. Chceš porozumět, jak BayesOpt pracuje — pak spusť demo run a sleduj convergence chart. 2. Plánuješ novou třídu scén (např. „akvária“ nebo „antický nábytek“), pro kterou vestavěné 10 předvoleb nesedí dokonale. Mentálně definuj search space, otestuj ho zde s „Bowl demo“ nebo „Densify“ předvolbou, exportuj pak best config jako JSON a použij ji jako výchozí bod pro skutečný tréninkový run. 3. Chceš inspektovat default search spaces definované v RKBayesOpt balíku (Mip-Subset, RadianceKit Defaults) — jsou listovány v parameter panelu levého sidebaru.

**JAK INTERPRETOVAT** - **Convergence chart** (středový sloupec): Y = dosud nejlépe dosažená hodnota objective funkce. X = trial index. Zpočátku strmě stoupající (BayesOpt zkouší initial samples náhodně, některé z nich mají štěstí), pak stále plošěji, protože oblast blízkého optima je vyčerpaná. Pokud čára zůstane 20+ trialů plochá, můžeš run zastavit — další trialů už nic nepřinesou. Jednotlivé body v chartu jsou individuální trial hodnoty (tedy ne „best so far“), obarvené podle fáze: šedá = initial sample, modrá = bayesopt acquisition, oranžová = restart. - **Trial tabulka** (pravý sloupec): #1, #2, #3, ... vždy s hodnotou a phase tagem. Dosud nejlepší trial je označen žlutou hvězdou. Z tabulky můžeš identifikovat best trial a jeho parametry si při exportu prohlédnout. - **Search space inspector** (levý sidebar): ukazuje pro zvolenou předvolbu všechny názvy parametrů a jejich rozsahy hledání  $[lo, hi]$ . Pokud stojíš na předvolbě „RadianceKit defaults (6-dim)“, vidíš např. „densifyGradThreshold [5e-7, 5e-6]“ — tedy log-uniform mezi těmito hodnotami.

 Zvol předvolbu „RadianceKit defaults (6-dim)“, Trial Budget 40, Seed 42. Klikni „Start“. Pozoruj: prvních 8 trialů jsou šedé (initial samples, LHS Latin Hypercube), následné modré (BayesOpt acquired). Convergence chart bude strmý do trialu ~15, poté se zplošťuje. Při trialu ~30–40 se stabilizuje nejlepší hodnota. Klikni „Save Best Config“, — uloží se `bayesopt-best.json` s názvem předvolby, trial indexem, hodnotou a dekódovanými hodnotami parametrů. Tento JSON pak můžeš ručně převzít do své definice předvolby.

**W30 Tlačítko „Start„**

KDE

Toolbar vlevo, ve stavu Idle/Finished..



Resetuje seznam trialů, přepíná do Running stavu, generuje nové Run ID (pro stale detection při vícenásobných start klicích) a vytváří čerstvou pause gate. Pak startuje background task, který spouští optimalizér jako asynchronní stream. Velikost initial samples vyplývá z  $\min(8, \text{budget} / 4 + 1)$  — tedy typicky 8 Latin Hypercube samples při budgetu  $\geq 28$ , méně u malého budgetu. Trial updaty se přijímají inkrementálně a připojují do seznamu. Stale run protection: pokud mezitím druhý start klik nastaví Run ID znovu, updaty ze starého runu se zahazují.

Primary action style pro prominentní vzhled tlačítka.

**JEDNODUŠE ŘEČENO**

Startuje čerstvý optimalizační běh s aktuálním search space, budgetem a seedem.

**W31 Tlačítko „Pause„**

KDE

Toolbar vlevo, ve stavu Running..



Nastavuje pause gate aktivně a přepíná do Paused stavu. Skutečný efekt: runner čeká v 50 ms polling loopu, než vyhodnotí další objective funkci. To znamená, že právě běžící trial se dokončí (je přece syntetický a trvá jen mikrosekundy), ale žádný další trial se nespouští. Jakmile běží Resume, jde to dál tam, kde to skončilo.

**JEDNODUŠE ŘEČENO**

Zastaví běh. Aktuální výpočet ještě doběhne, pak se pauzuje.

**W32 Tlačítko „Stop„**

KDE

Toolbar vlevo, ve stavu Running a Paused..



TECHNICKY

Ruší runner task, nuluje referenci, řeší pause gate (pokud bylo paused), a přepíná do Finished stavu (pokud trialy existují) nebo Idle stavu (pokud žádné). Již vypočítané trialy zůstávají v seznamu viditelné — Stop je nemaže. Destruktivní role tlačítka zobrazuje tlačítko červeně, protože ruší run.

**JEDNODUŠE ŘEČENO**

Definitivně ruší běh. Trialy zůstávají viditelné, můžeš best config přesto exportovat.

**W33 Tlačítko „Resume„**

KDE

Toolbar vlevo, ve stavu Paused..



TECHNICKY

Řeší pause gate a přepíná zpět do Running stavu. Runner task už běží (čeká v polling loopu); jakmile loop pozná, že pauza je zrušena, běží dál a spouští další trial.

**JEDNODUŠE ŘEČENO**

Obnovuje pauzovaný běh.

**W34 Tlačítko „Save Best Config„**

KDE

Toolbar vpravo, vždy viditelné (ale disabled, pokud neexistuje bestTrial)..



TECHNICKY

Otevírá Save dialog s defaultním názvem `bayesopt-best.json`, omezeno na JSON. Po potvrzení se staví payload dictionary: preset name, trial index, hodnota (objective score), parametry (dictionary dekódovaných parameter names → hodnoty). Decodování projektuje normalizované souřadnice search space v  $[0,1]^d$  zpět do původního rozsahu hodnot (s log-uniform/ linear/integer škálami příslušně). JSON výstup je pretty-printed a se seřazenými klíči. Při chybě zápisu (v aktuální demo verzi) se tiše ignoruje — žádné error UI, protože je to demo cesta. Tlačítko zůstává šedé, dokud žádný trial neproběhl.

**JEDNODUŠE ŘEČENO**

Ukládá hodnoty parametrů do- sud nejlepšího trialu jako JSON. Tyto hodnoty pak můžeš ručně převzít do své konfigurace před- volby.

**W35 Picker „Search Space,, předvolba**

KDE

Levý sidebar, nahoře..

## TECHNICKY

Menu picker se čtyřmi preset možnostmi: - „RadianceKit defaults (6-dim)„ — kompletní standardní search space se všemi Q7 hyperparametry. - „Mip subset (2-dim)“ — jen `mipSmoothing3DScale` [0.05, 0.5] log-uniform a `mipFilter2DVariance` [0.1, 0.6] linear. Užitečné, pokud chceš ladit Mip-Splatting pro třídu scén. - „densify-until + ssim-weight 1. grad-thresh„ — tři densify relevantní parametry (`densifyGradThreshold` log-uniform, `ssimWeight` linear, `densifyUntilIter` integer). - „Bowl demo (1-dim)„ — pedagogický single parameter search space pro „takhle funguje BayesOpt“ dema.

Zatímco běh je aktivní, search space nelze měnit (zmátlo by to optimizér).

## JEDNODUŠE ŘEČENO

Který hyperparametrový search space BayesOpt prochází. Standard je „RadianceKit defaults„. Pro cílené Mip ladění „Mip subset“. Pro pochopení, jak BayesOpt pracuje, „Bowl demo„.

**W36 Slider „Trial Budget,,**

KDE

Levý sidebar, pod search space pickerem..

## TECHNICKY

Slider od 10 do 200, krok 5. Default 40. To znamená: BayesOpt smí udělat maximálně N trialů. Z nich jsou prvních pár initial samples (Latin Hypercube), zbytek skutečné BayesOpt trialů. Pravidla pro praxi: search space s d dimenzemi potřebuje asi  $10d$  až  $20d$  trialů pro dobré optimum. U 6-dim defaultů tedy 60–120, u 2-dim Mip Subsetu 20–40, u 1-dim Bowl Dema 10–20.

Během běhu je slider deaktivován.

## JEDNODUŠE ŘEČENO

Kolik optimalizačních pokusů maximálně. Více pokusů = lepší řešení, ale stojí to víc času. 40 je dobrý default pro demo objective.

**W37 Slider „Random Seed„**

KDE

Levý sidebar, pod budget sliderem..



TECHNICKY

Slider od 1 do 100, krok 1. Default 42. Seed se předává jak initial Latin Hypercube samples, tak noise komponentě demo objective. Reprodukovatelnost: stejný seed + stejný search space + stejný budget dává přesně identickou trial sekvenci. Užitečné pro „dostanou všichni tvoji kolegové stejný běh, když demo nastavují podle tebe?“. Během běhu deaktivováno.

**JEDNODUŠE ŘEČENO**

Řídí generátor náhodných čísel. Stejný seed = stejný běh — pro reprodukci.

**W38 Chart (Convergence)**

KDE

Středový sloupec okna..



TECHNICKY

Swift Charts diagram se dvěma vrstvami: 1. čára pro „best-value-so-far„ na trial — monotónně stoupající nebo zůstávající křivka v akcentové barvě. 2. bod na trial s individuální objective hodnotou, obarvený podle fáze. Velikost symbolu 40. Tři phase labely: „init“ (šedá), „bo„ (modrá), „restart“ (oranžová).

Malá legenda ukazuje phase barvy nahoře vlevo. Pokud je seznam trialů prázdný (před prvním startem), zobrazí se místo toho empty state s chart ikonou a hintem „Press Start to begin a BayesOpt run.“

**JEDNODUŠE ŘEČENO**

Průběhový chart. Souvislá čára je „nejlépe dosud nalezené řešení“; body jsou jednotlivé pokusy. Pokud čára dlouhou dobu zůstává plochá, BayesOpt našel optimum.

**W39 Table (Trial Log)**

KDE

Pravý sloupec okna..



TECHNICKY

Scroll oblast s lazy stohovanými trial řádky. Na řádek horizontální stack: trial číslo (3místné monospaced, vlevo), hodnota (monospaced, zarovnáno vpravo, 70 pt široké), phase tag (capsule, vyplněný phase barvou při 25 % opacity), volitelně žlutá hvězda, pokud je tento trial aktuálně nejlepší. Auto scroll mechanismus automaticky skáče na konec, jakmile přijde nový trial — takže můžeš live průběh číst u spodního okraje obrazovky, aniž bys musel sám scrollovat.

**JEDNODUŠE ŘEČENO**

Tabulka všech pokusů. Hodnota, fáze, hvězda pro nejlepší. Auto scrolluje s sebou, nové trialy se objevují dole.

**Hlavní okno: průběh ztráty a Gaussian Count (I39–I41, křížový odkaz)**

Tři ze zobrazení Inspektoru v hlavním okně si zaslouží vlastní vysvětlení, protože jsou během běžícího tréninku stále vidět a existují důležitá pravidla, kdy průběh vypadá zdravě. Zobrazení jsou v Inspektoru pod sekci „Loss Chart,“ (viz Kapitolu 2 — Inspektor) a doplňují holdout analýzu z pomocného okna výše.

**Kdy je loss křivka zdravá?** Zdravá loss křivka ukazuje tři fáze: (1) **warmup** — v prvních 200–500 iteracích klesá loss strmě z vysoka (typicky 0.15–0.25 pro L1+SSIM kombinováno podle scény) asi na polovinu. Pokud loss v této fázi NEKLESÁ, většinou je vstup špatný (obrázky poškozené, SfM pozice špatné, počet initial Gaussianů příliš malý). (2) **densifikace** — mezi ~500 a densifyUntilIteration (klasicky 15K, MCMC do 20K nebo 25K) loss klesá dál, často s malými skoky dolů, když densify operace vkládají nové Gaussiany a optimizér je využívá. Gaussian count v této fázi stoupá. (3) **refinement** — poté loss běží do plošícího se tailu. Typické koncové hodnoty: Tanks-&Temples Truck s P4 Quality přistává na  $L1 \approx 0.023$ , Horse s Full Classic V546 na  $L1 \approx 0.0230$ , outdoorové Mip-NeRF360 scény často horší (0.04–0.07).

**Co znamená plató?** Plató (loss křivka běží horizontálně přes několik tisíc iterací) má dvě interpretace: (a) model konvergoval, další trénink už nic nepřinese — to je dobrý případ. (b) model je stuck (lokální minimum, špatná gradient informace, cap na buffer limitu) — špatný případ. Oba vypadají v chartu identicky. Rozlišení: podívej se na Gaussian count. Pokud je také plochý a blízko MCMC capu (např. 150K z 150K u `.fullMCMC`), jsi na limitu — buď cap zvýšit nebo plató akceptovat. Pokud Gaussian count ještě roste, ale loss neklesá, je to stuck.

**Kdy přerušit vs trénovat dál?** Pravidlo: 10K iterací bez zlepšení Min Loss → přerušit, další iterace jsou promarněné. Předtím: můžeš přes Cmd+T (menu Training → Continue Training → +5K iterations) ještě připojit prodloužení, pokud vidíš okrajové zlepšení. Pozor: u MCMC je plató často skutečné — cap je přirozená hranice.

**Gaussian count plató NENÍ signál „hotovo“.** Znamená pouze, že MCMC dosáhlo capu nebo že Classic densifikace je vyčerpáná. Skutečnou otázku „hotovo“ klade až holdout analýza — PSNR/SSIM/LPIPS na nezávislém test setu, vyhodnoceno v holdout okně (W23–W29) nebo přes `--benchmark` flag.

**PSNR/Holdout je pravda, Loss jen proxy.** Loss je relativní metrika: klesá, dokud se tvůj model přizpůsobuje tréninkovým views. Nízký loss ale neznamená automaticky dobrý model — pokud se model nazpaměť naučil tréninkové obrázky (overfitting), byl by loss malý, ale PSNR na neviděných views (holdout) by byla špatná. Proto: pro finální posouzení kvality se vždy dívej na holdout metriky, ne pouze na end loss.

## Pravidlo box

- User Guide a Keyboard Shortcuts jsou statická nápověda — u dotazů na klíčová slova rychlé, pro hloubku použij tuto příručku.
- Manage Storage otevřít, jakmile disk klesne pod 10 % volného místa. Logy a Imports staging jsou obvyklí viníci.
- Pareto Dashboard má smysl až po alespoň třech nebo čtyřech tréninkových reportech. X osa = náklady (Time / Gs), Y osa = kvalita (PSNR / SSIM). Pareto fronta ukazuje efektivní kombinace.
- Holdout Analysis použít, než publikuješ PSNR benchmarky s ostatními — zajistí ti, že tvůj test set je skutečně reprezentativní.
- BayesOpt Console je primárně učební a inspekční nástroj pro definice search space. Pro skutečné tréninkem řízené ladění hyperparametrů použij offline CLI workflow.
- Loss plató a Gaussian count plató je třeba interpretovat odděleně. Cap limit není „hotovo“ signál. Skutečnou kvalitu měří jen holdout PSNR.
- 10K iterací bez zlepšení Min Loss → trénink zastavit.

## KAPITOLA

## Kapitola 6 — Konfigurace tréninku

```
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
  }
}
```

Obrázek 26: Preview předvolba exportovaná jako JSON + zobrazená v TextEditu — pole `id/name/category/version/createdAt/description`, `trainingConfig` se všemi relevantními parametry (`maxIterations` 5000, `densifyUntilIteration` 3500, `ssimWeight` 0.20, `renderScale` 0.50, `strategy` `classic`, `cameraAlignment` `applePhotogrammetry`, `densifyGradThreshold` `2.0e-06`, `opacityResetInterval` 3000, `minOpacity` 0.005, šest `bool` toggľů)

**CO JE NA OBRÁZKU** Typický JSON export předvolby. Top-level pole: `id` (UUID), `name`, kategorie (`classic` | `mcmc` | `sceneClass` | `custom`), verze schématu, timestamp, `free-text`. Vnořený objekt obsahuje pro reprodukovatelnost kritické parametry — při importu se celý blok deserializuje do struktury `TrainingConfig` a defaulty z verze aplikace doplní pole, která v JSON chybí (např. po update aplikace). Kdo předvolbu předává na jiný Mac, posílá jednoduše tento JSON soubor.

Struktura `TrainingConfig` je srdcem každého tréninkového běhu v RadianceKitu. Sdružuje každý parametr, který trénink ovlivňuje — od maximálního počtu iterací přes osm learning rate až po speciální pole pro MCMC, Mip-Splatting, curriculum a scene-aware cap logiku. Editujš ji v sidebaru v sekci Konfigurace tréninku (Expert View), ukládáš ji jako předvolbu nebo předáváš jako JSON export na jiný Mac. Při tréninku se přesně tento objekt zmrazí a předá GPU backendu.

Tato kapitola je referenční materiál pro power-usery a autory skriptů. Listuje všech 81 veřejných polí, 9 statických předvoleb a jednu veřejnou metodu. Zdrojový soubor je `TrainingConfig.swift` — v případě pochybností platí tam uložený doc comment a default inicializéru jako source of truth.

#### POZNÁMKA · UI VS. PRESET/CLI

Pouze 12 z 81 polí má přímý slider, toggle nebo picker v Inspektoru (sandboxed App Store build): **T1, T2, T17, T20, T22, T38, T56–T58, T60, T61, T73**. Zbývajících 69 polí se nastavuje přes zvolenou **předvolbu** (Kapitola 7) a lze je přímo přepsat pouze přes **CLI flag** (viz Kapitola 5). Toto rozdělení je záměrné: defaulty zůstávají stabilní a produkčně ověřené, power-useři mají přesto únikový poklop. Pokud tě nějaké pole obzvlášť zajímá: nejprve nahlédni do Kapitoly 2 (Inspektor) a Kapitoly 5 (CLI), zda ho dosáhneš bez JSON ručních zásahů.

#### Obsah:

1. Iterace (T1–T2)
2. Learning Rates (T3–T10)
3. Densifikace — Classic (T11–T16)
4. Loss (T17–T20)
5. Progrese SH stupně (T21)
6. Výkon (T22–T25)
7. Diagnostika a příprava mraku bodů (T26–T30)
8. Regularizace (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 densifikace (T61–T73)
14. Mip-Splatting (Q1.5) (T74–T76)
15. Adaptivní densifikace (Q5) (T77–T79)
16. Curriculum (Q6) (T80–T81)
17. Statické předvolby (TP1–TP9)
18. Metoda:
19. Které pole na co? (Cheat-sheet)
20. Nebezpečná pole

## Iterace (T1–T2)

### T1 maxiterations

#### DETAILY

**Default:** 30 000 (Initializer), 35 000 ( `.full` ), 200 000 ( `.fullMCMC` ) **Range:** 1 000 – 500 000 (UI slider), žádná tvrdá horní hranice v logice **Defined in:**

#### TECHNICKY

Celkový počet tréninkových iterací, které backend proběhne. Iterace označuje forward render jedné tréninkové kamery, backward pass přes všechny loss komponenty (L1 + SSIM + volitelné regularizace + sky mask) a Adam optimizer step. Toto číslo působí přímo na ostatní schedule: position learning rate sleduje cosine-annealing křivku od 0 buď k `T1` samému nebo k `T49` positionLRScheduleEndIteration; densifikace končí u `T2` densifyUntilIteration; MCMC noise decay končí u `T69` mcmcNoiseDecayEnd; SH degree upgrady se dějí na třech značkách definovaných v `T21`. U klasické densifikace leží empiricky určený sweet spot na 20 000–35 000 iteracích (Sessions 1–32, V546 testy), u MCMC na 60 000–200 000 (V534). Drastické zvýšení nad hodnoty uložené v předvolbě zřídka přináší dodatečnou kvalitu — Adam momentum se sytí a bez LR decay konce loss stagnuje. Naopak podstřelení ~5 000 vede k neúplně zkonvergovaným geometriím (density control má příliš málo času na klonování/štěpení).

#### JEDNODUŠE ŘEČENO

Jak dlouho aplikace počítá. Více iterací = lepší výsledek, ale od určitého okamžiku už ne ztelně lepší, zato mnohem delší. Předvolby jsou zvoleny tak, že bez přemýšlení máš dobrou hodnotu: Quick 1 000, Preview 5 000, Balanced 20 000, Quality 35 000, MCMC Quality 200 000. Pokud sám ladíš: u MCMC můžeš jít vysoko (100 000–200 000), u Classic ne nad 40 000 — pak už nic nepřináší.

## **T2** densifyUntilliteration

### **DETAILY**

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

### **TECHNICKY**

Iterace, od které densifikace přestává. Až sem se Gaussiány klonují, štěpí a prune podle pravidel parametrizovaných v `T11–T16` (Classic) nebo `T67–T70` (MCMC); poté zůstává počet Gaussianů konstantní a optimalizují se už jen pozice, rotace, škály, opacity a SH koeficienty (refinement fáze). V originálním 3DGS paperu leží hodnota na 50 % `T1`, v RadianceKit `.full` předvolbě jen na ~14 % (5 000 z 35 000) — důsledek V310/V338 experimentů, které ukázaly, že po 5 000 iteracích další densifikace výsledek spíše zhoršuje (více floaterů, větší paměťová náročnost, žádný kvalitativní zisk). MCMC oproti tomu nechává relokaci běžet do 80 % `T1` (V504b), protože MCMC neprodukuje škodlivé floatery. Pokud je `T2` zvoleno příliš malé (< 1 000), vznikne příliš málo Gaussianů; příliš velké u Classic (> 50 % `T1`) vede k overgrowth a RGB saturation outlierům (viz outdoor overtraining findings).

### **JEDNODUŠE ŘEČENO**

Do kdy aplikace smí generovat nové Gaussiány. Poté se už jen vyladuje to, co tam je. U klasického tréninku s 35 000 iteracemi je tu 5 000 správná hodnota — vše nad to udělá scénu rozmazanější. U MCMC je to 80 % celkových iterací (tedy 160 000 u běhu 200 000). Pokud měníš Quality předvolbu, tohle pole raději nech být.

## Learning Rates (T3–T10)

### T3 positionLearningRate

#### DETAILY

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

**Defined in:**

#### TECHNICKY

Adam learning rate pro XYZ pozici každého Gaussianu na začátku tréninku (iterace 0). Sleduje cosine annealing křivku a v průběhu tréninku klesá na T4 `positionLearningRateFinal`. Default 0.00016 pochází z originálního 3DGS paperu (Kerbl et al. 2023) a v RadianceKitu se neškáluje ani při zvýšení rozlišení obrazu — pozice se pohybuje ve světovém souřadnicovém systému, ne v pixel prostoru. Výrazné zvýšení ( $> 0.0005$ ) způsobí, že Gaussiany skáčou na dlouhé vzdálenosti a loss se stane nestabilním; hodnoty výrazně nižší ( $< 0.00005$ ) vedou k tomu, že špatně inicializované mraky bodů nikdy nenajdou své místo. V414 testoval zdvojnásobení init hodnoty → 16.8 % horší L1 loss; V544a tuningy potvrdily paper default jako optimální. Pozor: u `.fullMCMC` ponecháváme tuto hodnotu záměrně na defaultu — MCMC potřebuje konstantní learning rates pro svou relokační logiku, takže ladění tady nepřináší nic.

#### JEDNODUŠE ŘEČENO

Jak rychle se splatové body smí pohybovat v prostoru. Standardní hodnota je velmi dobře vyladěná a v podstatě nepotřebuje změnu. Jen pokud ti splaty v obrazu „kymácí“, nebo chybí celý roh, protože se nic nepohne, byla by learning rate bodem k otáčení — ale pak typicky neodpovídá už něco jiného (pozice kamer, počítačční mrak bodů).

**T4** positionLearningRateFinal **DETAILY**

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

 **TECHNICKY**

Koncová hodnota position LR cosine annealing křivky. Je dosažena buď u `T1 maxIterations`, nebo, pokud je nastaveno, u `T49 positionLRScheduleEndIteration`. RadianceKit `.full` předvolba používá 0.000016 — tedy 10× vyšší než paper default 0.0000016. V420 experimenty ukázaly, že 0.5× final hodnoty (0.000008) zhoršuje loss o 6.4 %; V414 ukázal, že 2× init hodnota ho zhoršuje o 16.8 %. Vysoká final hodnota není trade-off, ale záměrná volba: u příliš silného decay ztrácí Gaussiany během refinement fáze schopnost se přizpůsobit nově přidaným densifikačním kandidátům. Přes rozšíření V431/V433 lze schedule fázi zkrátit (`T49 < T1`), takže `T4` je dosaženo už před koncem tréninku a zbytek tréninku běží s konstantní mini-LR — typická konfigurace: `T49 = 20 000`, `T1 = 35 000`, refinement tedy na 0.000016 po 15 000 iterací.

 **JEDNODUŠE ŘEČENO**

Jak pomalou bude position learning rate na konci tréninku. Záměrně jsme to nastavili méně agresivně než originální paper — splaty se ještě do konce mohou trochu kymáčet, to je činí ostřejšími. Pokud na tom ladíš: vyšší = neklidnější splaty na konci, nižší = splaty se nemohou už přizpůsobit, když se objeví nové.

**T5** shDCLearningRate **DETAILY**

**Default:** 0.0025 (Initializer + paper), 0.005 ( `.full` a všechny MCMC předvolby — 2×) **Range:** 0.0001 – 0.05 **Defined in:**

 **TECHNICKY**

Adam learning rate pro DC podíl (degree 0, tedy konstantní albedo) spherical harmonic barvy. SH-DC odpovídá směrově nezávislému základnímu tónu Gaussianu, jaksi „základní barva“. V176 a V188 experimenty našly 2× vyšší než paper default optimální — rychlejší konvergenci barev, právě proto, že u krátkého tréninku (< 5 000 iterací) SH-DC jinak nedojde do formy. Na rozdíl od geometrických LR nemá SH-DC žádný decay; learning rate zůstává přes všechny iterace konstantní (nebo sleduje pouze volitelný extended-phase decay z `T51`). V416 testoval zčtyřnásobení na 0.01 → 6.4 % horší loss u beta2=0.99-Adam.

 **JEDNODUŠE ŘEČENO**

Jak rychle se přizpůsobuje základní barva každého splatu. Hodnotu mění málokdy sám — předvolby mají správnou hodnotu. Vyšší by šlo rychleji, ale může vést k nestabilním barvám.

**T6** shRestLearningRate **DETAILY**

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

 **TECHNICKY**

Adam learning rate pro SH koeficienty vyššího řádu (Degree 1, 2, 3 — tedy view direction závislé barevné podíly, které zajišťují lesklá místa, odrazy a jemné stínování). 20× menší než `T5` per paper konvenci, protože tyto koeficienty rostou kvadraticky v počtu (3 pro degree 1, 5 pro degree 2, 7 pro degree 3 → celkem 15 floatů na Gaussian) a bez menší learning rate by obraz přesyceně. Aktivuje se ve dvou krocích — až do první značky v `T21` `shDegreeUpgradeIterations` je aktivní pouze degree 0 (tedy jen `T5`), poté 1, později 2, nakonec 3. Nízké hodnoty zde jsou obzvláště důležité na scénách s mnoha difuzním osvětlením; u velmi lesklých povrchů (laku aut, voda) se ladění nevyplatí — SH reprezentace sama je omezená.

 **JEDNODUŠE ŘEČENO**

Jak rychle se učí směrově závislé barevné efekty (odrazy, lesk). Standardně velmi malé, jinak by se vše lesklo. Hodnotu raději nech — kdo chce lépe zvládnout lesklá místa, je lépe poslán k MCMC a delšímu tréninkovému času než k této LR.

**T7** opacityLearningRate **DETAILY**

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

 **TECHNICKY**

Adam learning rate pro logit opacity každého Gaussianu. Aplikace ukládá opacity jako neomezenou float hodnotu a transformuje ji sigmoidem do [0, 1]; LR působí v logit space. Paper default 0.05 je po V50 testech (best single run L1 0.1664) obnoven, V71 revertoval V67 0.025. V188 zdvojnásobení na 0.1 dělá pruning efektivnější — mrtvé Gaussiany rychleji klesnou pod `T14 pruneOpacityThreshold`. V418 ukázal: 0.05 s `beta2=0.99` Adam je o 7.1 % horší než 0.1 — interakce s Adam konfigurací není triviální. Nízké hodnoty ( $< 0.01$ ) vedou k tomu, že „dead“ Gaussiany leží věčně dokola a žerou paměť; příliš vysoké hodnoty ( $> 0.5$ ) mohou vést k opacity explosion, proto je logit hodnota v optimizeru clampována na  $[-15, 3]$  (viz poznámku „Opacity Explosion Prevention“ v CLAUDE.md).

 **JEDNODUŠE ŘEČENO**

Jak rychle se splaty stávají průhlednými nebo neprůhlednými. Důležité pro úklid — splaty, které nepřispívají, musí rychle zmizet, aby nevznikla závoj. Standardní hodnota sedí, mění ji jen profíci.

**T8** opacityLearningRateFinal **DETAILY**

**Default:** 0.0 (= „žádný decay,“) **Range:** 0 nebo 0.001 – **Defined in:**

 **TECHNICKY**

Volitelná cosine decay end hodnota pro opacity LR (V427). Pokud 0.0, decay deaktivován a opacity LR zůstává přes celý trénink konstantní na `T7`. V427 testoval decay  $0.1 \rightarrow 0.01$  — výsledek 11.5 % horší loss; revertováno, proto default „vypnuto“. Hypotéza za polem: v refinement fázi by konstantní opacity LR mohla vést k oscilaci, takže splaty, které už dosáhly správné míry transparency, se náhodnými gradient fluktuacemi opět posouvají. Empiricky se to nepotvrzuje — logit clamping logika to stejně zachytí. Pole zůstává dostupné pro budoucí experimenty; i velmi dlouhé MCMC běhy ( $> 500K$  iterací) by z toho mohly profitovat.

 **JEDNODUŠE ŘEČENO**

Zda by se měla opacity learning rate ke konci zmenšovat. Standard: ne. Zkusili jsme to, bylo to horší, necháme deaktivováno. Zůstaň na 0.

**T9** `scaleLearningRate` **DETAILY**

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

 **TECHNICKY**

Adam learning rate pro tři škálové komponenty každého Gaussianu v log space (RadianceKit ukládá  $\log(\text{scale})$ , aby škály zůstaly kladné). Paper default 0.005, v RadianceKitu zdvojnásobeno na 0.01 pro lepší škálovou konvergenci u optimalizovaných LR konfigurací. V423 experiment: 0.005 s  $\beta_2=0.99$  Adam  $\rightarrow$  18.7 % horší loss a viditelně příliš málo Gaussianů (density control nemohl klonovat, protože škálové updaty byly příliš lenivé). Škála řídí rozšíření každého Gaussianu — příliš rychlé učení vede k „needle“ Gaussianům (extrémně dlouhé tenké splaty, viz T34 `scaleRatioPruneThreshold`), příliš pomalé učení nechává splaty zůstat příliš kompaktními a density control musí příliš často štěpit.

 **JEDNODUŠE ŘEČENO**

Jak rychle se přizpůsobuje tvar splatů. Standard je dobrý. Pokud to roztočíš nahoru, dostaneš rád „jehlové“ splaty — extrémně dlouhé tenké kapky, které obraz floatují.

**T10** `rotationLearningRate` **DETAILY**

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

 **TECHNICKY**

Adam learning rate pro čtyři kvaternionové komponenty každého Gaussianu. Kvaternion se v každém optimizer kroku po Adam update opět normalizuje ( $L_2 \text{ norm} = 1$ ) — jinak by kovarianční matice degenerovala. RadianceKit zdvojnásobuje paper default v Quality předvolbách, protože rotation má vůči škále / pozici menší absolutní gradient magnitudy (na jednotkové sféře zůstává každý krok krátký) a bez 2x by rotation v 35 000 iterací byla výrazně pod-konvergovaná. V188 dokumentuje. Na NeRF-Blender scénách (Lego, Chair) se rotation projevuje obzvláště — hrany objektů se vyrovnají správně až po 5 000–10 000 iteracích.

 **JEDNODUŠE ŘEČENO**

Jak rychle se splaty učí otáčet — tedy zaujmout správnou orientaci na povrchu objektu. Standard sedí. Jinak řečeno: pokud splaty vypadají jako šikmo položené kostky místo přilnutí k povrchu, je spíše tréninkový čas příliš krátký, ne tato learning rate příliš nízká.

## Densifikace — Classic (T11–T16)

### T11 `densifyGradThreshold`

#### DETAILY

**Default:** 0.000002 (Initializer, kalibrováno pro 0.5× rozlišení), 0.0000011 ( `.full` , kalibrováno pro 1.0×), 0.000004 ( `.quickTest` , kalibrováno pro 0.25×), 2e-7 ( `.fullClassicPaper` ) **Range:** 1e-8 – 1e-3 (závislé na rozlišení) **Defined in:**

#### TECHNICKY

Threshold pro L2 normu screen-space projektovaného gradientu `dMean2D` , nad kterým je Gaussian označen pro klonování nebo štěpení. Absolutní hodnota závisí přímo na tréninkovém rozlišení — `dMean2D` škáluje zhruba jako  $1/\text{rozlišení}^2$  (více pixelů = menší per-pixel gradienty). Proto každý T22 `trainingRenderScale` stupeň potřebuje kalibrováný `threshold`: 0.25× → 4e-6, 0.5× → 2e-6, 1.0× → 5e-8 ... 1.1e-6 ( `.full` ). Paper default 0.0002 je NDC normalizován a v RadiancesKit world space pipeline není přímo srovnatelný. S ve V440 přidaným T52 `adaptiveDensifyThreshold` flagem lze hodnotu v běhu spočítat z p98 aktuální gradient distribuce — ale V440 to testoval na reálných scénách a produkoval 63 K Gaussianů (katastrofální pruning ztráta); flag zůstává vypnut. Q5 (T77–T79) dodává alternativní adaptivní logiku přes rolling median. **Bezpečné toto pole není** — halvení produkuje 2–4× více Gaussianů (paměťový tlak, OOM riziko); zdvojnásobení může scénu pod-densifikovat.

#### JEDNODUŠE ŘEČENO

Jak citlivá je aplikace, když má rozhodnout, zda je splat příliš málo zobrazen a musí být zmnožen. Nižší hodnota znamená citlivější = více splatů. Vyšší = méně splatů. To je jedna z nejnebezpečnějších hodnot vůbec: příliš nízká a Mac se naplní miliony splatů a může spadnout. Nech pole na pokoji nebo to měň jen po krocích 10 %.

**T12** densifyFromIteration **DETAILY****Default:** 500 **Range:** 100 – 5 000 **Defined in:** **TECHNICKY**

První iterace, od které je densifikace aktivní. Předtím se děje pouze „nahá“ učení na počátečním SfM mraku bodů, aniž by vznikaly nové Gaussiány. Default 500 pochází z 3DGS paperu a dává inicializaci čas, aby se stabilizovala — pokud se densifikuje už od iterace 0, špatně umístěné SfM body se klonují mnohonásobně, ještě než najdou své správné místo. V349 testoval 1000 → mírně horší loss; default je optimální.

 **JEDNODUŠE ŘEČENO**

Kdy aplikace začne poprvé klonovat splaty. Předtím se učí pouze z již existujících bodů. 500 je standard — dává aplikaci dost času, aby se nejprve orientovala, než zmnoží.

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

Kolik iterací leží mezi dvěma densifikačními kroky. V paperu defaultu 100 — každých 100 iterací se vyhodnocuje seznam densify kandidátů, klonuje/štěpí se a současně se odstraňuje seznam prune kandidátů ( $\text{sigmoid}(\text{opacity}) < T_{14} \text{ pruneOpacityThreshold}$ ). V112 testy našly 200 jako optimální pro `.full` — odlehčí GPU, protože méně reorganizačních passů běží, a dává každému Gaussianu více času, aby se po klonovací akci usadil. V417 testoval 100 s  $\text{beta2}=0.99$  → 5.8 % horší (957 K Gaussianů, over-densifikace). U MCMC se totéž pole interpretuje jako relocation interval; viz T67 `mcmcRelocationInterval` pro MCMC specifickou logiku.

 **JEDNODUŠE ŘEČENO**

Jak často aplikace hledá nové splaty. 100 = často, 200 = středně. Vyšší znamená: každý splat má delší čas se zařídit, než se zmnoží. To je dobré. Snížit na 50 může GPU trvale zaměstnat, aniž by to bylo znatelně lepší.

**T14** `pruneOpacityThreshold` **DETAILY**

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

 **TECHNICKY**

Sigmoid opacity threshold, pod kterým je Gaussian při dalším densifikačním kroku smazán. Působí dohromady s `T7 opacityLearningRate` a logit clamp logikou v optimizeru. V393 snížil default z 0.005 na 0.001 v `.full` — důsledek: splaty, které hrají roli jen pod exotickými úhly pohledu, zůstávají déle zachovány a přispívají k SH detailu. V394 testoval 0.0001 → mírně horší (málo pruning, plýtvání paměti). Důležité: density control musí VŽDY prune, i když je kapacita bufferu jinými opatřeními už plná (viz „Density Control Must Always Prune“, v CLAUDE.md) — jinak se hromadí mrtvé Gaussiany a count zamrzne.

 **JEDNODUŠE ŘEČENO**

Kdy splat platí jako „dost průhledný“, aby byl smazán. 0.005 je paper standard, my máme v Quality 0.001 — tedy splatům dávejte déle šanci. To zvládne lépe měkké světlo a slabé stíny. Vyšší (přes 0.01) snižuje počet splatů rychle — může to být užitečné při nedostatku paměti, ale stojí to detail.

**T15** `opacityResetInterval` **DETAILY**

**Default:** 3 000 (Initializer + paper), 100 000 (`.full` = efektivně deaktivováno), 200 000 (`.fullMCMC` = deaktivováno) **Range:** 1 000 – 100 000+ **Defined in:**

 **TECHNICKY**

Po kolika iteracích se opacity všech Gaussianů resetuje na nízkou hodnotu (~0.01) — opatření z 3DGS paperu pro opětovné posouzení „zmrazených“ splatů. V194 ukázal, že s RadianceKit warmup + stochastic training setup + 2× learning rates opacity reset stojí 5.5 % kvality a logit clamp už funkci resetu pokrývá. Proto v `.full` prakticky deaktivováno (100 000 > 35 000 takže nikdy nespuštěno). V421 testoval reset každých 3 000 s `beta2=0.99` → 4.9 % horší; revertováno. U `.fullClassicPaper` (Q1.5-A, paper-věrný test) je záměrně opět nastaveno na 3 000 — to byla jedna z pák, kterými mělo být dosaženo paper-magnitude Gaussian budgetů.

 **JEDNODUŠE ŘEČENO**

Po kolika iteracích aplikace resetuje viditelnost všech splatů na „téměř neviditelné“, — jakýsi resetovací knoflík pro opacity. U nás deaktivováno (hodnota tak vysoká, že se to nikdy nestane), protože jiné mechanismy to dělají zbytečným. Jen u paper-věrných experimentů zapínat.

**T16** maxScreenSize **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 (off) nebo > 0 **Defined in:**

 **TECHNICKY**

Maximální screen space velikost (v projektovaných pixelech), kterou Gaussian smí dosáhnout, než bude násilně rozštěpen. Hodnota je nastavena na 0 (V48 testoval a reverteroval) — RadianceKit density control místo toho používá world space scale threshold z `dMean2D` logiky. Zůstává v katalogu polí, protože budoucí experimenty s Mip-Splatting (T74–T76) nebo scéně-specifickými splatting strategiemi z toho mohly profitovat. Aktivace (hodnota > 0, např. 20) by donutila velmi velké splaty na obrazovce rozdělit se — relevantní u velkých hladkých stěn, kde jeden obří splat nabízí málo detailu.

 **JEDNODUŠE ŘEČENO**

Omezení, jak velký jeden splat na obrazovce smí být. U nás vypnuto. Zapnuto by způsobilo, že obří ploché splaty (např. na stěně) jsou násilně rozloženy na více malých. Nech to vypnuté, pokud výslovně experimenty.

## Loss (T17–T20)

### T17 `ssimWeight`

#### DETAILY

**Default:** 0.2 (Initializer + paper + `.full`), 0.05 (všechny MCMC předvolby) **Range:** 0.0 – 1.0 **Defined in:**

#### TECHNICKY

Váha D-SSIM podílu v kombinované loss funkci  $loss = (1 - \lambda) * L1 + \lambda * D\text{-SSIM}$ , kde  $\lambda = T17$ . 3DGS paper default 0.2 je pro classic densifikaci optimální – V383 testoval 0.3 → 28.9 % horší, V373b potvrdil 0.2 jako sweet spot. Pro MCMC bylo ve V521b/V534 nezávisle zjištěno: 0.05 je optimální, protože MCMC přes svou stochastickou exploraci potřebuje silnější L1 signál – vyšší SSIM váhy by relokační rozhodnutí rozředily. SSIM je výrazně dražší spočítat než L1 (lokální 11×11 okna přes celý obraz); Radiance-Kit používá MPS-akcelerovanou implementaci, která zůstává pod 1 ms na 1080p obraz. Q7 BayesOpt sweepy našly scéně-specifická optima mezi 0.05 (`.outdoorPreset : 0.082`) a 0.171 (`.indoorPreset`).

#### JEDNODUŠE ŘEČENO

Jak důležité aplikace vedle „každý pixel sedí“, najde i „strukтуры jsou podobné“. 0.2 je standard a dodává dobrý obraz. Nižší = pixel-přesnější, ale může dostat měkčí přechody. Vyšší = strukturálně podobnější, ale details se stanou měkčími. Nech předvolby rozhodnout.

### T18 `ssimWeightRefinement`

#### DETAILY

**Default:** 0.0 (= „žádné přepnutí, ponechej `ssimWeight`“), **Range:** 0 nebo 0 – 1.0 **Defined in:**

#### TECHNICKY

Volitelná SSIM hodnota pro refinement fázi po `T2 densifyUntilIteration`. V428 testoval 0.2 → 0.3 v refinementu → 16 % horší loss (jak L1, tak SSIM se zhoršily); revertováno, proto default 0.0. Hypotéza za polem byla, že po densifikaci – když už nevznikají nové Gaussiany – by silnější SSIM podíl maximalizoval strukturální ostrost. Empiricky špatně: zvýšit SSIM váhu znamená nepřímo snížit L1 váhu, a L1 je výrazně výmluvnější signál ve finální refinement fázi. Pole zůstává dostupné pro budoucí experimenty s perceptual loss (T60) nebo edge loss (T19), kde by refinement-specifická loss kompozice mohla mít smysl.

#### JEDNODUŠE ŘEČENO

Speciální nastavení pro druhou tréninkovou fázi (refinement po zmnožení splatů). U 0.0: stejná SSIM váha jako předtím. Ladění empiricky nic nepřináší, proto vypnuto.

**T19** edgeLossWeight **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.001 – 1.0 **Defined in:**

 **TECHNICKY**

V437 experimentální loss: váha Sobel gradient domain L1 loss, který porovnává hrany obrazu přímo (ground truth Sobel vs render Sobel) navíc k L1+SSIM. Hypotéza: edge informace je perceptuální základ kvality obrazu a explicitní term by měl Gaussiany povzbuzovat zasáhnout hrany lépe. Výsledky testů: váha 0.1 → 11 % horší loss, 0.01 → quality-neutrální ale 10 % pomalejší. Sobel pass stojí další MPS forward na ground truth a render. Proto trvale deaktivováno. Budoucí use case: scény s tvrdými umělými hranami (architektura, nábytek, renderingy) by mohly profitovat — Q7 Scene-Class předvolby to ale nevybraly, místo toho škálovaly SSIM váhu.

 **JEDNODUŠE ŘEČENO**

Experimentální přídavek, který bere hrany extra důležitě. Empiricky nic nepřináší. Zůstává vypnuto.

**T20** skyMaskingEnabled **DETAILY**

**Default:** false (Initializer a všechny předvolby) **Range:** boolean **Defined in:**

 **TECHNICKY**

Zapíná Sky Masking. Při tom se v každém obrazu přes Apple Vision Framework (`VNGenerateForegroundInstanceMaskRequest`) sky oblast vymaskovává a loss v této oblasti se nastavuje na nulu. Smysl: outdoor scény často trpí tím, že modré/šedé/bílé sky pixely přivedou aplikaci k tomu, aby umístila Gaussiany přesně tam — co je vnímáno jako „floater“. Bez sky mask by loss v této oblasti nikdy nebyl nulový, protože obloha v obrazu lehce variuje a aplikace se věčně snaží to splatý napodobit. Vision maska se počítá jednou na kameru před tréninkem a drží v RAM. Typicky se aktivuje společně s [T45 skyDomeEnabled](#) (UI logika ve Settings View). U interiérových scén nebo syntetických renderingů nech deaktivováno — maska by tam chybně rozpoznala stropy nebo stěny jako „sky“.

 **JEDNODUŠE ŘEČENO**

Zapne speciální režim pro venkovní záběry: obloha se při tréninku ignoruje, aby se nepokoušelo o její splatý napodobit. Doporučeno pro každou venkovní scénu. U interiéru nebo u 3D renderingů z Blenderu nechej vypnuto.

## Progrese SH stupně (T21)

### T21 shDegreeUpgradeIterations

#### DETAILY

**Default:** [1\_000, 2\_000, 3\_000] (Initializer), [2\_000, 5\_000, 8\_000] (.full, MCMC), [1\_000, 2\_000] (.preview — degree 3 přeskóčen) **Range:** [Int], každá hodnota v [0, maxIterations], monotónně rostoucí **Defined in:**

#### TECHNICKY

Iterace, na kterých se aktivní SH degree přepíná 0→1, 1→2, 2→3. Před první značkou jsou aktivní pouze DC komponenty (tedy T5 shDCLearningRate), po první značce DC + 3 degree-1 koeficienty, po druhé značce + 5 degree-2 koeficientů, po třetí značce všech 15 koeficientů. Paměťová náročnost na Gaussian roste stupňovitě — 4 floaty → 16 floatů → 36 floatů → 64 floatů. Quality předvolby zpožďují upgrade vůči initializer defaultům (V228), protože geometrie má nejprve stabilizovat, než přijdou barevné detaily s vyšší frekvencí. V384 testoval [1K, 2K, 3K] pro .full → 9.3 % horší — potvrzuje delay. .preview kape u degree 2, protože degree 3 v 5 000 iteracích nekonverguje a jen plýtvá optimizer kapacitou. Q6 (T80–T81) nabízí alternativní curriculum logiku, která tento seznam dynamicky přepíše.

#### JEDNODUŠE ŘEČENO

Na jakých bodech tréninku se aplikace učí, že barvy z různých úhlů pohledu mohou vypadat různě (lesklá místa, odrazy). Až pozdě — aby nejprve seděl tvar, pak barva. Hodnoty v předvolbách jsou nastaveny tak, že to dobře funguje. Nic na tom nemění, pokud přesně nevíš proč.

## Výkon (T22–T25)

### T22 trainingRenderScale

#### DETAILY

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

#### TECHNICKY

Render rozlišení při tréninku relativně k původnímu rozlišení tréninkových obrázků. Při 0.5 se každý obraz přepočítává na 50 % šířky × 50 % výšky (tedy 25 % pixelů) a Gaussian rendering se děje v tomto menším rozlišení. Redukuje jak paměťovou, tak výpočetní náročnost kvadraticky. Důležité: `T11 densifyGradThreshold` musí odpovídat zvolenému rozlišení — gradient magnitudy škálují s  $1/\text{rozlišení}$ , proto má `.quickTest` (0.25×) mnohem vyšší threshold ( $4e-6$ ) než `.full` (1.0×,  $1.1e-6$ ). RadianceKit varuje u velmi velkých obrazů a automaticky upravuje — cíl 3 MP. Při extrémních 4K vstupních obrazech by 0.5 nebo dokonce 0.25 dávalo smysl, jinak běží každý Mac jen v CPU compaction.

#### JEDNODUŠE ŘEČENO

Jak velké jsou obrázky při tréninku. 1.0 = originál, 0.5 = poloviční. Poloviční velikost = čtyřikrát rychlejší, ale chybí nejméně detaily. Předvolby volí správnou hodnotu; u extrémně velkých vstupních obrázků (přes 12 megapixelů) aplikace automaticky přepíná dolů.

### T23 resolutionWarmupScale

#### DETAILY

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.1 – **Defined in:**

#### TECHNICKY

V133 optimalizace: trénuj densifikační fázi (iter 0 do `T2`) v nižším rozlišení než refinement fázi. V308 ji pro `.full` opět vypnul, protože při `T22 = 1.0` a cosine annealing byl časový zisk marginální a kvalita minimálně trpěla. Zůstává v katalogu polí, protože u 4K vstupů a dlouhých tréninkových běhů by mohla opět dávat smysl — Q6 curriculum (T80) přebralo podobnou logiku, ale je tam spárováno s LR schedule. Pokud aktivováno a `T80 curriculumResolutionRamp` také true, vyhrává Q6 a tuto hodnotu přepisuje.

#### JEDNODUŠE ŘEČENO

Speciální feature: v první polovině tréninku s menšími obrázky se učít, ve druhé s velkými. Šetří čas. Vypnuto, protože novější Q6 varianta to řeší lépe.

**T24** tileSize **DETAILY****Default:** 16 **Range:** 8, 16, 32 **Defined in:** **TECHNICKY**

Velikost rasterizačních tile v pixelech. Gaussian Splatting rendering je tile-based: obraz se dělí do 16×16 pixel dlaždic, každá dlaždice sbírá pro sebe relevantní Gaussiány, řadí je podle hloubky a vkládá. 16 je standard použitý prakticky všemi 3DGS implementacemi a v RadianceKit Metal kernelech napevno zakódováno; změna této hodnoty by vyžadovala re-kompilaci shaderů a v aktuálním stavu není efektivní. Zůstává jako pole, pokud budoucí engine verze podporuje tile size dynamicky.

 **JEDNODUŠE ŘEČENO**

Interní render parametr. Standard 16, neměnit.

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

Umělé zpoždění mezi tréninkovými iteracemi v milisekundách. 0 = plná rychlost (standard). Vyšší hodnoty činí Mac během tréninku „použitelnějším“, protože GPU/CPU pravidelně dostávají oddechové pauzy — pohodlí používání ostatních aplikací roste, ale tréninkový čas lineárně se zpožděním. Typické hodnoty: 1–2 ms („lehké“ throttling, +5 % tréninkového času, Mac se cítí responzivnější), 5 ms („středně těžký“, +15 % tréninkového času), 10+ ms („Eco“, potenciálně dvojnásobný tréninkový čas). V Inspektoru se nabízí pod „Performance“, ale není ve standardním zobrazení — viz backlog `dev_ux-backlog.md`, který navrhuje ho z Expert View odstranit, protože špatně pochopen drasticky prodlužuje tréninkový čas.

 **JEDNODUŠE ŘEČENO**

Kolik milisekund pauzy aplikace mezi tréninkovými kroky dělá. 0 = žádná pauza, co nejrychleji. Vyšší hodnoty činí Mac během tréninku lépe použitelným — ale trénink pak také trvá déle. Na M3 Ultra nebo Mac Studio to můžeš nechat na 0; na MacBook Air by 2 nebo 5 byla dobrá hodnota.

## Diagnostika a příprava mraku bodů (T26–T30)

### T26 depthDistortionWeight

#### DETAILY

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.0001 – 0.05 **Defined in:**

#### TECHNICKY

V366 experimentální: váha depth distortion regularizačního lossu. Trestá Gaussiany, které jsou podél render paprsku sice hloubkově odstupňované, ale konceptuálně patří k téže ploše — podporuje koncentrované hloubkové distribuce a redukuje floatery. Testy: 0.01 → 4.5 % horší, 0.001 → 8.1 % horší. Teoretická výhoda — vylepšit multi-view konzistenci — se nepromítá do L1 lossu, protože hypotéza implicitně předpokládá, že SfM geometrie je správná a Gaussiany se musejí jen „naskládat“. V praxi je SfM mrak bodů obvykle nejslabší komponentou, ne skládání. Zůstává dostupné pro multi-view datasety s obzvláště čistými pozicemi (Synthetic, Mip-NeRF 360 s ground truth).

#### JEDNODUŠE ŘEČENO

Experimentální funkce pro vyhnutí se více splatům za sebou na tomtéž místě. Neaktivováno, protože testy nic nepřinesly.

### T27 singleViewOverfit

#### DETAILY

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

#### TECHNICKY

Diagnostický flag: pokud true, v každé tréninkové iteraci se musí použít kamera index 0 místo náhodné z camera poolu. Smysl: pokud model nemůže overfittovat ani jediný view (tedy loss na view 0 ani po 10 000 iteracích nepoklesne k nule), je v forward/backward passu fundamentální bug. Tento přepínač byl během vývoje Metal shaderů a differentiable rasterizer kernelů intenzivně používán — V42–V47 fáze. Dnes dostupný už jen jako sanity check, pokud někdo modifikoval backend kód a chce udělat regression test. Přes CLI s `--single-view`.

#### JEDNODUŠE ŘEČENO

Test režim pro vývojáře. Mohou tím kontrolovat, zda aplikace vůbec dokáže učit se z JEDNOHO obrázku. Pro normálního uživatele irelevantní, vždy nech vypnuté.

**T28 maxCameras** **DETAILY**

**Default:** 0 (= „použít všechny kamery,“) **Range:** 0 nebo 1 – N **Defined in:**

 **TECHNICKY**

Diagnostický limit z V43: trénuj jen s prvními N kamerami, ignoruj všechny další. Smysl původně: testovat hypotézu, že příliš mnoho kamer generuje gradient konflikty (příliš mnoho protichůdných loss signálů pro tentýž Gaussian). Výsledek testu: žádná systematická výhoda u umělého omezení — více framů přináší prakticky vždy více kvality. Zůstává jako CLI flag ( `--max-cameras N` ) pro cílené experimenty, např. „funguje trénink na prvních 100 obrázcích 1 500 obrázkového dronového letu?“. V UI neexponováno.

 **JEDNODUŠE ŘEČENO**

Diagnostické pole pro vývojáře — jen prvních N obrázků použít, zbytek ignorovat. Normální uživatel nepotřebuje, hodnota na 0 = všechny obrázky. Více obrázků = lepší výsledek (viz `feedback_more-frames-better.md`).

**T29 maxInitialPoints** **DETAILY**

**Default:** 0 (= „použít všechny SfM body,“) **Range:** 0 nebo 1 000 – 200 000+ **Defined in:**

 **TECHNICKY**

V54 pojistka: limituje počet počátečních SfM bodů, se kterými trénink startuje. Husté COLMAP rekonstrukce mohou produkovat > 60 000 bodů, což u velkých init škál vede ke 200–300 Gaussianům na pixel-overlap — to dělá „mlhové pole“, ve kterém trénink nekonverguje. Subsampling na ~16 000 bodů (hard-cap logika v tréninkovém enginu) přivádí počáteční hustotu na úroveň, kterou používá referenční 3DGS, a dramaticky redukuje overlap. U velmi hustých SfM se nastavuje automaticky; přes CLI s `--max-points N`.

 **JEDNODUŠE ŘEČENO**

Kolik počátečních bodů z rekonstrukce kamer se používá. U velmi hustých rekonstrukcí (více než 60 000) aplikace automaticky limituje na 16 000 — jinak je na začátku příliš mlhy. Nemusíš to nastavovat; aplikace to řídí.

**T30** cameraClusterOutlierMultiplier **DETAILY**

**Default:** 10.0 (všechny předvolby — nikdy přepisováno) **Range:** 1.0 – 100.0 **Defined in:**

 **TECHNICKY**

Multiplikátor pro camera cluster outlier filtr, zavedený v Phase 3.10 A.1. Před tréninkem tréninkový engine počítá centroid všech pozic kamer a maximální vzdálenost kamery od centroidu. SfM body, jejichž vzdálenost od centroidu překračuje  $\text{multiplier} \times \text{maxCameraDistance}$ , se zahazují jako outliery. Default 10x zachovává chování před fází 3.10. Subtilní bug: tighter SfM (kamery těsněji u sebe) → menší → menší threshold → více bodů se zahazuje jako outlier. Looser SfM → větší threshold → méně bodů se zahazuje. To je jedna z příčin Phase 3.9 funnel-vs-training anti-korelace: lepší SfM může downstream vést k horšímu tréninku, protože je zabito příliš mnoho počátečních bodů. Pole leží jako CLI override (`--camera-cluster-outlier-multiplier`) pro A.3 sweepy; v UI neexponováno. Hodnoty pod 5 jsou zpravidla příliš restriktivní, nad 20 bezúčinné.

 **JEDNODUŠE ŘEČENO**

Speciální filtr, který zahazuje body z rekonstrukce, které leží daleko od mraku kamer. 10 = aplikace je velkorysá, drží téměř všechno. Zvýšení může mít smysl, pokud vzdálené body (hory v dáli) v obrazu vypadají jako plovoucí chuchvalce. Nižší nastav jen v nouzi — ztrácíš tím detail v dáli.

## Regularizace (T31–T37)

### T31 coarseToFineBlurRadius

#### DETAILY

**Default:** 0 (= deaktivováno) **Range:** 0 nebo 1 – 10  
**Defined in:**

#### TECHNICKY

V369 experimentální: box blur radius, který se na začátku densifikační fáze aplikuje na ground truth obraz a lineárně do konce densifikace ( T2 ) snižuje na 0. Hypotéza: coarse-to-fine trénink — nejprve hrubé struktury, pak detaily — by měl dodávat stabilnější geometrii. Testy:  $r=3 \rightarrow 9.6\%$  horší,  $r=1 \rightarrow 5.1\%$  horší. Důvod selhání: densifikace rozhoduje na základě image domain gradientů, a blurování redukuje právě signály, které jsou důležité pro „zde se musí klonovat,“. Zůstává v katalogu polí pro budoucí testy s jiným density control schématem.

#### JEDNODUŠE ŘEČENO

Experimentální „nejprve-hrubě-potom-detailně,“ režim. Nic nepřinesl, zůstává vypnuto.

### T32 scaleRegWeight

#### DETAILY

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.0001 – 0.05 **Defined in:**

#### TECHNICKY

V370 experimentální: L1 regularizace na world space škálu. Trestá Gaussiany, kteří se stávají příliš velkými — zabraňuje „mega splatům,“, které pokrývají celé stěny jediným Gaussianem. Testy: 0.01  $\rightarrow$  200 % horší loss (2 M Gaussianů, totální exploze), 0.001  $\rightarrow$  214 % horší. Důvod: scale regularizace přichází do konfliktu s density control — menší škály znamenají, že je potřeba více Gaussianů, takže density control štěpí častěji, což opět znamená více gradient práce. Disabled, ale dokumentováno pro Mip-Splatting experimenty (T74): v tomto kontextu by mohla mít smysl spodní hranice škály.

#### JEDNODUŠE ŘEČENO

Regularizace, která splaty nutí zůstat malé. V testech vyvolala exploze splatů (miliony splatů). Neaktivovat.

**T33** anisotropyRegWeight **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.0001 – 0.05 **Defined in:**

 **TECHNICKY**

V445 experimentální: penalty na poměr  $\max(\text{scale})/\min(\text{scale})$ , má zabránit extrémně dlouhým „needle“, Gaussianům, které jsou vnímány jako floatery. Testy: 0.01 → 69 % horší, 0.001 → 15 % horší. Důvod: regularizace nutí splaty směrem ke „kulaté“ formě, což je na ploché ploše (stěna, stůl, podlaha) přesně špatně — tam je plochý široký Gaussian efektivnější než kulový. Disabled. V549f nabídl s T34 `scaleRatioPruneThreshold` alternativní cílenější přístup, který byl rovněž revertován.

 **JEDNODUŠE ŘEČENO**

Regularizace, která trestá příliš dlouhé tenké splaty. Zní rozumně, v testech byla ale horší. Vypnuto.

**T34** scaleRatioPruneThreshold **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 5.0 – 100.0 (typicky 10.0 – 30.0) **Defined in:**

 **TECHNICKY**

Experimentální post-training pruning, který maže každý Gaussian, jehož poměr  $\max(\text{scale})/\min(\text{scale})$  překračuje zde nastavený lineární threshold. Cílí na extrémně dlouhé „needle/disc“, floatery, které nelze eliminovat samotnou regularizací. V testu pruning odstraňoval floatery jak doufáno, ale současně i smysluplné ploché splaty na stěnách a podlahách — obraz se stal děravějším. Proto defaultně vypnuto, CLI flag (`--scale-ratio-prune N`) zůstává pro cílené experimenty. Doporučené hodnoty pokud někdo přesto chce testovat: 30 (velmi konzervativní, odstraní pouze extrémní outliers), 10 (agresivní, stojí detail).

 **JEDNODUŠE ŘEČENO**

Pokus vyfiltrovat velmi dlouhé splaty po tréninku. Bylo netto-negativní — floatery pryč, ale i detail pryč. Vypnuto.

**T35 opacityRegWeight** **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.0001 – 0.05 **Defined in:**

 **TECHNICKY**

V446 experimentální: binary cross entropy penalty, který opacity táhne k 0 nebo 1 (tedy pryč od „polo-transparentní“). Hypotéza: ostřejší opacity distribuace by zlepšila čistotu obrazu. Test s T33 kombinován → regularizace stojí kvalitu, obě deaktivována. Disabled. Pozor: v 1.4.3-Beta se vynořil bug, který přesně toto pole měl v default value změně (Initializer = 0.01), což vedlo k mass extinction Gaussianu (460 K → 5 v jedné iteraci). Od 1.4.4 pevně zakotveno na 0.0 jako default.

 **JEDNODUŠE ŘEČENO**

Regularizace, která splaty činí buď úplně transparentními, nebo úplně solidními. Nic nepřináší, může být dokonce nebezpečná (1.4.3 bug mass extinction). Nech na 0.

**T36 opacityDecayFactor** **DETAILY**

**Default:** 0.0 (Initializer = deaktivováno), 0.9995 ( `.full` , `.classicBalanced` — HTGS standard) **Range:** 0 (off) nebo 0.95 – 1.0 **Defined in:**

 **TECHNICKY**

V546 implementace HTGS schématu (Hierarchical Time Gating, Eurographics 2025): každé T37 `opacityDecayInterval` iterací se sigmoid opacity každého Gaussianu násobí tímto faktorem. 0.9995 × 100 aplikací dává ~95 % zůstatek na densifikační fázi — lehký, ale stálý sestupný tlak na všechny opacities, který slaběji přispívající Gaussiany spolehlivě nechává klesnout proti T14 `pruneOpacityThreshold`. Výsledek: 14 % lepší L1 loss na Horse Full (3-trial avg V546) oproti V438 bez decay. Aktivní pouze během densifikační fáze (do T2), poté trénink běží bez decay dál, aby opacities etablované v refinementu zůstaly stabilní. U MCMC se nepoužívá (MCMC má vlastní mechanismy přes T67 `mcmcRelocationInterval` + T68 `mcmcDeadOpacityThreshold`).

 **JEDNODUŠE ŘEČENO**

„Měkké vyblednutí“, všech splatů přes tréninkový čas. Činí neaktivní splaty rychleji transparentními, takže při úklidu mizí. Byl nejdůležitější quality páka V546 updatu: 14 % lépe. V Quality předvolbě zabudováno. Sám otáčet nedoporučeno, protože přesně vyladěno.

**T37** opacityDecayInterval **DETAILY****Default:** 50 **Range:** 10 – 500 **Defined in:** **TECHNICKY**

Iterační interval, ve kterém se `T36` `opacityDecayFactor` aplikuje. HTGS paper default 50, v `.full` ponecháno. Dlouhé intervaly (>200) ruší efekt částečně, protože mezi dvěma aplikacemi se děje dost gradient updatů, takže opacity opět stoupá. Kratší intervaly (<20) činí decay příliš agresivním. Aktivní pouze v densifikační fázi.

 **JEDNODUŠE ŘEČENO**

Jak často se „blednutí“ aplikuje. 50 = každých 50 iterací trochu blednoucí krok. Sedí.

**Refinement (T38–T44)****T38** gradientAccumulationSteps **DETAILY****Default:** 1 (= „jeden view na Adam krok“) **Range:** 1 – 8 **Defined in:** **TECHNICKY**

V424 feature: počet views, jejichž gradienty se akumulují, než se provede Adam update. Při `> 1` aplikace běží na samostatné „unfused“ backward project cestě, která sčítá gradienty v samostatném bufferu; finální aplikace škáluje s  $1/N$ , aby magnituda zůstala konstantní. V424 testoval 2-view → quality neutrální, ale 10 % pomalejší (protože unfused cesta je dražší než fused). Revertováno pro `.full`, ale pro MCMC záměrně použito — `.fullMCMC` běží se zapnutým, ale V544a testy ukázaly, že s ním quality gap k Classic klesá na 5 % (místo 11 %). V default inicializéru 1, v aktuální předvolbě 1, zůstává CLI flag (`--accum-steps N`).

 **JEDNODUŠE ŘEČENO**

Kolik obrázků aplikace sleduje, než upraví splaty. 1 = každý obrázek samostatně. Vyšší = několik obrázků prohlédnout najednou a pak aplikovat průměr. Ve standardním případě nic nepřináší; u MCMC může 2 trochu pomoci.

**T39** testViewIndices **DETAILY**

**Default:** `[]` (= prázdné, všechny views se používají pro trénink) **Range:** `Set<Int>`, libovolná podmnožina camera indexů **Defined in:**

 **TECHNICKY**

V546 feature: set camera indexů, které se NEPOUŽÍVAJÍ pro trénink, ale šetří se jako holdout pro PSNR/SSIM/LPIPS vyhodnocení. Automaticky se nastavuje, když je aktivní `--benchmark` CLI flag: pak každý osmý view, počínaje indexem 0 (LLFF standard, identický s Mip-NeRF-360 a 3DGS paper konvencemi). Bez benchmarku prázdné — trénink používá všechny views. **Pozor:** manuální nastavení tohoto pole bez porozumění indexům může učinit benchmark nepoužitelným (např. pokud jsou všechny indexy nastaveny na hodnoty nad N, zatímco existuje pouze N-50 views → žádné holdouts → žádné vyhodnocení). U vlastního exportu předvolby se testViewIndices nepersistuje, protože je scéně-závislé a jinak by mezi různými datasey nechalo nesmyslné hodnoty.

 **JEDNODUŠE ŘEČENO**

Které obrázky se při tréninku „šetří“, aby se později použily pro měření kvality. Nenastavuješ to sám; flag `--benchmark` to dělá automaticky (každý osmý obrázek je test). Pokud nastavíš vlastní indexy: nebezpečné, může to benchmark zfalšovat.

**T40** refinementPruneInterval **DETAILY**

**Default:** 0 (= deaktivováno) **Range:** 0 nebo 100 – 5 000 **Defined in:**

 **TECHNICKY**

V425 feature: každých N iterací během refinement fáze (po `T2`) se spustí další prune pass, který odstraňuje Gaussians s `sigmoid(opacity) < T41 refinementPruneOpacityThreshold`. Smysl: během densifikace se konají pravidelné density control calls, poté už ne — Gaussians, jejichž opacity dál klesá, ale zůstávají v bufferu. V425 testoval a revertoval: dodatečné pruning korelovalo s V426 (Two-Phase Densification, rovněž zhroucením do 0 Gaussianů). Disabled. CLI flag dostupný pro experimenty; pokud aktivováno, 1 000 nebo 2 000 jsou smysluplné hodnoty.

 **JEDNODUŠE ŘEČENO**

Dodatečné uklízení během refinement fáze. Nic nepřináší, zůstává vypnuto.

**T41 refinementPruneOpacityThreshold****DETAILY**

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

**TECHNICKY**

V425b: samostatný opacity threshold pro refinement pruning. Po densifikaci dosáhla většina Gaussianů výrazně vyšší opacity (> 0.001), takže standardní T14 pruneOpacityThreshold by byl příliš šlovivavý. Pokud je T40 aktivní, toto pole určuje vlastní threshold. Při 0.0 se dál používá T14. Relevantní pouze když T40 > 0.

**JEDNODUŠE ŘEČENO**

Threshold pro dodatečné refinement uklízení (viz T40). Obě pole neaktivní, takže irelevantní.

**T42 midTrainingCompactificationIterations****DETAILY**

**Default:** [] (= deaktivováno) **Range:** [Int], hodnoty v (densifyUntilIteration, maxIterations) **Defined in:**

**TECHNICKY**

V549 feature: explicitní iterační body během refinement fáze, na kterých běží compactification pass (odstraňuje sigmoid(opacity) < 0.01 + outlier-scale Gaussians, stejná logika jako T56 postTrainingCompactification). Smysl: dlouhé refinement fáze mohou ukazovat konfeti/floater akumulaci, jejíž SH pak overfittuje na view-specifické artefakty. Typická konfigurace pokud aktivováno: [10000, 20000, 30000] pro 40K Classic. **ALE:** V549 A/B testy na Family datasetu ukázaly ve všech konfiguracích horší L1: [10K, 20K, 30K]@0.01 → -48 % count ale +36 % L1; [20K, 30K]@0.005 → -44 % count ale +45 % L1; [20K, 30K]@0.001 → -17 % count ale +87 % L1. Proto disabled. CLI flag --mid-compact "10000,20000" dostupný, pokud někdo preferuje vizuální floater tradeoff (méně konfeti v náhledu) oproti loss regresi.

**JEDNODUŠE ŘEČENO**

Mezilehlé úklidové akce během tréninku. V testech to úklid učinil konečný výsledek horším (sice méně floaterů, ale také méně detailu). Vypnuto, lze zapnout přes CLI, pokud tě floatery trápí víc než trochu rozmazanější obraz.

**T43 frustumCullEnabled** **DETAILY****Default:** false **Range:** boolean **Defined in:** **TECHNICKY**

V549b feature: po tréninku se odstraní všechny Gaussiany, které leží mimo sjednocení všech tréninkových camera frust. Takové Gaussiany nebyly nikdy omezeny loss signálem a jsou vždy floatery. Obzvláště efektivní pro scény, ve kterých novel-view leží za nebo vedle camera path (např. zadní strana lineárního dronového letu) — floatery tam nikdy v tréninkové fázi nejsou viditelné, ale při pozdějším pohybu v 3D vieweru velmi ano. V549b A/B na dronových letech pozitivní výsledky, proto jako opt-in dostupné. Default false, protože u object captures s plným orbit coverage frustum union obsahuje celou scénu a feature nic neodstraňuje — v Settings se nabízí pod „Floater Reduction“, a v Q9 Outdoor předvolbě implicitně přes T44 `frustumCullExpansion` testováno (Q7 BayesOpt to ale neaktivoval, protože Outdoor Sky Dome řeší stejný problém lépe).

 **JEDNODUŠE ŘEČENO**

Speciální filtr pro dronové lety nebo lineární záběry: po tréninku se mažou splaty, které žádná kamera „neviděla“. Volitelně zapnutelné v Settings. U jednoduchých záběrů objektů zbytečné.

**T44 frustumCullExpansion** **DETAILY****Default:** 1.1 **Range:** 1.0 – 2.0 **Defined in:** **TECHNICKY**

NDC margin pro T43 `frustumCullEnabled`. 1.0 by řezal přesně na okraji obrazu, což by viklavé splaty u okraje příliš zkrátil. 1.1 = 10 % padding přes přesný framing kamery — dává trochu tolerance pro hraniční pixely, které by mohly být v lehce posunutém novel-view přece viditelné. Hodnoty > 1.2 činí cull prakticky neúčinným, protože rozšířený frustum zahrnuje mnohem více prostoru.

 **JEDNODUŠE ŘEČENO**

Jak striktně výše popsany filtr ořezává. 1.1 = trochu bezpečnostního odstupu od okraje obrazu. Nech hodnotu.

## Sky-Dome (T45–T48)

### T45 skyDomeEnabled

#### DETAILY

**Default:** false (Initializer + všechny předvolby kromě P9 Outdoor) **Range:** boolean **Defined in:**

#### TECHNICKY

V549e feature: před startem tréninku se generuje sférický mrak bodů (Fibonacci sphere s T46 sample points), v poloměru  $T47 \text{ skyDomeRadiusMultiplier} \times \text{scene\_extent}$  kolem středu scény umístěn a inicializován barvami z sky-maskovaných pixelů všech tréninkových kamer (viz T20 skyMaskingEnabled). Tyto Sky-Dome Gaussiany se vkládají na začátek Gaussian bufferu a během tréninku „zamrznou“, (position/scale/rotation gradienty = 0, jen SH a opacity zůstávají optimalizovatelné). Efekt: místo černých „konfeti“ oblastí v dáli vidí uživatel v novel-views skutečnou oblohu. V549e MVP funguje na dronových a krajinných scénách velmi dobře; v P9 Outdoor předvolbě default-on. U interiérových scén nech vypnuté — sféra by visela nesmyslně mimo místnost.

#### JEDNODUŠE ŘEČENO

Zapne umělou „nebeskou kopuli“, kolem scény. Dělá venkovní záběry mnohem hezčí: místo černých chuchvalců u okraje obrazu aplikace zobrazuje skutečnou oblohu. Povinné pro dronové lety a krajiny, nesmyslné pro interiéry.

### T46 skyDomeSampleCount

#### DETAILY

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

#### TECHNICKY

Počet Fibonacci sphere sample points na Sky-Dome sféře. Vyšší hodnoty → hustší Sky-Dome (lepší u velkých rozlišení a velmi viditelné oblohy), ale více paměťové náročnosti. 5 000 je sweet spot pro 4K renderingy; u nižších rozlišení stačí 2 000–3 000. Body se podle cosine distance ke každému tréninkovému camera view vektoru inicializují odpovídajícími sky-maskovanými pixely — sample points, jejichž view cone nevidí žádná kamera, zůstávají s nízkou počáteční opacitou vzadu, ale v tréninku se nemění (zamrznuté).

#### JEDNODUŠE ŘEČENO

Jak hustá je umělá obloha. 5 000 bodů obvykle stačí. Více = lepší přechod z dálky, ale stojí to trochu paměti.

**T47 skyDomeRadiusMultiplier****DETAILY**

**Default:** 30.0 (Initializer + většina předvoleb), 59.0 (P9 Outdoor, Q7 BayesOpt optimum) **Range:** 5.0 – 200.0 **Defined in:**

**TECHNICKY**

Poloměr Sky-Dome sféry relativně ke scéně rozsahu (= střední vzdálenosti mezi pozicemi kamer). 30 = koule má 30-násobný průměr camera mraku. Příliš malé (< 5) → Sky-Dome interferuje se scénou samotnou (např. Sky-Dome splat přistane v popředí); příliš velké (> 100) → float32 ztráta přesnosti na Sky-Dome pozicích, což spouští render glitche v dálce. Q7 BayesOpt na Bicycle (Mip-NeRF 360) našel 59.0 jako scéně-specifické optimum pro outdoor — to naznačuje, že standardní 30.0 je pro hluboké krajiny příliš malé a Sky-Dome pixely v hraničních oblastech obrazu jsou viditelné jako „stěna“.

**JEDNODUŠE ŘEČENO**

Jak daleko by měla umělá nebeská kopule být. 30 = celkem daleko. U velkých krajin je 50–60 lepší (Outdoor předvolba to dělá automaticky). Příliš malé by bylo, jako bys měl chuchvalce přímo před objektivem.

**T48 frozenGaussianCount****DETAILY**

**Default:** 0 (= žádné zmrazené Gaussiany) **Range:** 0 nebo 1 – T46 **Defined in:**

**TECHNICKY**

Počet Gaussianů na začátku bufferu, jejichž position/scale/rotation gradienty se v optimizeru nastavují na nulu — zůstávají přes celý trénink prostorově ztuhlí. Density control je nesmí klonovat, štěpit ani prune. Použito pro Sky-Dome injection (viz T45): pokud je Sky-Dome zapnutý, toto pole se automaticky nastavuje na T46 skyDomeSampleCount. Manuální nastavení je možné (např. pro zmrazení předem umístěného mraku bodů z LiDAR skenu), ale v UI ne přímo přístupné. Důležité: prvních N Gaussianů v bufferu jsou vždy zmrazené — pořadí v bufferu rozhoduje, ne explicitní index.

**JEDNODUŠE ŘEČENO**

Kolik splatů na začátku je pevných a nesmí se pohnout. Nastavuje se automaticky na počet Sky-Dome bodů, když je Sky-Dome zapnutý. Sám otáčet nepotřebuješ.

## Adam + LR schedule (T49–T55)

### T49 adamResetIteration

#### DETAILY

**Default:** 0 (= deaktivováno) **Range:** 0 nebo 100 –  
**Defined in:**

#### TECHNICKY

V430 feature: iterace, na které se Adam optimizer momentum akumulátory (m1, m2) resetují na nulu. Bias korekce poté běží s `iter - adamResetIteration` místo s `iter`. V430 testoval reset na 5 000 (po konci densifikace) → 12.8 % horší loss. Důvod: Adam momentum, které se nahromadilo během densifikace, nese informaci o typických gradient magnitudách a zrychluje refinement fázi. Zahodit ho stojí prvních ~500 iterací refinementu na konvergenci. Disabled. Zůstává CLI flag pro výzkumné experimenty.

#### JEDNODUŠE ŘEČENO

Reset knoflík pro interní Adam optimizer „paměť“. V testech to škodilo, zůstává vypnuto.

### T50 positionLRScheduleEndIteration

#### DETAILY

**Default:** 0 (Initializer = „použij maxIterations“), 20 000 ( `.full` — cosine končí u 20K přes `maxIter=35K`), 30 000 ( `.fullClassicPaper` ) **Range:** 0 nebo 1 000 – **Defined in:**

#### TECHNICKY

V431 feature: iterace, na které position LR cosine annealing křivka dosáhne svého minima. Pokud 0, identické s `T1 maxIterations`. Pokud > 0, schedule běží do této hodnoty a poté zůstává konstantní na `T4 positionLearningRateFinal`. To umožňuje „extended refinement phase“, s minimální, ale konstantní learning rate — pomalu vyladuje pozice bez opětovného decay. `.full` to dělá (schedule konec na 20K, trénink běží do 35K), V434c/V434d potvrdily: 15K a 25K obě zhruba stejně, 20K minimálně optimální. Používá se ve spojení s `T51` pro modifikaci i non-position LR v extended phase.

#### JEDNODUŠE ŘEČENO

Kdy aplikace přestane snižovat position learning rate dál. Pokud nižší než maximální iterace, běží poté konstantní mini-rate — to vyladuje velmi pomalu, ale velmi stabilně. V Quality předvolbě zabudováno, nemusíš otáčet.

**T51** extendedPhaseLRDecay **DETAILY**

**Default:** 0.0 (= deaktivováno, konstantní LR) **Range:** 0 nebo 0.01 – 1.0 **Defined in:**

 **TECHNICKY**

V433 feature: minimální multiplikátor pro non-position LR (scale, rotation, opacity, SH) v „extended phase„ — tedy: poté co je dosaženo T50 a position LR je už na T4. Pokud 0.1, scale/rotation/opacity/SH jsou samy cosine decay od 1.0 (= jejich standardní LR) na 0.1× svého standardu. Pokud 0.0 (default), zůstávají konstantní. V457 testoval plný decay (0.0 = decay-do-nuly) proti žádný decay a našel: avg 0.0400 (2 runy) = stejný loss jako V438 bez decay. Chování čistší s decay, ale ne měřitelně lepší. Proto disabled. Zůstává v CLI jako `--nonpos-lr-scale F`.

 **JEDNODUŠE ŘEČENO**

V pozdní refinement fázi i barvo-  
vé a tvarové learning rates udělat  
menšími. Činí trénink „stabilněj-  
ším“, ale empiricky ne lepším.  
Vypnuto.

**T52** adaptiveDensifyThreshold **DETAILY**

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

 **TECHNICKY**

V440 experimentální: pokud true, aplikace v každém densifikačním kroku počítá p98 aktuální gradient distribuce a používá ho jako dynamický threshold (clampovaný na alespoň 0.5× konfigurované hodnoty z T11, aby moc neunikla). Hypotéza: automatické přizpůsobení aktuální scénové fázi by činilo density control robustnějším — např. na začátku přísnější pruning, později uvolněnější, nebo opačně. V440 testoval a revertoval: katastrofální pokles na 63 K Gaussianů (mass pruning, protože p98 je v prvních iteracích extrémně vysoké a pak skoro nic threshold nepřekročí). Pevný threshold je už dobře kalibrován, dynamické přizpůsobení škodí víc, než přináší. Q5 (T77) nabízí alternativní adaptivní logiku přes rolling median, která problém obchází.

 **JEDNODUŠE ŘEČENO**

Adaptivní verze densify thresholdu. V testech katastrofální (počet splatek spadl na 63K). Vypnuto. Q5 má lepší variantu toho.

**T53 mergeAfterDensification** **DETAILY**

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

 **TECHNICKY**

V438 feature: na konci densifikační fáze (iter `T2` ) se provede jednorázový merge pass, který spojuje blízko sebe ležící Gaussiany s podobnou škálou a barvou. Redukuje počet Gaussianů typicky o 5–15 % bez viditelné ztráty kvality. Smysl: po intenzivním klonování vznikají clustery kvazi-identických Gaussianů, které nic nového nepřispívají — merging uvolňuje kapacitu optimizéru pro jiné oblasti. Standard v Classic Quality předvolbách. U MCMC se nepoužívá, protože MCMC díky své relokační logice takové clustery vůbec nenechá vzniknout.

 **JEDNODUŠE ŘEČENO**

Na konci splatové množovací fáze klony, které jsou skoro identické, sloučit. Redukuje datovou množinu bez viditelného efektu. Standardně v Quality předvolbě zapnuto.

**T54 densifyPhase2FromIteration** **DETAILY**

**Default:** 0 (= deaktivováno) **Range:** 0 nebo `T2` – `T1` **Defined in:**

 **TECHNICKY**

V426 experimentální: umožňuje druhou densifikační fázi, která startuje po refinement pauze na této iteraci a běží do `T55` . Hypotéza: po refinement fázi mají gradient akumulátory stabilnější magnitudy a mohou přesněji říci, které oblasti ještě potřebují další Gaussiany. V426 testoval a revertoval: Two-Phase densifikace spadla do 0-Gaussians-cascade-failure (s V425 refinement pruning kombinováno zničilo buffer). Disabled. CLI flag dostupný pro experimenty.

 **JEDNODUŠE ŘEČENO**

Druhé kolo množování po pauze. V testech to zničilo splatovou populaci. Vypnuto.

**T55** densifyPhase2Untillteration **DETAILY****Default:** 0 **Range:** 0 nebo T54 – T1 **Defined in:** **TECHNICKY**

Konec V426 Two-Phase densifikace. Relevantní pouze když `T54 > 0`. Obě pole dohromady disabled.

 **JEDNODUŠE ŘEČENO**

Konec druhého kola zmnožování (viz T54). Obě vypnuto.

## Post-processing + Apple AI (T56–T60)

**T56** postTrainingCompactification **DETAILY****Default:** true (ve všech production předvolbách), false ( `.quickTest` , `.preview` ) **Range:** boolean **Defined in:** **TECHNICKY**

V443 feature: po konci tréninku se Gaussiany s `sigmoid(opacity) < 0.01` tvrdě odstraňují (prakticky nepřispívají k obrazu). Redukuje Gaussian count typicky o 58 % a export velikost souboru o 55 % bez viditelné ztráty kvality. Standardně v production předvolbách aktivní — konečný výsledek má být dodáván co nejkompaktněji. V `.quickTest` vypnuto, protože diagnostický běh se stejně neexportuje. Na rozdíl od T42 `midTrainingCompactificationIterations` (V549) probíhá compactification až na konci — refinement může do té doby všechny Gaussiany používat.

 **JEDNODUŠE ŘEČENO**

Uklid po tréninku: téměř neviditelné splaty se odstraňují. Dělá export soubor zhruba poloviční bez ztráty kvality. Povinná funkce, nech vypnutou jen u diagnostických běhů.

**T57** metalFXUpscaling **DETAILY****Default:** false **Range:** boolean **Defined in:** **TECHNICKY**

V444 feature: aktivuje Apple MetalFX Spatial Upscaler místo bilineární interpolace v 3D viewer output. Pokud je tréninkové rozlišení < velikost náhledu (např. trénink na 0.5x, náhled v plném rozlišení), může MetalFX poskytovat výrazně ostřejší obraz. Mění se živě v náhledu, není potřeba re-training. Vylučuje se s T58 `mpsLanczosScaling` — MetalFX má přednost. Doporučení: zapnout, pokud obraz v vieweru působí „rozmazaně“, oproti očekávanému detailu.

 **JEDNODUŠE ŘEČENO**

Apple ML-based vyostření obrazu v 3D vieweru. Pomáhá, pokud jsi trénoval v nižším rozlišení a výsledek ukazuješ v plném obrazu. Live toggle, vyzkoušej.

**T58** mpsLanczosScaling **DETAILY****Default:** false **Range:** boolean **Defined in:** **TECHNICKY**

V444 feature: `MPSImageLanczosScale` pro viewport škálování místo bilineární interpolace. Lanczos je klasický sinc-based resampling postup, který dodává výrazně ostřejší výsledky než bilineární s minimálním overheadem. Live toggle. Je přepsáno T57, pokud jsou oba zapnuté.

 **JEDNODUŠE ŘEČENO**

Klasický postup pro vyostření v 3D vieweru (Lanczos). MetalFX (T57) je ML-based a obvykle lepší; Lanczos je méně agresivní alternativa.

**T59** `livePreviewInterval` **DETAILY**

**Default:** 50 (Initializer a většina předvoleb) **Range:** 0 (off) nebo 10 – 5 000 **Defined in:**

 **TECHNICKY**

Jak často se během tréninku aktualizuje 3D viewer s aktuálními Gaussiany. 50 = každých 50 iterací nový render ve vieweru — dobré pro sledování postupu, aniž by zpomalovalo trénink. 0 = viewer se vůbec neaktualizuje (background trénink, max rychlost). Typické přizpůsobení: u `.quickTest` dolů na 10 (chce se vidět každý krok), u dlouhých MCMC běhů nahoru na 500–2000 (update overhead v součtu znatelný).

 **JEDNODUŠE ŘEČENO**

Jak často se 3D náhled během tréninku aktualizuje. 50 = každých 50 iterací. Vyšší = méně často = trochu rychleji, ale méně často vidíš postup. 0 = žádný náhled (pro maximální tempo).

**T60** `perceptualLossWeight` **DETAILY**

**Default:** 0.0 (= deaktivováno) **Range:** 0 nebo 0.001 – 0.5 **Defined in:**

 **TECHNICKY**

V444 future-feature: váha perceptuálního loss termu přes MPSGraph (VGG-like malá síť). Zachytávala by strukturální a texturní podobnost na vyšší sémantické úrovni než L1+SSIM — typické v research pipelines, kde „pixel-perfect“ je méně důležité než „vypadá realisticky“. Implementace ještě nedokončená (code stub k dispozici, ale forward pass neimplementován). Default 0.0. Zůstává v katalogu polí pro budoucí aktivaci; CLI flag `--percep-weight F` rezervován.

 **JEDNODUŠE ŘEČENO**

Plánovaná funkce, která s AI pomocí cílí na „přirozený vzhled“, místo „pixelově přesného“. Ještě není hotová implementace.

## MCMC densifikace (T61–T73)

### T61 densificationStrategy

#### DETAILY

**Default:** `.classic` (Initializer + Classic předvolby),  
`.mcmc` (všechny MCMC předvolby + Scene-Class)  
**Range:** `.classic` nebo `.mcmc` **Defined in:**

#### TECHNICKY

Vybírá mezi Classic densifikací (klonování/štěpení/prune, Kerbl et al. 2023) a MCMC densifikací (Stochastic Gradient Langevin Dynamics s relokací, Kheradmand et al. NeurIPS 2024). Při `.classic` se vyhodnocují T11–T16, při `.mcmc` T62–T73. Pozor při změně: Classic defaulty a MCMC defaulty jsou úplně jinak kalibrované — kdo v Expert View flipne picker, aniž by načel odpovídající předvolbu, riskuje 1.4.3 bug-style mass extinction (460 K → 5 v jedné iteraci, protože MCMC opacity reg na 0.01 zabije Classic opacities). Proto jsou MCMC init defaulty záměrně „rozředené“ (všechny reg hodnoty 0.0).

#### JEDNODUŠE ŘEČENO

Jaký algoritmus se používá ke zmnožování splatů. Classic = původní metoda (rychlá, mnoho splatů). MCMC = novější metoda (pomalejší, mnohem méně splatů, ale kompaktnější). Předvolby volí správné. Sám přepínej, jen pokud i odpovídající předvolbu (P5–P7 nebo P8–P10) načteš.

**T62** mcmcMaxGaussians **DETAILY**

**Default:** 150 000 (Initializer + `.fullMCMC` + `.mcmcBalanced`), 100 000 (`.mcmcPreview`), 1 500 000 (`.fullMCMCMip` — Mip-Splatting varianta s 10x rozpočtem), 1.19 M (`.renderPreset`), 1.25 M (`.outdoorPreset`), 670 K (`.indoorPreset`) **Range:** 0 (= „použít kapacitu bufferu,“) nebo 10 000 – 5 000 000 **Defined in:**

 **TECHNICKY**

Tvrdá horní hranice pro počet Gaussianů u MCMC strategie. Počet roste graduálně o `T70 mcmcGrowthRate` (typicky 5 %) per relocation step až po tento cap. V473/V531 našly 150 K jako sweet spot — nad 200 K rozředuje splatovou kvalitu (příliš mnoho malých redundantních Gaussianů), pod 100 K zůstává scéna pod-densifikovaná. U velmi velkých scén (např. 1 545 obrázkový dronový let s 158 K SfM-init) je 150 K příliš nízké — proto rozšíření 1.4.5 `T72 mcmcCapMultiplier` + `T73 mcmcAutoScaleByScene`. Q7 BayesOpt našel scéně-specifická optima mezi 670 K (Indoor) a 1.25 M (Outdoor). U hodnoty 0 engine používá plnou kapacitu bufferu jako cap.

 **JEDNODUŠE ŘEČENO**

Maximální počet splatů u MCMC. 150 000 je standard a stačí pro většinu scén. Outdoor a Render předvolby (P8, P9) jdou na 1+ milion pro detailně bohatší scény. Zvýšení může přinést detail, stojí paměť; snížení je spíše nouzová brzda.

**T63** mcmcNoiseScale **DETAILY**

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

 **TECHNICKY**

Multiplikátor pro Gaussovo rušení, které se v každé MCMC iteraci přidává k pozici každého Gaussianu (SGLD logika). Vyšší = více exploração (Gaussiany více putují, potenciálně najdou lepší místa), nižší = více exploitation (Gaussiany zůstávají tam, kde jsou už dobří). V467 a V536 potvrdily 5e-5 jako optimální — 1e-5/2e-5 příliš málo exploração, 1e-4 příliš mnoho (splaty se rozprásují). Cosine decayuje přes tréninkový čas do `T69 mcmcNoiseDecayEnd` — na konci decay oblasti je rušení efektivně 0 a Gaussiany konvergují.

 **JEDNODUŠE ŘEČENO**

Kolik náhodného „kymáčení,“ aplikace splatům povoluje, aby si samy našly nejlepší místo. Standardní hodnota je optimálně otestovaná. Pokud to zvyšuješ, splaty se stanou neklidnými.

**T64** mcmcOpacityRegWeight **DETAILY**

**Default:** 0.0 (= deaktivováno v RadianceKit defaultech, paper: 0.01) **Range:** 0 nebo 0.001 – 0.05  
**Defined in:**

 **TECHNICKY**

MCMC specifická L1 penalty na opacity. Paper default 0.01 (tlačí nepoužívané Gaussiany k nule, činí je dostupnými pro relokaci). V464b ale ukázal: bez reg je to v RadianceKitu měřitelně lepší (Session 28 potvrdila). Důvod: s T68 mcmcDeadOpacityThreshold definované pruning kritérium stačí samo — dodatečná L1 penalty nutí i hodnotné nízko-opacity Gaussiany umírat. Proto default 0. **Pozor:** v 1.4.3-Beta buildu byl initializer default chybně 0.01, což vyústilo v mass extinction bug (viz T61 vysvětlení); od 1.4.4 fixováno na 0.0.

 **JEDNODUŠE ŘEČENO**

MCMC speciální regularizace. Vypnuto, protože jiný MCMC mechanismus (threshold v T68) to už pokrývá. Nech na 0.

**T65** mcmcScaleRegWeight **DETAILY**

**Default:** 0.0 (= deaktivováno, paper: 0.01) **Range:** 0 nebo 0.001 – 0.05 **Defined in:**

 **TECHNICKY**

MCMC specifická L1 penalty na škálové eigenvalues. Paper default 0.01. V464b: bez reg lepší, stejné odůvodnění jako T64. Disabled ve všech RadianceKit MCMC předvolbách. Pozor jako u T64: 1.4.3 bug.

 **JEDNODUŠE ŘEČENO**

Jako T64, ale pro velikost splatu. Vypnuto.

**T66** **mcmcRelocationInterval** **DETAILY**

**Default:** 100 (Initializer + všechny MCMC předvolby, paper standard), 155 (P9 Outdoor — Q7 BayesOpt optimum) **Range:** 50 – 500 **Defined in:**

 **TECHNICKY**

Iterační interval, ve kterém MCMC relokuje mrtvé Gaussiany (sigmoid(opacity) < T68 mcmcDeadOpacityThreshold) na nové pozice. V537 testoval 50 (příliš disruptivní, loss kolísá) a 200 (marginálně horší, MCMC ztrácí reaktivitu). 100 je optimální. Q7 BayesOpt na Bicycle našel 155 jako scéně-specifické optimum pro outdoor — mírně delší intervaly dávají Adam více času integrovat nově umístěné Gaussiany, než je další reloc event dostane pod tlak.

 **JEDNODUŠE ŘEČENO**

Každých kolik iterací MCMC mrtvé splaty někam jinam posune. 100 je standard. Sám otáčet nepotřebuješ — Outdoor předvolba má už optimální hodnotu.

**T67** **mcmcWarmupIterations** **DETAILY**

**Default:** 500 **Range:** 100 – 5 000 **Defined in:**

 **TECHNICKY**

Počet počátečních iterací, ve kterých ještě neprobíhá MCMC relokace. Až po tomto warmupu začíná reloc logika. Smysl: v prvních iteracích opacity hodnoty ještě nejsou ustálené — pokud by se hned reloc start, Gaussiany by byly umísťovány na špatná místa a hned znovu pohybovány, což ničí Adam momentum. Paper default 500. RadianceKit přebírá tuto hodnotu, protože V464b ukázal, že je robustní.

 **JEDNODUŠE ŘEČENO**

Kolik iterací MCMC nejprve smí „dorazit“, než začne přestavovat splaty. 500 je standard a sedí.

**T68** `mcmcDeadOpacityThreshold` **DETAILY**

**Default:** 0.005 (Initializer, paper standard), 0.01 (`.fullMCMC` a všechny MCMC předvolby — V535 optimum) **Range:** 0.001 – 0.05 **Defined in:**

 **TECHNICKY**

Sigmoid(opacity) threshold, pod kterým je Gaussian „mrtvý“, a může být relokován. V535 našel 0.01 jako optimální (0.005 marginální, 0.02 horší). Vyšší = agresivnější reloc (více Gaussianů se pohybuje), nižší = opatrnější. 0.01 odpovídá zhruba „0.5 % vizuální viditelnost“. P10 Indoor používá přes Q7 BayesOpt 0.0142 jako optimum.

 **JEDNODUŠE ŘEČENO**

Od jaké průhlednosti je splat „mrtvý“, takže MCMC ho jinak přesune. 0.01 je optimální v našich testech. Sám otáčet nepotřebuješ.

**T69** `mcmcNoiseDecayEnd` **DETAILY**

**Default:** 0 (Initializer = „žádný decay“), 160 000 (`.fullMCMC` = 80 % z 200K), 96 000 (`.mcmcBalanced` = 80 % z 120K), 40 000 (`.mcmcPreview`) **Range:** 0 nebo 1 000 – **Defined in:**

 **TECHNICKY**

Iterace, na které je `T63` `mcmcNoiseScale` rušení úplně utlumeno na nulu (cosine decay od iter 0 sem). V497c/V502 našly 80 % `maxIterations` optimální — dává MCMC dost času pro exploraci, ale nechává posledních 20 % na konvergenci bez rušení. 0 = konstantní rušení přes všechny iterace (zřídka rozumné, MCMC pak nemůže konvergovat).

 **JEDNODUŠE ŘEČENO**

Kdy náhodné „kymácení“, splatů přestává. V MCMC předvolbách na 80 % celkových iterací — nejprve explorace, pak konvergence. Nech hodnotu.

**T70** mcmcGrowthRate **DETAILY**

**Default:** 0.05 (paper standard = 5 %) **Range:** 0.01 – 0.2 **Defined in:**

 **TECHNICKY**

Růstová rychlost MCMC populačního targetu per relocation step. Logika: při každém reloc eventu se cíl populační velikosti zvyšuje o  $(1 + \text{growthRate})$ , dokud se nedosáhne `T62 mcmcMaxGaussians` (nebo varianta škálovaná přes `T72/T73`). V512/V522 našly 0.05 jako optimální — vyšší hodnoty vedou k příliš rychlému růstu (Gaussiany se vkládají, než je Adam momentum stihne integrovat), nižší k pod-densifikovaným scénám na konci.

 **JEDNODUŠE ŘEČENO**

Jak rychle počet splatek u MCMC roste. 5 % per krok je optimální. Nech hodnotu.

**T71** mcmcSigmoidK **DETAILY**

**Default:** 100.0 **Range:** 10.0 – 500.0 **Defined in:**

 **TECHNICKY**

Sigmoid sharpness parametr pro MCMC noise attenuation. V SGLD kroku je per-Gaussian rušení tlumeno — vysoko opaké Gaussiany (jejichž logit je pozitivní) dostávají exponenciálně méně rušení než nízko opaké.  $K = 100$  je ostré, tedy přechod z „plné-noise“ na „žádné-noise“ se děje velmi rychle kolem opacity 0.5. V484–V487 našly  $K = 100$  optimální — menší hodnoty (10–50) nechávají i vysoko opaké Gaussiany kymácet (ničící konvergované Gaussiany), větší ( $> 500$ ) činí přechod umělý tvrdý a mrtvé Gaussiany se vůbec nepohybují.

 **JEDNODUŠE ŘEČENO**

Speciální parametr, který určuje, jak ostře MCMC rozlišuje mezi „dost průhledným k přesunu“, a „solidní, nesahat“. Standardní hodnota je optimální. Neotáčet.

**T72** mcmcCapMultiplier **DETAILY**

**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 (= deaktivováno) nebo 1.0 – 10.0 **Defined in:**

 **TECHNICKY**

1.4.5 feature: scéně-adaptivní cap škálování. Pokud je `T73 mcmcAutoScaleByScene true`, efektivní cap se počítá jako (clampované na kapacitu bufferu). Pozadí: u velkých scén (např. 1 545 obrázkový dronový let → 158 K SfM-init) je `T62 = 150 000` příliš nízké — density control by vůbec nemohl růst. S multiplier 3.0 se cap u tohoto příkladu škáluje na 474 K (158 K × 3.0). Q7 BayesOpt našel scéně-specifická optima: outdoor profituje z vysokého multiplikátoru (5.32 → ~830 K cap u 156 K bicycle init), indoor se spokojuje s 1.76 (stěny syt rychleji). Kompletní řešení capu viz metoda.

 **JEDNODUŠE ŘEČENO**

Multiplikátor, který splat cap automaticky přizpůsobuje velikosti scény. Velká scéna = více počátečních bodů = vyšší cap. Standard 3× sedí pro většinu scén; Outdoor předvolba jde na 5× (velké hloubkové rozsahy), Indoor na 1.76× (stěny beztak omezují).

**T73** mcmcAutoScaleByScene **DETAILY**

**Default:** true (Initializer + všechny MCMC předvolby) **Range:** boolean **Defined in:**

 **TECHNICKY**

1.4.5 feature: master switch pro scene-aware cap logiku (viz T72 +). Pokud false, používá se výhradně `T62 mcmcMaxGaussians` jako cap (zpět k 1.4.4 chování). Standardně zapnuto, protože mass extinction problémy u velkých scén z 1.4.3 by jinak vrátily. Ručně deaktivovat jen, pokud explicitně chceš nastavit tvrdý cap — např. pro trénink 150 K varianty, jejíž konečná velikost je plánovatelná.

 **JEDNODUŠE ŘEČENO**

Zapne automatické přizpůsobení splat capu velikosti scény. Standardně zapnuto. Vypnuté nech jen, pokud sám chceš mít přesně určitý počet splatů.

**Mip-Splatting (Q1.5) (T74–T76)**

**Status:** Q1.5 byl 2026-05-25 po 14 autonomních iteracích + overnight 1.5M confidence check zavržen jako „closed no-win“, (max  $\Delta@2\times = +0.27$  dB, originální gate vyžadoval  $\geq +1.5$  dB průměr přes  $0.5\times/2\times$ , FAIL na 0/11 pair-scenes). Pole zůstávají

**opt-in** pro výzkumné experimenty; všechny production předvolby mají vypnuto. Viz verdikt: docs/plans/2026-05-25-phase-q1.5-final-verdict.md.

## T74 useMipSplatting

### DETAILY

**Default:** false (všechny production předvolby), true (.fullMCMCMip — výzkumný sourozenec) **Range:** boolean **Defined in:**

### TECHNICKY

Aktivuje Mip-Splatting (Yu et al. CVPR 2024): 3D smoothing filtr + 2D filtr +  $\alpha$  kompenzace, který omezuje per-Gaussian frekvenci na Nyquist hranici hustoty sampling rate tréninkové kamery. Teoretický cíl: eliminace aliasingu při renderingu v off-training škálách (0.5x nebo 2x tréninkového rozlišení). V preprocess a backward projection shaderech aktivováno, funkčně správně ověřeno v Q1.5-D testu. Ale: originální akceptační gate ( $\Delta@1x \geq +0.3$  dB A avg( $\Delta@0.5x$ ,  $\Delta@2x$ )  $\geq +1.5$  dB) nebylo dosaženo na žádné z 11 pair-scenes. Maximálně pozorováno: family 750K classic  $\Delta@2x = +0.270$  dB. Outdoorové scény (Truck, Flowers) dokonce ukázaly zhoršení 1x a 0.5x. Hypotéza: 3D smoothing konkuruje s MCMC relokací u high-Gs. Pole zůstává pro budoucí multi-scale re-eval se správnou Mip-NeRF-360 metodologií (viz O3 backlog v benchmark cestě).

### JEDNODUŠE ŘEČENO

Antialiasing filtr z 2024 paperu. Teoreticky skvělé, prakticky v našich testech nic nepřineslo a občas dokonce škodilo. Zůstává dostupné pro experimentátory, ale nedoporučujeme. Vypnout.

## T75 mipSmoothing3DScale

### DETAILY

**Default:** 0.2 (paper default) **Range:** 0.05 – 1.0 **Defined in:**

### TECHNICKY

3D smoothing škálový parametr (Yu et al. §3.3, paper default 0.2). Větší = více world space hlazení per Gaussian (= více antialiasingu, ale i více blur na default škále), menší = ostřejší ale náchylnější k aliasingu. Konzultuje se pouze pokud T74 useMipSplatting = true. V Q1.5 testech dále neoptimalizováno — A/B gate prohrál už s paper defaultem 0.2, další sweepy by byly zbytečné.

### JEDNODUŠE ŘEČENO

Mip-Splatting parametr. Pokud Mip nemáš zapnutý, irrelevantní.

**T76 mipFilter2DVariance** **DETAILY**

**Default:** 0.3 (= přesně V242 legacy chování) **Range:** 0.1 – 1.0 **Defined in:**

 **TECHNICKY**

2D Mip filter variance, která se přidává k  $\Sigma_{2D}$  diagonále (variance přímo, ne kvadratura). 0.3 je přesně V242 legacy hodnota, která byla před Mip-Splatting hardcoded v kernelu. Pokud T74 `useMipSplatting = false`, kernel tuto hodnotu kompletně ignoruje a zapisuje napevno 0.3 — takže baseline nemůže regrese (Codex round-1 S3-1 záruka). Pokud zapnuto, použije se zde nastavená hodnota. Zůstává v katalogu polí pro Mip sweepy.

 **JEDNODUŠE ŘEČENO**

Další Mip-Splatting parametr. U Mip vypnutém: irrelevantní.

**Adaptivní densifikace (Q5) (T77–T79)****T77 adaptiveDensification** **DETAILY**

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

 **TECHNICKY**

Q5 feature: rolling median tracker jako alternativa k pevnému T11 `densifyGradThreshold`. Pokud true, v každém densify kroku se aktuální threshold přepíše `median(posledních N avgGrad samples) × T79 adaptiveDensifyMultiplier`. `N = T78 adaptiveWindow`. Striktnější než V440 p98 (katastrofální 63 K pruning past), median + 2× sedí zhruba na p70–p80 gradient distribuce v steady state. Q5 testy: samostatně FAIL 0/3 scén, ale společně s Q6 (viz T80/T81) PASS 1/3 scén — bundle Q5+Q6 byl 2026-05-25 propuštěn jako opt-in a aktivuje se přes CLI `--adaptive-densify`. Q6 je při tom „carrier“, quality zisku, Q5 přispívá spíše ke stabilitě.

 **JEDNODUŠE ŘEČENO**

Samoučící se densify threshold. Místo pevně nastavené citlivosti aplikace se přizpůsobuje scéně. Samostatně testováno není lepší, ale s curriculum z Q6 dohromady ano. Obě společně zapnout nebo obě vypnout.

**T78 adaptiveWindow** **DETAILY****Default:** 1 000 **Range:** 100 – 10 000 **Defined in:** **TECHNICKY**

Rolling median window v densification eventech (NE iteracích — každý T13 `densifyInterval` step dodá sample). Default 1 000 — u to znamená posledních 100 000 tréninkových iterací přispívá k median, tedy typicky celá tréninková historie do zde. Raná fáze (před T78 samples): tracker vrací nil → fallback na pevný threshold T11. Relevantní pouze pokud zapnuto.

 **JEDNODUŠE ŘEČENO**

Kolik starých densify kroků se započítává do median pro T77. Standard 1000 je dobrý. Relevantní jen pokud je Q5 adaptivní zapnuté.

**T79 adaptiveDensifyMultiplier** **DETAILY****Default:** 2.0 **Range:** 1.0 – 4.0 **Defined in:** **TECHNICKY**

Multiplikátor na rolling median pro adaptivní threshold. Default 2.0 odpovídá zhruba p70–p80 typické gradient distribuce. Nižší = agresivnější růst (více klonů), vyšší = přísnější (méně klonů). Q5 testy v rozsahu 1.5–3.0 — 2.0 nejlepší default. Relevantní pouze pokud zapnuto.

 **JEDNODUŠE ŘEČENO**

Faktor pro T77/T78. Standard 2.0 = přísnější než typický medián. Neotáčet.

## Curriculum (Q6) (T80–T81)

### T80 curriculumResolutionRamp

#### DETAILY

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

#### TECHNICKY

Q6 feature: tréninkové rozlišení startuje na 0.5× a přepíná u T50 `positionLRScheduleEndIteration / 2` (nebo T1 `maxIterations / 2`, pokud T50 není nastaveno) na T22 `trainingRenderScale`. Používá v Q1.5.1 vyvinutou `resize/restoreImageBuffers` infrastrukturu. Přepisuje T23 `resolutionWarmupScale`, pokud aktivováno. Q6 je propuštěn jako „carrier quality zisku“, v Q5+Q6 bundle (viz T77) — postupné zvyšování rozlišení dává aplikaci čas najít hrubou geometrii v nižším rozlišení, než přejde k jemné detailní práci. Přes CLI: `--curriculum-resolution`.

#### JEDNODUŠE ŘEČENO

„Nejprve hrubě, pak jemně“, pro tréninkové rozlišení. Poloviční rozlišení v první polovině, pak plné rozlišení. Pomáhá v určitých scénách, v jiných ne — nejlépe s T81 spolu zapnout.

### T81 curriculumSHProgression

#### DETAILY

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

#### TECHNICKY

Q6 feature: přepisuje T21 `shDegreeUpgradeIterations` s `[maxIter/4, maxIter/2, maxIter*3/4]`, rozkládá tedy SH upgrade rovnoměrně přes tréninkový čas místo front-to-load. Hypotéza: stabilní geometrie se etabluje před color detail explosion, což view-direction závislé lesklé efekty přesněji umísťuje. Q5+Q6 dohromady PASS 1/3 scén, Q6 jako carrier zisku (Q5 samo FAIL). Přes CLI: `--curriculum-sh`.

#### JEDNODUŠE ŘEČENO

„Nejprve tvar, pak barva“, — lesklé efekty se otevírají až pozdě v tréninku, takže splaty nejprve najdou svou pozici a velikost. S T80 dohromady zapnutelné; samostatně nepřináší tolik.

## Statické předvolby (TP1–TP9)

Zde pouze strukturální rozdíly proti default inicializéru. Plný marketingový popis 11 UI předvoleb P1–P11 najdeš v Kapitole 7.

**TP1** `.preview`

 **DETAILY**

Diagnostická/preview předvolba pro systémy  $\geq 10$  GB RAM. Overrides oproti Initializeru: - 30 000  $\rightarrow$  5 000 - 15 000  $\rightarrow$  3 500 (70 % maxIter) -  $1.6e-6 \rightarrow 1.6e-5$  (10x vyšší, méně agresivní decay) - několik LR 2x (V176) - 3 000  $\rightarrow$  100 000 (efektivně off, V172: reset zničí krátké tréninky) - [1K, 2K, 3K]  $\rightarrow$  [1K, 2K] (V182: degree 3 nekonverguje v 2K iter) - 1.0  $\rightarrow$  0.5

 **JEDNODUŠE ŘEČENO**

Jakékoli počáteční posouzení nově importované série obrázků — 2–3 min čekání, poté výsledek stačí pro binární otázku „vyplatí se Quality běh?„.

**TP2** `.full`

 **DETAILY**

Production-Quality Classic. Overrides: - 30 000  $\rightarrow$  35 000 (V550: 40K testy Truck overtraining +10.7 % Gs při -1.3 % L1) - 15 000  $\rightarrow$  5 000 (V310 sweet spot, V338 7K horší) - všechny LR 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 % zlepšení) - 50 (nezměněno, V546) - false  $\rightarrow$  true (V438) - 0  $\rightarrow$  20 000 (V431) - true (V443, už initializer default pro `.full`)

 **JEDNODUŠE ŘEČENO**

Jakýkoli standardní foto záběr (objekt, malá místnost, socha) s  $< 500$  obrázky. 14 % loss zlepšení V546 slibovaná proti V438 byla 3-trial průměrována na Horse Full potvrzena.

**TP3** `.fullClassicPaper`

 **DETAILY**

Q1.5-A test sourozenec TP2, paper-věrný Classic. Overrides oproti TP2: - 35 000  $\rightarrow$  30 000 (paper standard) - 5 000  $\rightarrow$  15 000 (paper: 50 % maxIter) -  $1.6e-5 \rightarrow 1.6e-6$  (paper default) - několik LR zpět na paper defaulty (0.05, 0.005, 0.001) -  $1.1e-6 \rightarrow 2e-7$  (kalibrováno pro ~1–2M Gs na Bicycle) - 200  $\rightarrow$  100 (paper) - 0.001  $\rightarrow$  0.005 (paper default) - 100 000  $\rightarrow$  3 000 (paper §5.2, rizikové — může vyvolat V194 regresi) - 0.9995  $\rightarrow$  0.0 (paper nemá decay) - 20 000  $\rightarrow$  30 000 (cosine běží na 100 % maxIter)

 **JEDNODUŠE ŘEČENO**

Q1.5 výzkumné experimenty, které potřebují paper-magnitude Gaussian budgety (1–2 M) pro Mip-Splatting testy. Po Q1.5 „closed no-win“ verdiktu zůstává předvolba pro advanced users dostupná, ale není production doporučená.

**TP4** `.fullMCMC` **DETAILY**

Production-Quality MCMC. Overrides oproti Initializeru: - 30 000 → 200 000 (V534, MCMC potřebuje 5x více iter než Classic) - 15 000 → 160 000 (V504b 80 % maxIter) - 1.6e-6 → 1.6e-5 - LR schedule jako TP2 (všechny 2x) - 0.2 → 0.05 (V521b/V534: MCMC potřebuje silnější L1 signál) - [1K, 2K, 3K] → [2K, 5K, 8K]

- `.classic` → `.mcmc` - 150 000 (v Initializeru už, v předvolbě

potvrzeno) - 5e-5 (V467/V536 optimální) - 0.005 → 0.01 (V535 optimální) - 0 → 160 000 (80 % maxIter, V497c/V502) - 3.0 (v Initializeru už) - true (v Initializeru už) - 3 000 → 200 000 (efektivně off, MCMC používá reloc místo reset)

 **JEDNODUŠE ŘEČENO**

Webová dodávka, object captures s nárokem na detail, dronové lety (i když pak je P9 Outdoor ještě lepší). 71 % méně Gaussianů než Classic při srovnatelném L1.

**TP5** `.fullMCMCMip` **DETAILY**

Q1.5-D test sourozenec TP4, s Mip-Splatting + paper-magnitude MCMC rozpočtem. Overrides oproti TP4: - `mcmcMaxGaussians` 150 000 → 1 500 000 (10x, paper magnitude) - `useMipSplatting` false → true (Mip-on)

 **JEDNODUŠE ŘEČENO**

Všechna ostatní pole identická s TP4. Q1.5 D-PASS na Bicycle 2026-05-24 (přeruší 12-iter multi-scale FAIL streak). Q1.5 finální verdikt 2026-05-25 přesto closed-no-win — Mip-Splatting zisk není reprodukovatelný přes 11 pair-scenes. Předvolba zůstává opt-in.

**TP6** `.classicBalanced` **DETAILY**

Mid-tier Classic. Overrides oproti TP2: - 35 000 → 20 000 (V149: 20K = 30K při 33 % méně času) - 20 000 → 0 (cosine běží na maxIter = 20K, žádná extended phase)

 **JEDNODUŠE ŘEČENO**

Standardní případy s kratší dobou čekání. V149 identifikován jako sweet spot.

**TP7** `.mcmcPreview`**DETAILY**

MCMC diagnostika. Overrides oproti 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)

**JEDNODUŠE ŘEČENO**

Rychle vidět MCMC výsledek, abys posoudil, zda se TP4 nebo Scene-Class předvolba vyplatí.

**TP8** `.mcmcBalanced`**DETAILY**

Mid-tier MCMC. Overrides oproti TP4: - 200 000 → 120 000 (V518) - 160 000 → 96 000 (80 %) - 160 000 → 96 000 (80 %)

- 3.0 → 2.5 (mezi Preview 2.0 a Full 3.0)

**JEDNODUŠE ŘEČENO**

MCMC bez plného 200K běhu. ~120 K iterací je dobrý kompromis mezi kvalitou a čekáním.

**TP9** `.quickTest`**DETAILY**

Čistý funkční test. Overrides oproti Initializeru:

- 30 000 → 1 000 - 15 000 → 500 - 2e-6 → 4e-6 (kalibrováno pro 0.25× rozlišení) - 100 → 50 - 3 000 → 100 000 (off, protože příliš krátké) - 1.0 → 0.25

**JEDNODUŠE ŘEČENO**

Sanity check „startuje trénink vůbec smysluplně?“. Trvání < 30 s na M3 Ultra. Vypadá garantovaně rozmazaně.

## Metoda: resolveMcmcMaxGaussians

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

**TECHNICKY** Jediný source of truth pro otázku „kolik Gaussianů smí MCMC maximálně vyrůst?“. Počítá se ze tří vstupů: konfigurovaného `T62 mcmcMaxGaussians` (s `mass extinction floor 150 000`, pokud 0), `initialPointCount` (počet SfM init bodů) a `bufferCapacity` (předalokovaná velikost Gaussian bufferu). Logika:

1. `base = T62 > 0 ? T62: 150_000` (mass extinction floor chrání proti initializer default bugům jako 1.4.3 mass extinction incident)
2. Pokud `T73 mcmcAutoScaleByScene && initialPointCount > 0 && T72 mcmcCapMultiplier > 0`:
  - `scaled = max(base, ceil(initialPointCount × T72))` jinak
3. Pokud `bufferCapacity > 0`: `return min(scaled, bufferCapacity)`
4. Jinak `return scaled`

Příklad: Bicycle (Mip-NeRF 360, 194 foto framů) → SfM init ~156 K bodů,  
`T62 = 150 000`, `T72 = 5.32`, kapacita bufferu 8 M. Resolved cap =  $\min(8M, \max(150K, \text{ceil}(156K \times 5.32))) = \min(8M, 830K) = 830$   
 K. To je efektivní růstový cap, ke kterému se drží MCMC relokační logika.

**JEDNODUŠE ŘEČENO** Počítá skutečný maximální počet splatů u MCMC. Bere tvé nastavení, dívá se, kolik bodů má tvá scéna na začátku, a škáluje s `Multiplier`, pokud je automatické přizpůsobení zapnuto. Tak se cap přizpůsobí scéně, místo aby pro malou a obří scénu vynucoval stejnou hodnotu. Metodu nemusíš volat sám — trénink ji používá interně.

## Které pole na co? (Cheat-sheet)

Cíl	Pole k otáčení
Více detailu v dáli	<code>T62 mcmcMaxGaussians</code> vysoko, <code>T72 mcmcCapMultiplier</code> 5+
Více detailu obecně (Classic)	<code>T1 maxIterations</code> vysoko ( $\leq 40K$ ), <code>T2 densifyUntilIteration</code> $\leq 14\%$ <code>T1</code>
Redukovat floatery v dronových letech	<code>T43 frustumCullEnabled</code> on, <code>T20 skyMaskingEnabled</code> on, <code>T45 skyDomeEnabled</code> on
Krásná obloha ve venkovních scénách	<code>T45 skyDomeEnabled</code> on, <code>T47 skyDomeRadiusMultiplier</code> 30–60
Menší export soubor	Strategie <code>.mcmc</code> ( <code>T61</code> ), <code>T56 postTrainingCompactification</code> on, <code>T62 mcmcMaxGaussians</code> $\leq 200K$
Rychlejší trénink	<code>T22 trainingRenderScale</code> 0.5, <code>T1 maxIterations</code> na polovinu — ale ne obojí!
Lepší lesklá místa	<code>T21 shDegreeUpgradeIterations</code> s [2K, 5K, 8K] (žádný early-front-load), MCMC + 200K iter
Udržet Mac responzivní	<code>T25 throttleDelayMs</code> 5–10 (stojí ~15 % tréninkového času)
Live náhled častěji	<code>T59 livePreviewInterval</code> dolů na 10–20
Měkčí přechody ve stínech	<code>T17 ssimWeight</code> trochu vysoko (0.15–0.25), ale ne nad 0.3
Udržet interiéry kompaktní	<code>P10 Indoor</code> předvolba (, <code>T72 = 1.76</code> )

## Nebezpečná pole

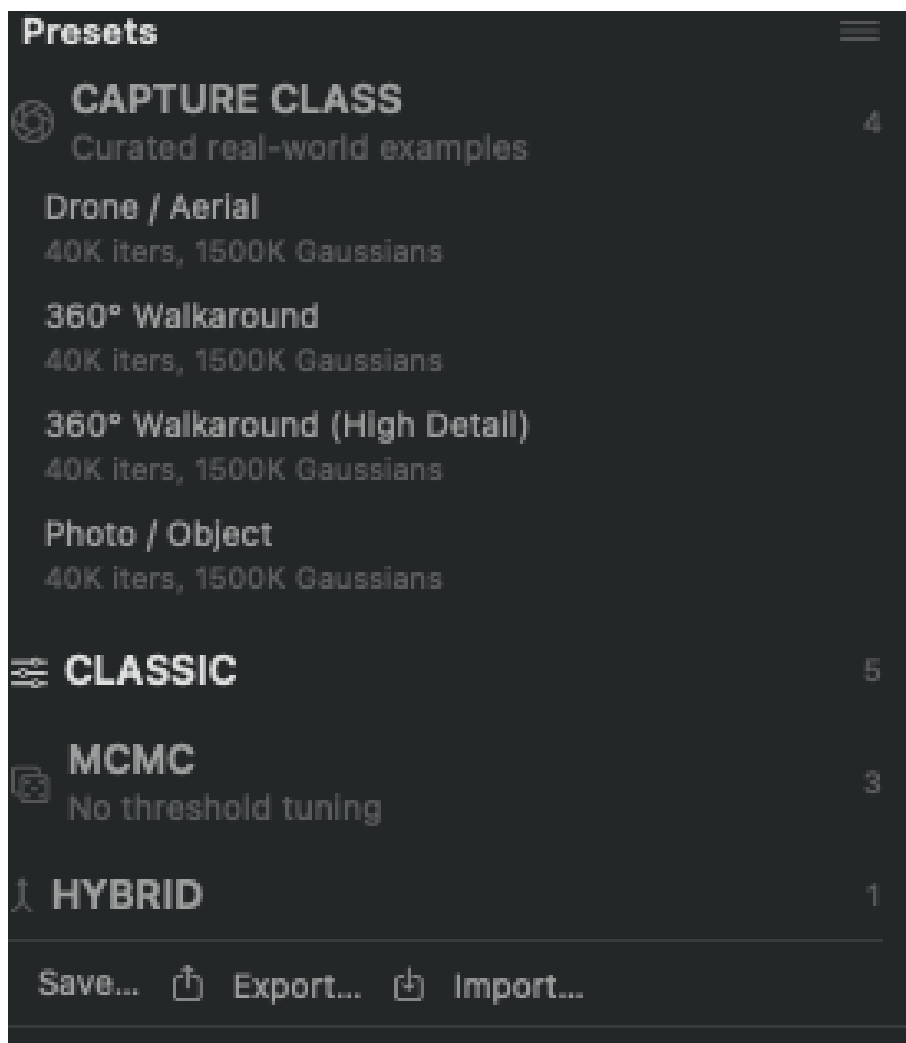
Tato pole mohou při špatné konfiguraci vést k OOM, pádu aplikace, mass extinction Gaussianů nebo nepoužitelným benchmark datům. Zacházet s opatrností:

- T11 `densifyGradThreshold` — halvení může vytvořit 2–4× více Gaussianů, což rychle vyhodí GPU paměť. Také pamatovat: musí odpovídat T22 `trainingRenderScale` ( $1.0\times \rightarrow 1e-6$ ,  $0.5\times \rightarrow 2e-6$ ,  $0.25\times \rightarrow 4e-6$ ).
- T72 `mcmcCapMultiplier` — u velkých scén s  $> 200$  K SfM init bodů a multiplifikátorem  $> 5$  vzniká resolved cap milionů Gaussianů. Na 36 GB RAM Maců OOM možné. Outdoor předvolba 5.32 funguje jen proto, že Mip-NeRF 360 Bicycle má 156 K init bodů  $\rightarrow$  830 K cap.
- T39 `testViewIndices` — manuální nastavení může učinit benchmark nepoužitelným (všechny indexy  $> N \rightarrow$  žádné holdouts). Nech to `--benchmark` flagu nastavovat.
- T64 `mcmcOpacityRegWeight` a T65 `mcmcScaleRegWeight` — v 1.4.3 betě nastaveno na 0.01, což vedlo k mass extinction (460 K  $\rightarrow$  5 Gaussianů v jedné iteraci). Od 1.4.4 fixováno na 0.0, ale ruční zvýšení může problém reprodukovat.
- T15 `opacityResetInterval` — pokud ne 100 000+ (efektivně off) a trénink je kratší než 10 000 iterací, reset zničí konvergenci. `.preview` to má proto na 100 000 přes `maxIterations = 5 000`.
- T54/T55 `densifyPhase2*` — Two-Phase densifikace v testech zhroucena na 0-Gaussians-cascade. Nech obě na 0.
- T74 `useMipSplatting` — Q1.5 closed-no-win 2026-05-25, může na některých outdoorových scénách dokonce zhoršit PSNR. Default off, opt-in jen pro výzkum.

Pokud pole stojí na tomto seznamu a chceš ho měnit, udělej nejdřív zálohu své aktuální předvolby (export jako JSON) a uvaž, zda můžeš výsledek reprodukovatelně měřit — jinak hned nevíš, zda jsi zlepšení nebo zhoršení způsobil.

## KAPITOLA

## Kapitola 7 — Vestavěné kvalitativní předvolby



Obrázek 27: Sekce předvoleb se všemi čtyřmi rozbalenými skupinami — CAPTURE CLASS (4 předvolby: Drone/Aerial, 360° Walkaround, 360° Walkaround (High Detail), Photo/Object), CLASSIC (5 předvoleb: Quick/Preview/Balanced/Quality/Ultra Detail), MCMC (3 předvolby, poznámka „No threshold tuning„), HYBRID (1 předvolba: Balanced (Hybrid))

**CO JE NA OBRÁZKU** Sekce předvoleb v Inspektoru, všechny čtyři skupiny rozbalené. CAPTURE CLASS se čtyřmi kurátorovanými reálnými předvolbami (Drone / Aerial, 360° Walkaround, 360° Walkaround (High Detail), Photo / Object) — to je primární skupina a v Režimu pro začátečníky jediná viditelná. CLASSIC s Quick (1K iterací), Preview (5K ite-

rací, aktivní výběr s modrým zatržením), Balanced (20K iterací), Quality (35K iterací) a Ultra Detail (35K iterací). MCMC s podnadpisem „No threshold tuning„ — MCMC nepotřebuje densify-until threshold: Preview (60K iterací, 100K Gaussianů), Balanced (120K, 150K), Quality (200K, 150K). HYBRID s předvolbou Balanced (Hybrid) (20K iterací, 150K Gaussianů). Řádek akcí v patičce: Save..., Export..., Import...

Předvolba je připravená konfigurace pro trénink. RadianceKit dodává třináct vestavěných předvoleb ve čtyřech skupinách: čtyři **Capture-Class** předvolby (P9–P12) — kurátorované recepty validované okem na reálném komunitním materiálu pro skutečné druhy záběrů (dron, 360° procházka, foto-objekt) a od v1.6 primární osa —, pět Classic předvoleb (P1–P5: Quick/Preview/Balanced/Quality/Ultra Detail), tři MCMC předvolby (P6–P8) a jednu hybridní předvolbu (P13), která kombinuje strategie Classic a MCMC. Dřívější „Scene-Class„ předvolby (Render/3D, Outdoor, Indoor, ve fázi Q7 akademicky laděné proti scénám z Mip-NeRF-360 a NeRF-Blender) byly ve v1.6 staženy jako viditelná skupina — okem na reálném materiálu validovaná Capture-Class je nyní primární osa; Q7-laděné konfigurace zůstávají zachovány jen interně. Vybíráš předvolby v postranním panelu v sekci **Předvolby** nebo v Režimu pro začátečníky při importu. Tlačítka + otevírají dialogy pro vytvoření vlastních předvoleb vedle nich — třináct vestavěných nelze smazat, ale lze je duplikovat.

V expertním zobrazení se předvolby objevují seskupené podle druhu záběru a strategie (Capture Class / Classic / MCMC / Hybridní). Kliknutí na položku zapíše uloženou konfiguraci tréninku do aktuálního stavu. Není to snapshot — pokud poté otáčíš posuvníky, mění se stav, ale samotná předvolba zůstává nezměněna; barevný indikátor pak ukáže „modified„.

Která předvolba je kdy ta správná, závisí především na typu scény a hardwaru. Tři tabulkové přehledy na konci kapitoly to shrnují.

## I P1 — Quick



Inspektor → Sekce předvoleb → Skupina „Classic„ → Položka „Quick“. UUID sufix `...001`.



Diagnostická předvolba s 1 000 iteracemi, klasickou (adaptivní) strategií densifikace a trénovacím škálováním rozlišení 0.25x (vstupní obrázek se před tréninkem zmenší na 25 %). Není určen pro dodání scény, ale pro rychlé zjištění, zda setup (pozice kamer, mrak bodů, série obrázků) vůbec vykazuje smysluplný pohyb v hodnotách ztráty. Na M3 Ultra typicky pod 30 sekund na 50–200 obrázcích. Malé rozlišení zastírá skutečnou kvalitu obrazu, ale udržuje paměťovou náročnost a render velmi nízké. Automaticky se vybírá jako výchozí při prvním spuštění, pokud má systém méně než 10 GB RAM.

### JEDNODUŠE ŘEČENO

Rychlý funkční test. Obrázky dovnitř, krátká půlminuta čekání, podívat se, zda se objeví hrubá silueta scény. Pokud obraz v prohlížeči vypadá jako rozmazaná skvrna — v pořádku, tak to má být. Pokud naopak vidíš jen tmavé body nebo zcela zkraslený tvar, jsou pravděpodobně špatné pozice kamer (viz Kapitola 9). Pro ukazatelný výsledek pak potřebuješ alespoň P2 nebo P3.

## | P2 — Preview (Classic)



KDE

Inspektor → Sekce předvoleb → Skupina „Classic“, → Položka „Preview“. UUID sufix `...002`.



TECHNICKY

5 000 iterací classic densifikace, škálování rozlišení 0.5×, dvojnásobné learning rates oproti standardu. Densifikace (klonování + štěpení) je aktivní přes prvních 2 500 iterací, poté už jen pruning. Výchozí předvolba pro systémy s  $\geq 10$  GB RAM. Na M3 Ultra typicky 90 sekund až 3 minuty pro scénu s 200 obrázky. Dodává použitelný dojem geometrie a pozice kamer, ale textury jsou viditelně rozostřené — škálování renderu 0.5× nelze později obejít opětovným tréninkem s P3 nebo P4, protože learning rates jsou kalibrovány na poloviční rozlišení.

### JEDNODUŠE ŘEČENO

Standard pro „jen rychle si to prohlédnout“. Pokud jsi právě importoval nové obrázky a chceš vidět, zda je scéna vůbec rekonstruovatelná, je to správná úroveň. Asi 2–3 minuty čekání, poté můžeš ve 3D prohlížeči otáčet a posoudit, zda má smysl investovat do dalších tréninkových běhů. Až když výsledek náhledu už vypadá dobře, vyplatí se Balanced nebo Quality.

## | P3 — Balanced (Classic)



KDE

Inspektor → Sekce předvoleb → Skupina „Classic“, → Položka „Balanced“. UUID sufix `...005`.



TECHNICKY

20 000 iterací classic densifikace v plném rozlišení obrazu. Densifikace běží přes prvních 15 000 iterací, od iter 3 000 s densify intervalem 100. Empiricky „sweet spot“, z dokumentovaných tréninkových relací: při classic densifikaci na Horse Full a Truck se L1 loss stabilizuje mezi iter 18 000 a 22 000, delší trénink už pod úroveň Quality (P4) nepřináší signifikantní zlepšení. Na M3 Ultra typicky 30–60 sekund na 200 obrázcích, 5–8 minut na 1 000+ obrázcích.

### JEDNODUŠE ŘEČENO

„Dobrý kompromis“. Většina scén už s tímto vypadá dobře, aniž bys musel hodinu čekat. Pokud chceš někde ukázat finální výsledek (sociální sítě, web, klientská demo), často to stačí. Až když chceš do splat modelu nazožovat nebo potřebuješ detaily povrchové textury, vyplatí se skok na P4 Quality nebo P7 MCMC.

## I P4 — Quality (Classic)



KDE

Inspektor → Sekce předvoleb → Skupina „Classic„ → Položka „Quality“. UUID sufix `...003`.



TECHNICKY

35 000 iterací classic densifikace s V546-„Opacity Decay„ (HTGS, Eurographics 2025): po každém cyklu densifikace se opacity všech existujících Gaussianů násobí faktorem  $< 1.0$ , což při pruningu spolehlivě odstraňuje neaktivní Gaussiany a tím dosahuje při identickém počtu iterací o 14 % lepší L1 loss než klasický běh s 35 000 iter. SSIM loss je aktivován (`ssimWeight=0.05`). Na M3 Ultra typicky 2–4 minuty na 200 obrázcích. Na NeRF-Blender (Lego, Chair, Drums) dodává finální L1  $\approx 0.023$  — nejlepší classic varianta v 560+ dokumentovaných experimentech. Pozor: potřebuje ~3–5 GB GPU paměti; na systémech s 8 GB je P3 bezpečná volba.

### JEDNODUŠE ŘEČENO

Nejlepší klasická varianta. Dodává ostrou texturu a jemnou geometrii, zvláště u záběrů objektů (socha, židle, váza). U velkých outdoorových scén nebo místností naopak skoro nepoznáš rozdíl proti Balanced — tam se vyplatí přechod na MCMC předvolbu (P6–P8) nebo Capture-Class předvolbu (P9–P12) více než skok z P3 na P4. Kdo chce absolutní maximum rodiny Classic, vezme P5 Ultra Detail.

## I P5 — Ultra Detail (Classic)



KDE

Inspektor → Sekce předvoleb → Skupina „Classic„ → Položka „Ultra Detail“. UUID sufix `...008`.



TECHNICKY

Zhruba 35 000 iterací classic densifikace — vítěz held-out běhu kvalitativní matice (2026-06-10). Na všech třech testovaných scénách Mip-NeRF-360 poráží Ultra Detail vestavěnou MCMC předvolbu „Quality„ (P8) při srovnatelném wall-clock čase o zhruba +0.94 dB PSNR. Je to tedy nejsilnější kvalitativní předvolba skupiny Classic a nejostřejší classic varianta, kterou RadianceKit dodává. Na M3 Ultra typicky ve stejném časovém rámci jako P4 Quality (2–5 minut na 200 obrázcích), potřebuje ale o něco více GPU paměti; na systémech s 8 GB zůstává P3 bezpečnou volbou.

### JEDNODUŠE ŘEČENO

Nejostřejší classic stupeň a held-out vítěz našich kvalitativních testů: na reálných scénách zhruba o decibel lepší než MCMC varianta „Quality„ — při podobné době čekání. Pokud chceš maximální věrnost detailu s osvědčenou klasickou densifikací a máš dost GPU paměti, je to první volba. Pokud paměť nestačí nebo potřebuješ co nejmenší exportní soubor, zůstaň u P4 Quality nebo MCMC předvolby.

## | P6 — Preview (MCMC)



Inspektor → Sekce předvoleb → Skupina „MCMC„ → Položka „Preview“. UUID sufix ...006 .



60 000 iterací MCMC densifikace (3DGS-MCMC, NeurIPS 2024) při capu 100 000 Gaussianů. MCMC nahrazuje heuristickou klon/split logiku Markov-Chain-Monte-Carlo relokací: mrtvé Gaussiany se znovu umísťují přes sigmoidem vážené sampling hloubky, což dává řízený a reprodukovatelný počet Gaussianů. Cap tvrdě omezuje maximální počet na 100K — to šetří paměť i render time, ale stojí to detail. Na M3 Ultra typicky 5–8 minut na 200 obrázcích. Vhodné jako „MCMC funkční test„ — pomáhá posoudit, zda by přechod z Classic na MCMC dával smysl, než investuješ více času do P7 nebo P8.

### JEDNODUŠE ŘEČENO

Jako P2 Preview, ale s novějším MCMC postupem. Často dodává trochu kompaktnější, rovnoměrněji rozložené splat mraky než classic varianta. Pro první posouzení scény obvykle stačí 5–8 minut. Pokud se ti výsledek náhledu líbí, dalším krokem je P7 (Balanced) nebo přímo P8 (Quality MCMC).

## | P7 — Balanced (MCMC)



Inspektor → Sekce předvoleb → Skupina „MCMC„ → Položka „Balanced“. UUID sufix ...007 .



120 000 iterací MCMC při capu 150 000 Gaussianů. Střední MCMC stupeň — téměř finální počet Gaussianů jako u P8 Quality, ale jen 60 % iterací. Empiricky leží L1 loss v dokumentovaných tréninkových relacích na 0.026–0.028 na Horse Full, vůči P8 s 0.0246 — tj. asi o 7 % výše, ale poloviční čekání. Na M3 Ultra typicky 8–15 minut na 200 obrázcích. Používá metodu, která škáluje efektivní Gaussian cap podle hustoty bodů vstupního SfM mraku (viz T75 v kapitole 6).

### JEDNODUŠE ŘEČENO

MCMC se slušnou hloubkou detailu, ale bez dlouhého plného běhu P8. Pro většinu scén to stačí, zvláště pokud chceš vměstnat MCMC běh do polední pauzy. Pokud je paměť na hraně (např. na M procesorech s pouze 16 GB), zůstaň zde — P8 potřebuje více GPU paměti.

## I P8 — Quality (MCMC)



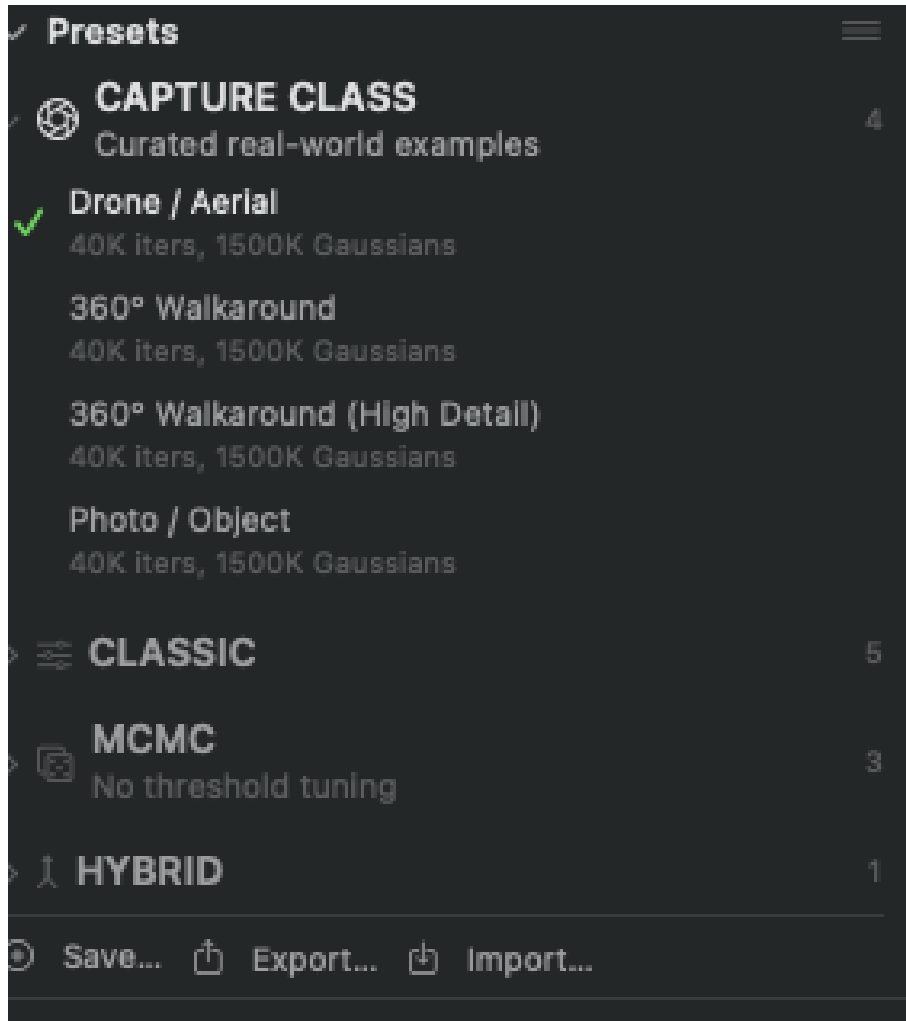
Inspektor → Sekce předvoleb → Skupina „MCMC„ → Položka „Quality“. UUID sufix `...004` .



200 000 iterací MCMC při capu 150 000 Gaussianů, SSIM loss 0.05, MCMC noise decay přes 80 % iterací. Best single run L1 v 560+ experimentech: 0.0238 na Horse Full, průměrováno přes 3 trialy 0.0246 (vůči P4 0.0230 na stejné scéně). MCMC přitom dodává 71 % méně Gaussianů (150K vs ~524K) — rozhodující, když chceš výsledek dodávat na web, protože menší mrak produkuje výrazně menší exportní soubory. Tréninkový čas na M3 Ultra typicky 20–35 minut na 200 obrázcích; na setech s 1 000+ obrázky spíše 1–2 hodiny. Nejlepší volba, pokud chceš maximální kvalitu obrazu při minimální výsledné velikosti.

### JEDNODUŠE ŘEČENO

Nejlepší MCMC varianta. Dodává velmi čisté, kompaktní splat mraky — ideální, pokud chceš výsledek později vložit jako web 3D prohlížeč nebo poslat jako soubor (soubor je menší než u P4 Quality při srovnatelné optické kvalitě). Potřebuje ale trpělivost — u velkých záběrů přes hodinu čekání. Plánuj to spíše jako „přes noc„ běh.



Obrázek 28: Skupina CAPTURE CLASS rozbalená se všemi čtyřmi kurátorovanými reálnými předvolbami — Drone / Aerial (MCMC, 40K iter, Cap 1,5 M), 360° Walkaround (MCMC, 40K, Cap 1,5 M), 360° Walkaround (High Detail) (Hybridní, 40K, Cap 1,5 M, opt-in) a Photo / Object (Hybridní, 40K, Cap 1,5 M). Tato skupina stojí úplně nahoře a v Režimu pro začátečníky je jediná viditelná.

**CO JE NA OBRÁZKU** Inspektor s rozbalenou skupinou CAPTURE CLASS — primární skupina předvoleb od v1.6, v Režimu pro začátečníky jediná zobrazená. Každá položka je okem na reálném komunitním materiálu validovaný recept pro konkrétní druh záběru (dron, 360° procházka, foto-objekt), nikoli hodnota optimalizovaná proti akademickému testovacímu setu. Výběr kliknutím zapíše uloženou konfiguraci tréninku do aktuálního stavu.

## | P9 — Drone / Aerial



Inspektor → Sekce předvoleb → Skupina „Capture Class„ → Položka „Drone / Aerial“. UUID sufix ...010 .



Capture-Class předvolba pro letecké a dronové orbity budov a krajiny. MCMC densifier, 40 000 iterací, cap 1,5 mil. Gaussianů, SSIM loss 0.5 plus edge-aware term 0.1. Rozhodující je penalizace anizotropie s váhou 0.003 při prahu poměru 6 — „zabiják špaget„ proti typicky jehlovitým artefaktům, které dronové záběry produkují. Validováno na skutečném DJI 4K dronovém letu nad viaduktem Pensford (prověřeno okem, nejen metricky).

### JEDNODUŠE ŘEČENO

Pro záběry ze vzduchu — dronové lety kolem budovy, nad krajinou, podél fasády. Silná penalizace anizotropie odklízí jehlovité, špagetovité artefakty, které dronový materiál rád produkuje. Pokud je tvůj materiál snímán ze země, hodí se spíše Photo / Object nebo Classic předvolba.

## | P10 — 360° Walkaround



Inspektor → Sekce předvoleb → Skupina „Capture Class„ → Položka „360° Walkaround“. UUID sufix ...011 .



Capture-Class předvolba pro 360° walkaround videa. MCMC densifier, 40 000 iterací, cap 1,5 mil. Gaussianů, SSIM loss 0.5 plus edge-aware term 0.1, mírná penalizace anizotropie (váha 0.001 při prahu poměru 15). Masky osob a oblohy je aktivní. Předvolba očekává 360° equirect video, které se interně reprojektuje na zhruba 90° široké perspektivní výřezy, než trénink začne. Validováno na 8K 360° procházkách se selfie tyčí (scéna Monument, prověřeno okem).

### JEDNODUŠE ŘEČENO

Pro 360° walkaround videa — procházíš s 360° kamerou nebo selfie tyčí místností nebo kolem objektu. RadianceKit rozloží kulové panorama sám do normálních úhlů pohledu a maskuje kolemjdoucí a oblohu. Pro maximální ostrost na stejném materiálu vyzkoušej navíc High-Detail variantu (P11).

## I P11 — 360° Walkaround (High Detail)



KDE

Inspektor → Sekce předvoleb → Skupina „Capture Class„ → Položka „360° Walkaround (High Detail)“. UUID sufix `...013` (opt-in).



TECHNICKY

Opt-in Capture-Class předvolba pro 360° walkaround videa s maximálním detailem. Hybridní densifier (klasické abs-gradient klonování/štěpení + MCMC šum + relokace), 40 000 iterací, cap 1,5 mil. Gaussianů, penalizace anizotropie 0.0015 při prahu poměru 15, SSIM loss 0.2 a edge-aware term 0 — zamčený „r50“ screen-split recept. Na 360° materiálu poráží standardní MCMC předvolbu „360° Walkaround„ (P10) v PSNR, LPIPS i viditelném konfetti, a to zhruba s třetinou počtu splatů. Stojí záměrně opt-in *vedle* standardní 360 předvolby, dokud nebude validována na více scénách.

### JEDNODUŠE ŘEČENO

Ostřejší alternativa ke standardní 360 předvolbě (P10): více detailu, méně konfetti, výrazně menší soubor. Stojí záměrně vedle, místo aby ji nahradila — zatím potvrzena na hrstce scén. Pokud je tvoje 360° procházka natočená čistě, vyzkoušej tuto předvolbu jako první a porovnej výsledek s P10.

## I P12 — Photo / Object



KDE

Inspektor → Sekce předvoleb → Skupina „Capture Class„ → Položka „Photo / Object“. UUID sufix `...012`.



TECHNICKY

Capture-Class předvolba pro orbity objektů z ostrých jednotlivých fotografií (žádné video). Hybridní t1 densifier (s relokací), 40 000 iterací, cap 1,5 mil. Gaussianů, SSIM loss 0.5 plus edge-aware term 0.1, mírná penalizace anizotropie (váha 0.001 při prahu poměru 15), opacity decay 0.9995 každých 50 iterací, **žádné** maskování. Validováno na 163 vysoko-rozlišovacích 41 MP fotografiích kostry (prověřeno okem). Málo pohledů (do zhruba 600) přitom zůstává pod prahem hybridního collapse.

### JEDNODUŠE ŘEČENO

Pro záběry objektů z ostrých jednotlivých fotografií — obcházíš sochu, model, produkt s kamerou a děláš fotky místo videa. Žádné maskování, protože ostré fotky mají obvykle čisté pozadí. Pro video zdroje vezmi místo toho 360° nebo Drone předvolbu.

## I P13 — Vyvážený (Hybridní)



Inspektor → Sekce předvoleb → Skupina „Hybridní“ → Položka „Vyvážený (Hybridní)“. UUID sufix `...009`.



20 000 iterací s hybridní strategií densifikace: klasický gradientem řízený clone/split umísťuje kapacitu tam, kde ji loss potřebuje, MCMC SGLD šum dál exploruje a mrtvé Gaussiany se relokují, místo aby se ztrácely pruningem. Opacity decay (V546) nahrazuje opacity resety; penalizace anizotropie (váha 0.001, práh poměru 15) drží jehlovité splaty na uzdě. Gaussian cap se škáluje se scénou (základ 150K, scene-aware x3.0). Validováno na pěti scénách proti čistému MCMC při stejném rozpočtu: v průměru +0.45 dB PSNR při o 20–30 % nižším počtu Gaussianů (stonehenge +1.23, family +0.82, garden +0.47 dB). Na M3 Ultra typicky 5–10 minut na 200 obrázcích.

### JEDNODUŠE ŘEČENO

Silná první volba pro finální výsledek: ostřejší detail než MCMC předvolby při podobně kompaktním souboru, za zlomek tréninkového času P8. Pokud máš čas jen na jeden kvalitní běh a žádáš z capture tříd jasně nepasuje, začni tady. Klasické předvolby zůstávají lepší pro rychlé testy a Capture-Class předvolby (P9–P12) jsou první volbou, když tvoje scéna jasně odpovídá jednomu z těchto druhů záběrů.

## Kdy kterou předvolbu?

Scénář	První test	Hlavní běh
Funkční test nových obrázků, < 30 s	<b>P1 Quick</b>	—
Orbit objektu z ostrých jednotlivých fotografií	P2 Preview	<b>P12 Photo / Object</b>
Sken jednotlivého objektu (video), < 500 fotografií	P2 Preview	<b>P4 Quality</b> nebo <b>P8 Quality MCMC</b>
360° walkaround video	P6 Preview MCMC	<b>P10 360° Walkaround</b> (ostře: <b>P11 High Detail</b> )
Letecký/dronový orbit, krajina	P6 Preview MCMC	<b>P9 Drone / Aerial</b>
Web dodávka (malé, kompaktní)	P2	<b>P8 Quality MCMC</b> (nejmenší soubor při plné kvalitě)
Ostrý detail za krátký čas, kompaktní export	P2 nebo P6	<b>P13 Vyvážený (Hybridní)</b>
Maximální věrnost detailu, strategie Classic	P3 nebo P6	<b>P5 Ultra Detail</b>
Tisk, marketing, plný detail	P3 nebo P6	<b>P4 Quality (Classic)</b> nebo <b>P5 Ultra Detail</b>

## Rychlé srovnání

Před- vol- ba	Strate- gie	Iter	Max Gs	Render škála	Typický čas (200 obr., M3 UI- tra)	Q-sweep
P1 Quick	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 Qua- lity	Classic	35 000	∞	1.0x	2–4 min	V546 HTGS
P5 UI- tra De- tail	Classic	~35 000	∞	1.0x	2–5 min	Mat- rix 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 Qua- lity 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	Oko / Via- dukt
P10 360° Wal- karound	MCMC	40 000	1.5 M	1.0x	10–25 min	Oko / Mo- nu- ment
P11 360° Wal- karound (High De- tail)	Hyb- ridní	40 000	1.5 M	1.0x	10–25 min	Oko (opt- -in)
P12 Pho- to / Ob- ject	Hyb- ridní	40 000	1.5 M	1.0x	10–25 min	Oko / Kos- tra
P13 Vy- vá- že- ný (Hy-	Hyb- ridní	20 000	150 K	1.0x	5–10 min	Mat- rix Δ+0.45 dB

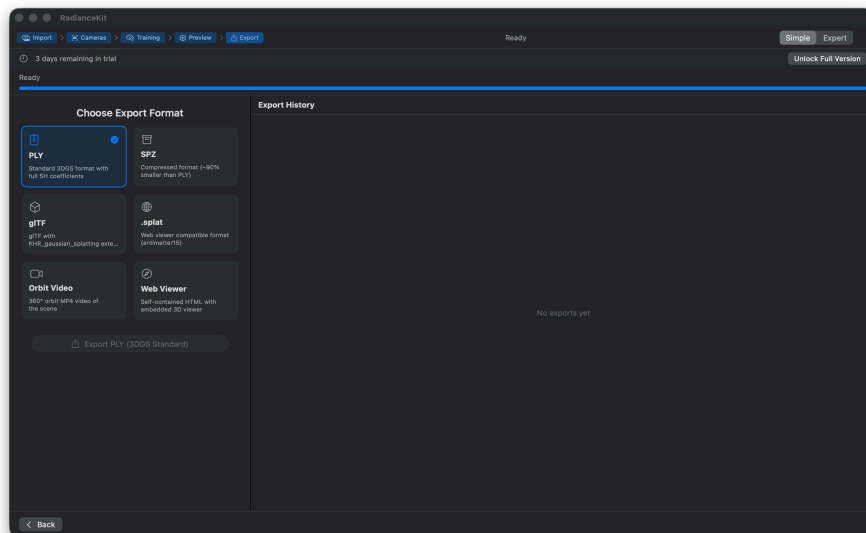
## Vlastní předvolby

Pomocí tlačítka **Save...** v sekci předvoleb (I1 v Kapitole 2) ukládáš aktuální konfiguraci tréninku jako vlastní předvolbu. Vlastní předvolby nejsou „Built-in“, a lze je přejmenovat, exportovat (jako JSON), sdílet přes drag-and-drop, duplikovat a mazat. Třináct vestavěných předvoleb P1–P13 zůstává tlačítkem smazání nedotčeno.

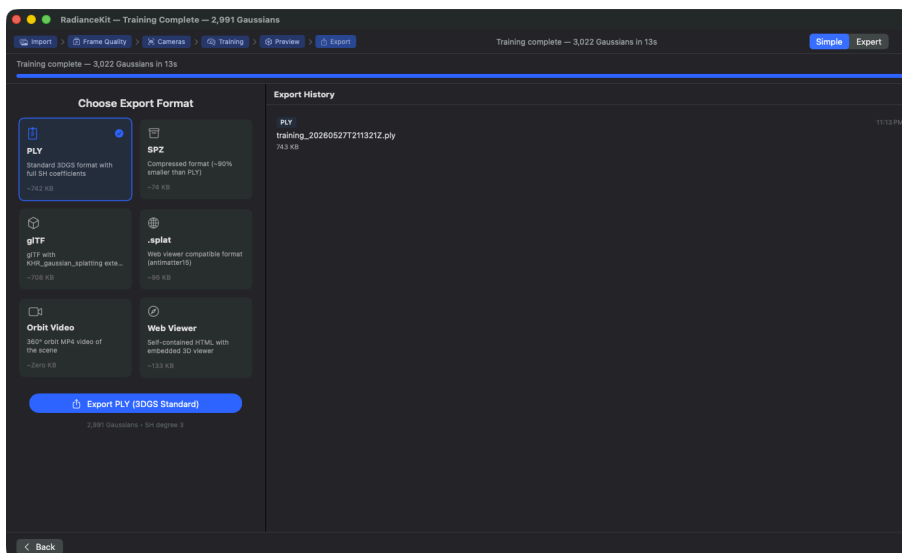
**Pravidlo:** Pokud na předvolbě měníš něco, co budeš potřebovat častěji — Sky-Dome zapnutý, vyšší SSIM váha pro určitou třídu scén, odlišné počty iterací — ulož variantu jako vlastní předvolbu. Tak hned při dalším běhu víš, že jde o konfiguraci odchylnou od standardu.

## KAPITOLA

# Kapitola 8 — Formáty exportu



Obrázek 29: Výběr formátu exportu v Režimu pro začátečníky — šest formátových karet



Obrázek 30: Mřížka formátů exportu živě po 5K iter tréningu — všech šest karet s dynamickým výpočtem velikosti (PLY 742 KB vybráno, SPZ 74 KB, glTF 708 KB, .splat 96 KB, Orbit Video ~Zero KB, Web Viewer 133 KB), historie exportu vpravo s již uloženým PLY

**Co je na obrázku (2 991 Gaussianů, SH stupeň 3, syntetické testovací sety):** Údaje o velikosti pod každou kartou formátu se počítají živě z aktuálního počtu Gaussianů a režie formátu — nikoli napevno. Z 2 991 Gaussianů (SH stupeň 3) tak

vzniká 742 KB PLY, 74 KB SPZ (faktor  $\sim 10\times$  menší díky kvantizaci), 708 KB glTF (s rozšířením `KHR_gaussian_splatting`, tedy téměř ekvivalentní PLY), 96 KB `.splat` (komprimovaný 24-bajtový formát na Gaussian). Orbit Video ukazuje „~Zero KB“, protože velikost je známa až po MP4 enkódování. Web Viewer (133 KB) balí samostatný HTML soubor s vloženým WebGL prohlížečem a komprimovanými `splat` daty — větší než čistý `.splat` kvůli režii prohlížeče. Historie exportu vpravo listuje již dokončený PLY export („`training_20260527T211321Z.ply`, 743 KB, 23:13“) s `pillou` formátu a akcí `reveal` ve Finderu.

Dokončený trénink dodává Gaussian Cloud — kolekci několika set tisíc až milionů 3D Gaussovských rozdělí, která dohromady rekonstruuji scénu. RadianceKit zná deset cest, jak tento cloud zapsat na disk. Šest z nich jsou čisté 3D datové formáty (PLY, Compressed PLY, SPZ, SOG, glTF, `.splat`), jeden balí cloud spolu s hotovým HTML prohlížečem (Web Viewer), jeden renderuje MP4 soubor z orbitální jízdy kamery (Orbit Video), a dva neexportují žádný Gaussovský obsah, pouze výsledek SfM (pozice kamer a hrubý mrak bodů) pro znovupoužití v jiných tréninkových pipeline (`transforms.json` + COLMAP workspace).

Který formát kdy je správný, závisí na cíli. Pro archivaci plných dat bez ztráty kvality bere se PLY. Pro web prohlížeč na vlastní stránce obvykle stačí `.splat` nebo vestavěný Web Viewer. Pokud má být soubor minimální, vyplatí se SPZ nebo SOG. Pro znovupoužití výsledku SfM v Nerfstudio, Postshot nebo Brush jsou `transforms.json` a COLMAP workspace správné cesty.

Všechny funkce exportu leží v menu „Export“, i v Režimu pro začátečníky na poslední wizardové úrovni. Většina formátů je plně sandbox-kompatibilní a funguje v App Store verzi. Pouze SOG vyžaduje externí binárku ( `cwebp` ), která ve App Store buildu nemusí být přítomná — detaily viz E4.

## I E1 — PLY (.ply)



Řádek nabídek → Export → 3D Formats → Export PLY... (⌘E). Režim pro začátečníky: wizardový krok Export → karta formátu „PLY„. **Velikost:** typicky 100 % (referenční hodnota). **Kompatibilní s:** SuperSplat, PolyCam, všechny 3DGS prohlížeče.



PLY je kanonický formát pro ukládání 3D Gaussian Splatting. RadianceKit zapisuje binární little-endian soubor se standardizovaným 3DGS property layoutem: na Gaussian tříkomponentní pozice, tři vždy nulové normály, tři DC-SH koeficienty ( `f_dc_0..2` ) pro základní RGB barvu, následně až 45 dalších SH koeficientů ( `f_rest_0..44` ) v transponovaném channel-major uspořádání definovaném Kerblovým paperem 2023 (nejprve všechny koeficienty R kanálu, pak všechny G, pak všechny B), následované logit-opacity (raw pre-sigmoid hodnoty), třemi log-space škálami a wxyz quaternion rotací. Maximální exportovaný SH stupeň je clampován na minimum uživatelského přání a skutečně naučeného stupně; default je 3 (45 rest koeficientů). Před zápisem se velikost payloadu počítá v 64-bitových integerech, aby se zachytil overflow u extrémně velkých cloudů. Soubor se zapisuje atomicky, což u velkých cloudů krátce zabírá dvojnásobné místo na disku.

### JEDNODUŠE ŘEČENO

Toto je „originální soubor„. Největší soubor, nejvyšší kompatibilita, žádné ztráty. Pokud nevíš, který formát zvolit, vezmi PLY — otevře se v téměř každém 3DGS nástroji. Pro 1 milion Gaussianů to je podle SH stupně mezi 200 a 800 MB. Pokud je soubor moc velký, podívej se na E2 (komprimované PLY) nebo E3 (SPZ).

## I E2 — Compressed PLY (.ply)



Řádek nabídek → Export → 3D Formats → Export Compressed PLY.... Režim pro začátečníky: karta formátu „Compressed PLY„. **Velikost:** cca 10–20 % oproti PLY (5- až 10násobná komprese). **Kompatibilní s:** SuperSplat, PlayCanvas engine, web prohlížeče.



PlayCanvas varianta PLY formátu s chunked kvantizací. Gaussiany jsou seskupené po 256. Pro každý chunk se v hlavičce ukládají oddělené min/max meze pro pozici, škálu a barvu; jednotlivé Gaussiany referencují své hodnoty relativně k těmto mezím a jsou komprimovány na 32 bitů: pozice a škála s 11-10-11 bit packingem, rotace jako 2-10-10-10 bit „smallest-three“, quaternion, barva jako 8-8-8-8 RGBA. Vyšší SH koeficienty se kvantizují pouze 8 bity na komponentu ( `shCoeffCount * 3` uchar na Gaussian). Samotný formát je stále PLY s ASCII hlavičkou a tedy v principu validovatelný PLY nástroji, ale vertex properties jsou deklarovány jako `uint` pole. SH stupeň je defaultně 0 (žádné rest koeficienty), aby se maximalizovala komprese — vyšší SH stupeň lze explicitně zvolit.

### JEDNODUŠE ŘEČENO

Úsporná PLY varianta. Identická engine kompatibilita jako normální PLY, ale 5- až 10krát menší. SuperSplat a PlayCanvas to čtou nativně. Pro nasazení na web téměř vždy lepší než normální PLY. Ztráta kvality kvantizací je vizuálně obvykle neznatelná, dokud scéna neobsahuje extrémně vysokofrekvenční detaily.

## | E3 — SPZ (.spz)



Řádek nabídek → Export → 3D Formats → Export SPZ... Režim pro začátečníky: karta formátu „SPZ“.

**Velikost:** cca 10 % oproti PLY (o 90 % menší).

**Kompatibilní s:** Niantic Scaniverse, Niantic Spatial Fields, MetalSplatler.



Niantic SPZ-v2 formát. Pozice jsou packované jako 24-bit fixed-point (to dává cca 0,25 mm rozlišení), škály jako 8-bit kvantizace v log prostoru, rotace jako 8-bit smallest-three (v2 se ukládá pouze xyz, w se v dekodéru odvozuje z normy quaternionu), opacity jako sigmoidované 8-bit hodnoty. DC-SH se ukládá s SPZ-specifickou pack formulí ( $dc\_raw * 0.15 * 255 + 0.5 * 255$ ), vyšší SH bandy s 5 bity (band 1) resp. 4 bity (band 2-3) na koeficient. Celý zapakovaný binární blob se následně komprimuje standardním gzipem (RFC 1952), což dává gzipped container formát s magickými bajty `1f 8b`. RadianceKit volá pro to systémový `gzip`, protože vestavěné Apple zlib API vytváří proprietární Apple framing, který by nebyl kompatibilní s SPZ readery v Spatial Fields nebo MetalSplatleru. Systémový `gzip` zůstává spawnovatelný i v rámci macOS sandboxu.

### JEDNODUŠE ŘEČENO

Nejmenší standardní soubor. Pokud znáš Niantic Scaniverse — to je formát, který aplikace používá. Velmi malý, velmi vstřícný k načítání pro mobilní aplikace. V Niantic vlastním cloud prohlížeči (Spatial Fields) přímo použitelné. Asi o 90 % menší než PLY se stejnými daty, přitom pro většinu scén opticky téměř nerozeznatelný.

## I E4 — SOG (.sog)



KDE

Řádek nabídek → Export → 3D Formats → Export SOG.... Režim pro začátečníky: karta formátu „SOG„. **Velikost:** cca 5–6 % oproti PLY (15- až 20násobná komprese — nejmenší možnost). **Kompatibilní s:** PlayCanvas engine, SuperSplat editor.



TECHNICKY

„Spatially Ordered Gaussians„ — PlayCanvas formát, který ukládá cloud GPU-ready v několika lossless WebP obrázcích. Nejprve se všechny Gaussiany prostorově seřadí podle 3D Morton kódu (30-bit Z-order, 10 bitů na osu), což obrázkům poskytuje pozdější cache lokalitu v rendereru. Pak se pozice s symetrickou logaritmickou transformací (pro lepší dynamický rozsah) kvantizují na 16-bit hodnoty a rozdělí do dvou RGBA obrázků ( `means_l.webp` pro spodních 8 bitů, `means_u.webp` pro horních). Rotace jsou kódovány jako `smallest-three` s 3×8 bit plus 2-bit mode v jednom RGBA obrázku (mode v alpha jako `252 + largest` ). Škály a DC-SH jsou kvantizovány codebookem se 256 položkami (percentilově rozdělené přes všechny hodnoty), indexy končí v `scales.webp` a `sh0.webp` . Pět obrázků plus `meta.json` s codebooky a hranicemi se zabalí do ZIP souboru (vlastní enkodér, protože sandbox blokuje systémový `zip` ) a uloží s příponou `.sog` .

**Pozor sandbox:** SOG je jediná formátová volba, která vyžaduje externí binárku. WebP enkodér volá `cwebp` z `/usr/local/bin/cwebp` nebo `/opt/homebrew/bin/cwebp`. Pokud žádná cwebp binárka nenajde, kód spadne na surové PNG enkódování — ale: **PNG fallback v SuperSplat nefunguje.** V App Store verzi vyhodnoť dostupnost podle buildové varianty; ve vývojářské variantě musí být cwebp instalován přes Homebrew ( `brew install webp` ).



JEDNODUŠE ŘEČENO

Nejmenší 3DGS formát vůbec, výrazně menší než SPZ. Ale: potřebuje na Macu nástroj `cwebp` , protože RadianceKit sám neumí všechny formáty obrázků vytvořit. Nainstaluj ho jednou pomocí Homebrew, pak vše běží. Ve verzi App Store případně ne plně funkční — pokud při exportu vychází PNG místo WebP, soubor v SuperSplatu přímo neotevřeš. Kdo pracuje bez Homebrew, vezme místo toho SPZ (E3).

## I E5 — glTF (.glb)



Řádek nabídek → Export → 3D Formats → Export glTF... Režim pro začátečníky: karta formátu „glTF„. **Velikost:** srovnatelná s PLY. **Kompatibilní s:** glTF prohlížeče s rozšířením KHR\_gaussian\_splatting (Khronos draft standard).



Zapisuje samonosný binární `.glb` soubor (žádná samostatná bin příloha) podle specifikace rozšíření KHR\_gaussian\_splatting. Pozice se ukládají jako regulární glTF POSITION vertex data (float3), všechny ostatní atributy (rotace jako float4, škála jako float3, opacity jako float, SH koeficienty jako float3 × sh-CoeffCount) leží v dodatečných vertex attributech a jsou referencovány přes rozšíření. Důležité: glTF používá pravotočivý Y-up souřadnicový systém, COL-MAP/3DGS pracuje Y-down / Z-forward. Exportér proto aplikuje 180° rotaci kolem X osy — pozice se přepisují jako  $(x, -y, -z)$ , quaterniony se upravují na  $(w, x, -y, -z)$ . To dává geometricky korektní, ručnou (ne zrcadlově převrácenou) reprezentaci v glTF prohlížečích. JSON a binární chunky jsou padované na 4-bajtové zarovnání, jak vyžaduje GLB standard.

### JEDNODUŠE ŘEČENO

Oficiální Khronos standardní formát pro 3D data, v čerstvém rozšíření pro Gaussian Splats. Výhoda: glTF je rozšířený ve všech velkých 3D enginech (Babylon.js, Three.js, Unity, Unreal). Nevýhoda: rozšíření je v roce 2026 ještě v draft stádiu, mnoho prohlížečů ho ještě neumí. Smysluplné především, pokud integruješ splat data do existující glTF pipeline nebo píšeš prohlížeč, který už umí glTF.

## I E6 — Splat (.splat)



KDE

Řádek nabídek → Export → 3D Formats → Export .splat... Režim pro začátečníky: karta formátu „.splat„. **Velikost:** přesně 32 bajtů na Gaussian. **Kompatibilní s:** gsplat.js, web prohlížeče (antimatter15 reference), většina browser 3DGS dem.

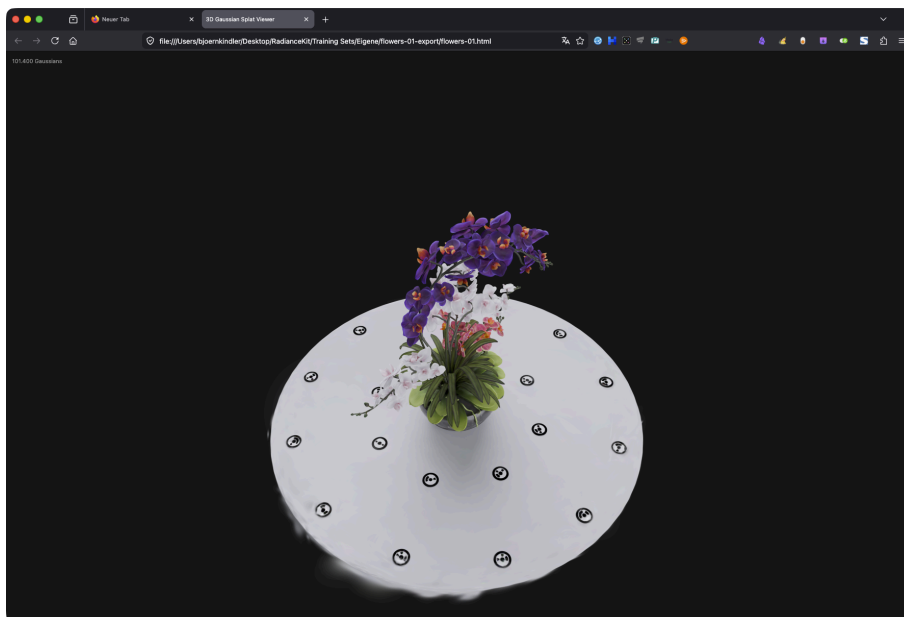


TECHNICKY

antimatter15 `.splat` formát — 32 bajtů na Gaussian, žádná hlavička, žádná indirekce. Layout na záznam: 3 × float32 pozice (světové souřadnice), 3 × float32 škála (exp transformované z log space interního bufferu), 4 × uint8 RGBA barva (DC-SH koeficient škálovaný s `SH_C0 = 0.282...` a clamped na [0,255]), 4 × uint8 quaternion (w,x,y,z, normalizovaný a kódovaný jako `128 + 128*q` do bajtového rozsahu). Ukládá se pouze DC-SH — vyšší SH bandy jsou zahozeny. To dělá formát extrémně kompaktní, ale stojí to view-dependent změny barvy, ke kterým dochází u odrazů nebo spekulárních highlights. Pořadí zápisu je přesně indexové pořadí cloudu (žádné prostorové třídění), web prohlížeče jako `gsplat.js` z toho při renderování vychází.

### JEDNODUŠE ŘEČENO

Formát volby, pokud chceš splat zobrazit ve vlastním web prohlížeči s `gsplat.js`. Velmi kompaktní (32 bajtů/Gaussian), ale žádný vyšší SH stupeň — tedy žádné lesklé odrazy ani jemné barevné změny podle úhlu pohledu. Pro většinu webových aplikací nic moc, protože DC barva úplně stačí a chybějící view-dependence stěží padne do oka.



Obrázek 31: Web Viewer otevřený ve Firefoxu — splat scéna renderovaná s okolními marker sférami kamer, lišta záložek prohlížeče nahoře viditelná, žádný CDN/server setup nepotřeba. Samostatný `flowers-01.html` přímo z Finderu otevřený dvojklikem ve výchozím prohlížeči — vložený WebGL2 program okamžitě renderuje Gaussian Cloud, bez sítě nebo serveru. Černé markery kolem scény jsou tréninkové kamery, volitelně zobrazitelné. Drag myší rotuje, scroll zoomuje.

## I E7 — Web Viewer (.html)



Řádek nabídek → Export → Media → Export Web Viewer... Režim pro začátečníky: karta formátu „Web Viewer,“. **Velikost:** splat data base64 kódovaná ( $\approx 4/3$  režie) + cca 5 KB HTML/JS shell.

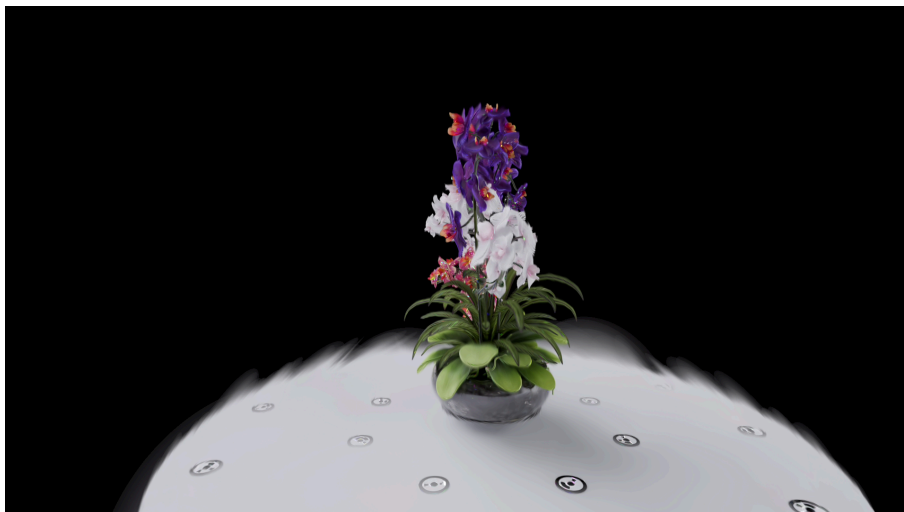
**Kompatibilní s:** každý moderní prohlížeč s WebGL2 (všechny desktopy, iOS 15+, Android 5+).



Balí Gaussian Cloud spolu s kompletně inline napsaným WebGL2 renderem do jediného `.html` souboru. Žádné CDN závislosti, žádné WASM, žádný druhý soubor. Cloud se interně nejprve kóduje jako `.splat` binárka (stejná 32-bajtová logika jako E6), pak se vloží base64, pak se v prohlížeči dekóduje s `atob`. Vestavěný renderer dělá vlastní WebGL2 třídění, ovládání orbit myši a CPU třídění na frame; veškerý JS kód (shadery, matematika, smyčka) je v output HTML viditelný. Konvence os na hranici úložiště-renderer je přesně stejná jako u E5: pozice  $(x, -y, -z)$ , quaternion  $(w, x, -y, -z)$ . Volitelně lze zobrazit branding overlay (free tier přepínač). Vzhledem k tomu, že vše je inline, soubor funguje i přímo z protokolu `file://` — pro testování není potřeba lokální web server.

### JEDNODUŠE ŘEČENO

Jediný HTML soubor, který můžeš někomu poslat mailem nebo vložit na webovou stránku. Dvojklik ve Finderu a prohlížeč ukazuje tvou scénu s otáčením myši. Žádný upload do cloudu, žádný druhý soubor, žádný server. Ideální pro klientské prezentace, portfolio, mailové přílohy. Nevýhoda: soubor je díky base64 kódování asi o třetinu větší než čistý `.splat` — pro velmi velké scény se proto vyplatí samostatné hostování `.splat` souboru spolu se standardním prohlížečem.



Obrázek 32: Jeden frame extrahovaný z `flowers-01.mp4` — bílá platforma s marker kamerami viditelná, černé pozadí (výchozí pozadí náhledu, lze změnit v Nastavení). Kamera obíhá scénu po parametrické dráze (elevace + vzdálenost pevné, yaw rotuje), doba typicky 6–10 sekund při 30 nebo 60 fps. Rozlišení framu škálovatelné od 480p do 8K přes VideoPreset.

## I E8 — Orbit Video (.mp4/.mov)



Řádek nabídek → Viewport → Record Turntable Video NEBO Řádek nabídek → Export → Media → Export Orbit Video.... Režim pro začátečníky: karta formátu „Orbit Video“, s posuvníkem trvání 3–30 s. **Velikost:** závisí na trvání, rozlišení, bitrate. **Kompatibilní s:** všechny platformy (H.264 a HEVC jsou Apple standard).



Renderuje Gaussian Cloud podél parametrické orbitální jízdy kamery a každý frame enkóduje přes AVASSETWriter do MP4 nebo MOV souboru. Konfigurace orbity řídí rychlost otáčení (otáčky), vzdálenost, elevaci, FOV, trvání a faktor ease-in/out. Export orbitálního videa běží přes vlastní ForwardPass RadianceKitu s plným vyhodnocením SH — pixel po pixelu identický s in-app viewportem (WYSIWYG). Na frame se matice zarovnání světa (vypočítaná rendererem, aby interní souřadnice otočila do Y-up orbitálního světa) násobí kamerou, poté se aplikuje překlopení konvence kamery (camFlip: orbit Y-up → COLMAP Y-down). Offscreen render target se přes IOSurface stahuje do CVPixelBuffer pro enkodér. Enkodér podporuje H.264 a HEVC, konfigurovatelný bitrate a rozlišení od 480p do 8K. Před prvním framem renderer čeká 200 ms, aby bylo dokončeno počáteční třídění splatů. Tento export je GPU bound — u 8K a milionů Gaussianů leží render čas na frame v několika sekundách, takže celkové render časy 10–30 minut pro 6sekundové video jsou možné.

### JEDNODUŠE ŘEČENO

Hotový MP4 soubor s rotací kolem tvé scény. Perfektní pro sociální sítě, marketing, představení. Můžeš nastavit trvání (3–30 sekund), směr rotace a rychlost. Soubor lze přímo vložit na YouTube, Instagram, do PowerPointu a kamkoli jinam. Občas to jde pomalu, protože aplikace musí každý frame kompletně renderovat — pro 8K video plánuj pět až třicet minut, podle počtu Gaussianů.

## I E9 — SfM Transforms (transforms.json)

### KDE

Řádek nabídek → Export → Photogrammetry → Export SfM (transforms.json)... **Velikost:** typicky 1–10 KB (pouze pozice + intrinsics, žádné obrázky, žádné Gaussiany). **Kompatibilní s:** nerfstudio, Brush, gsplat, OpenSplat, Meshroom, všechny moderní feed-forward 3DGS trénerky.

### TECHNICKY

Zapisuje nerfstudio `transforms.json` formát se seznamem pozic kamer plus sdílenými intrinsics. Na kameru se invertuje view matice (uvnitř RadianceKitu: world-to-camera v COLMAP konvenci), následně se zrcadlí Y a Z bazální vektory v kameře, aby se převedly do nerfstudio konvence (OpenGL style, kamera se dívá podél `-Z`, `+Y` je nahoře). Finální 4×4 matice končí jako row-major nested array doublů v poli `transform_matrix` každého framu. Intrinsics se ukládají na top-level (ohnisko `x/y`, hlavní bod `x/y`, šířka/výška obrázku, `camera_model = "OPENCV"`, plus distortion koeficienty `k1, k2, p1, p2`) — pokud exportér nedetekuje více různých intrinsic setů, pak se zapisují per frame. Cesty obrázků se píšou jako `images/<filename>` relativně k JSON souboru; uživatel musí vytvořit sibling složku `images/` s tréninkovými fotografiemi.

### JEDNODUŠE ŘEČENO

Tento JSON soubor popisuje pro každou fotografii, kde stála kamera a kam se dívala. Samotný soubor je malý a neužitečný — používá se společně s originálními obrázky ve složce. Nerfstudio, Brush a několik dalších trenérů čte přesně tento formát, takže můžeš své RadianceKit SfM výsledky předat do jiného nástroje, aniž by tam bylo potřeba znovu počítat rekonstrukci kamer. U velkých scén to šetří hodiny.

## I E10 — COLMAP Workspace (sparse/0/)



Řádek nabídek → Export → Photogrammetry → Export SfM (COLMAP Workspace).... **Velikost:** tři binární soubory dohromady typicky 4–8 MB — `points3D.bin` dominuje (jeden řádek na 3D bod sparse cloudu), `images.bin` a `cameras.bin` jsou každý výrazně pod 100 KB. **Kompatibilní s:** COLMAP sám, Nerfstudio, Postshot, Meshroom, všechny nástroje, které očekávají COLMAP `sparse/` adresař.



Zapíše standardní COLMAP `sparse/0/` layout se třemi binárními soubory: `cameras.bin`, `images.bin`, `points3D.bin`. Formátová reference je oficiální COLMAP dokumentace. `cameras.bin` obsahuje deduplikovaný seznam intrinsics (kamery s identickými intrinsics + velikostí obrazu se sloučí do jednoho záznamu); použitý camera model je `OPENCV` (model 4), s `fx/fy/cx/cy` plus čtyřmi distortion koeficienty `k1/k2/p1/p2`. `images.bin` listuje na obrázek pozici jako `wxyz` quaternion plus translaci, následuje ID kamery a název souboru; žádné 2D-3D korespondence se neukládají. `points3D.bin` obsahuje SfM mrak bodů s pozicí, barvou (0-255 RGB) a defaultními hodnotami pro reprojekci a délku tracku. Vše se zapisuje v little-endian. Re-import do RadianceKitu funguje přes File menu → „Import COLMAP/Metashape Workspace...“ (viz Q3 v kapitole SfM backendů).

### JEDNODUŠE ŘEČENO

Oficiální COLMAP formát. Pokud chceš pokračovat ve svém tréninku v Postshot, Nerfstudio nebo jiném COLMAP-schopném softwaru, je to ta cesta. Tři malé soubory plus tvé originální obrázky, a cílový program to akceptuje, jako by COLMAP sám byl zdrojový program. Více programů to chápe než formát `transforms.json` (E9), zároveň trochu méně pohodlné, protože binární místo textového.

## Který formát kdy?

Cíl	Formát
Web prohlížeč na vlastní stránce	E7 Web Viewer (.html)
Web prohlížeč s <code>gsplat.js</code>	E6 Splat (.splat)
Znovupoužití pipeline v Postshot / Nerfstudio	E9 transforms.json 1. E10 COLMAP Workspace
SuperSplat editace	E1 PLY nebo E2 Compressed PLY
Niantic Scaniverse / Spatial Fields	E3 SPZ
Maximální komprese	E4 SOG (vyžaduje cwebp)
Marketingové/sociální video	E8 Orbit Video

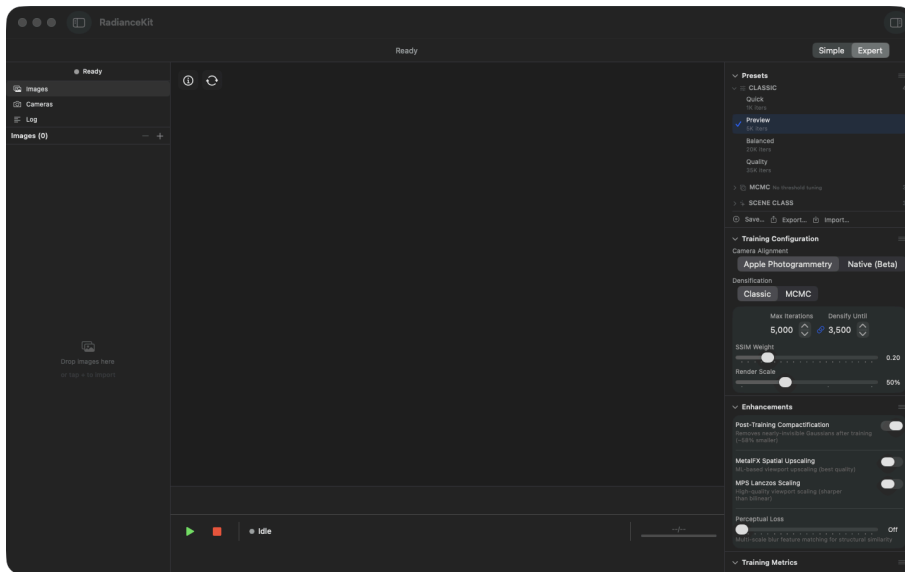
## Rychlé srovnání

Formát	Přípona	Sandbox	Velikost (1M Gauss)	Nejlepší použití
E1 PLY	<code>.ply</code>	ano	~250 MB	Archiv, nejvyšší kompatibilita
E2 Compressed PLY	<code>.ply</code>	ano	~40 MB	Web + SuperSplat
E3 SPZ	<code>.spz</code>	ano (gzip spawn)	~40 MB	Niantic + Mobile
E4 SOG	<code>.sog</code>	podmíněně (cwebp)	~20 MB	Maximální komprese
E5 glTF	<code>.glb</code>	ano	~250 MB	Khronos pipeline
E6 Splat	<code>.splat</code>	ano	~32 MB	gsplat.js web prohlížeč
E7 Web Viewer	<code>.html</code>	ano	~45 MB	Samostatný souborový prohlížeč
E8 Orbit Video	<code>.mp4 / .mov</code>	ano	proměnná	Social/Marketing
E9 SfM Transforms	<code>.json</code>	ano	~5 KB	Předání pozic
E10 COLMAP Workspace	Adresář	ano	~4–8 MB	Předání pozic binárně

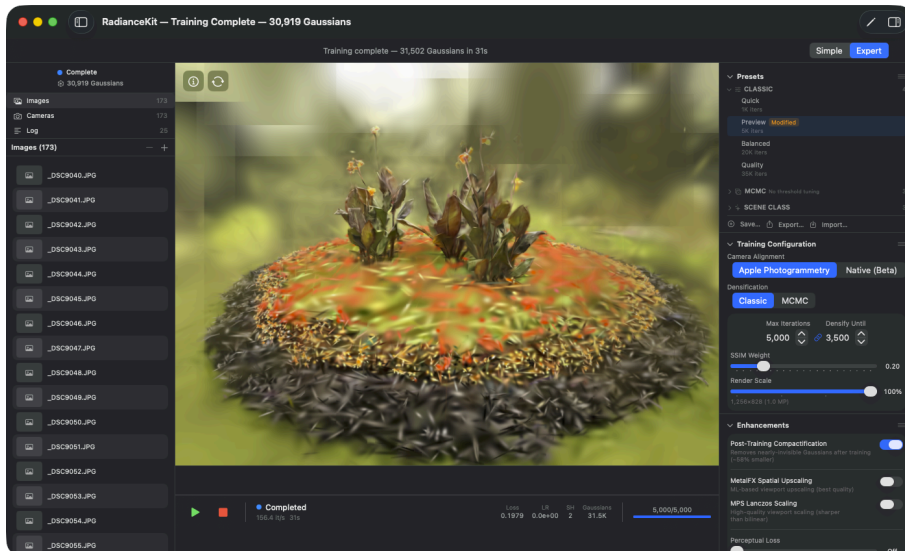
Sloupec velikost jsou hrubé orientační hodnoty pro 1 milion Gaussianů s SH stupněm 3. Reálné hodnoty se liší podle stlačitelnosti scény; SH stupeň 0 redukuje PLY/glTF faktorem 4.

## KAPITOLA

# Kapitola 9 — SfM Backendy



Obrázek 33: Expertní režim s výběrem Camera Alignment v Inspektoru (Apple Photogrammetry / Native (Beta))



Obrázek 34: Inspektor s aktivním Native (Beta) — Camera Alignment picker, druhá možnost vybrána, všechny ostatní parametry konfigurace tréninku beze změny

**CO JE NA OBRÁZKU** Camera Alignment picker v Inspektoru je segmentovaný ovládací prvek se dvěma možnostmi — Apple Photogrammetry (výchozí pro App Store buildy, plně sandbox-kompatibilní) a Native (Beta) (vlastní FAST+BRIEF+GLOMAP pipeline backend

RadianceKitu, vyvinutý během fáze 3.8/3.9, stav 2026-05). Native (Beta) je validován pouze pro orbitální záběry a je rychlejší než Apple Photogrammetry při  $\geq 1\,000$  snímcích, ale ještě nesplňuje quality gate fáze 3 §5 (finalLoss  $\leq 0.0115$ ) — odtud označení Beta. Externí SfM výsledky z Metashape, COLMAP nebo jakéhokoli jiného fotogrammetrického softwaru lze navíc importovat přes File menu (Q3 textový formát COLMAP, Q6 Workspace Import) — picker se nezmění, ale importované pozice nahradí výsledek SfM.

SfM znamená **Structure from Motion**. Z množiny překrývajících se fotografií software pro každý snímek rekonstruuje pozici a směr pohledu kamery ve společném 3D souřadnicovém systému. Při tom vzniká hrubý 3D mrak bodů, kterým se inicializuje trénink Gaussian Splatting. Výsledek SfM je vstupem pro samotný trénink a rozhodujícím způsobem určuje pozdější kvalitu obrazu.

RadianceKit nabízí pět SfM cest: dva backendy vestavěné v aplikaci (Q1 Apple Photogrammetry, Q4/Q5 Native), dvě importní cesty z externích nástrojů (Q3 textový formát COLMAP, Q6 binární Workspace Import), plus Q2 COLMAP binary, který je dostupný pouze ve vývojářských buildech mimo App Store. Která je ta správná, závisí na typu scény (orbit kolem objektu, interiér, dronový let) a na tom, zda externí software již poskytuje rekonstrukci.

## I Q1 — Apple Photogrammetry



Expert View → Inspektor → Konfigurace tréninku → Camera Alignment picker, položka „Apple Photogrammetry„.



Obaluje vestavěný fotogrammetrický framework od Apple, který byl původně vyvinut pro Object Capture. Apple interně extrahuje features proprietární pipeline (kroky nejsou veřejně dokumentovány), ověřuje je přes multi-view matching a řeší bundle adjustment na Apple Silicon Neural Engine + GPU. Backend je plně App Store kompatibilní (žádná externí binárka, Sandbox=true, on-device), ale dodává pouze pozice kamer plus hrubý mrak bodů — žádné diagnostické metriky jako délka tracku nebo reprojekovaná chyba. Podle doporučení Apple škáluje do několika set obrázků. Při více než ~500 snímcích v lineárních dronových letech nebo velkých outdoorových scénách byly reprodukovatelně pozorovány pády nebo tiché zahození jednotlivých kamer.

### JEDNODUŠE ŘEČENO

Toto je nejjednodušší cesta. Obrázky dovnitř, aplikace počítá. Funguje velmi dobře u klasických skenů objektů — když obejdeš kus nábytku nebo sochu a uděláš 50–200 fotografií. U dronových letů nad krajinou nebo při velmi mnoha obrázcích (přes 500) se ale Applův postup rád stane nestabilním. Pro takové scény vyzkoušej Native backend (Q4/Q5) nebo spočítej kamery v Metashape a načti je přes Workspace Import (Q6).

**POWER-USER**

Q2 COLMAP binary — spouští externí program COLMAP jako podproces a proto **není dostupný** v App Store verzi (sandbox). Funguje pouze ve vývojářských buildech mimo App Store. Pro kvalitu, kterou COLMAP poskytuje, existuje v App Store verzi Workspace Import (Q3 nebo Q6): spočítej SfM v COLMAP nebo Metashape externě a načti výsledek.

**Q3 — COLMAP textový formát (Metashape / ETH3D)**

KDE

Menu „File → Import COLMAP / Metashape Workspace...“ (Cmd+⇧+I) NEBO drag-and-drop složky s `sparse/0/cameras.txt`.



TECHNICKY

Čte standardizovaný textový export COLMAP — tři textové soubory `cameras.txt`, `images.txt`, `points3D.txt` v podsložce `sparse/0/` — a převádí je do interního modelu výsledku SfM. Stejná definice formátu jako binární export COLMAP, jen jako ASCII místo binárky. V přesně tomto rozložení je exportován Agisoft Metashape, RealityCapture, PolyCam a ETH3D benchmark. Parser sdílí detekci modelu kamery s binárním parserem (všechny běžné modely: SIMPLE\_PINHOLE, PINHOLE, OPENCV, OPENCV\_FISHEYE, FULL\_OPENCV). Robustní vůči komentářovým a prázdným řádkům. V testech škáluje až do ~1 400 kamer (ETH3D Tunnel) bez problémů.

**JEDNODUŠE ŘEČENO**

Pokud jsi již pracoval s Metashape, RealityCapture nebo jiným komerčním foto-3D softwarem a exportoval jsi výsledek — tento export můžeš v RadianceKitu načíst přímo, aniž by aplikace musela přepočítávat. Šetří to hodiny čekání. Stačí načíst celou složku přes File menu nebo ji přetáhnout do okna.

## I Q4 — Native SfM (inkrementální)



Expert View → Inspektor → Konfigurace tréninku → Camera Alignment picker, položka „Native (Beta)“, „. Inkrementální je výchozí režim tohoto backendu — v Inspektoru není samostatný picker pro mapper. Přes CLI lze režim explicitně nastavit pomocí `--native-sfm` nebo `--sfm-mapper incremental`.



Vlastní GPU-akcelerovaná implementace celé SfM pipeline: FAST+BRIEF features NEBO SuperPoint+LightGlue přes CoreML (s `--coreml-features`), následované Hamming-KNN matchingem, RANSAC fundamentální maticí, budováním tracků, výběrem počátečního páru, two-view bootstrapem (F→E plus DLT), greedy inkrementálním mapperem s PnP registrací a multi-view triangulací, a finálním bundle adjustmentem přes Schur-redukovaný Levenberg-Marquardt s Huber loss a analytickými Jakobiány přes Cholesky solve. Plně App Store kompatibilní: žádná externí binárka, `Sandbox=true`. S detektorem R2-collapse dodaným ve fázi 3.10: pokud aplikace zaregistruje méně než 60 % vstupních snímků nebo pokud poměr `points-per-camera` klesne pod 13, automaticky se přepne na globální mapper (Q5). Empiricky čistý na orbit/turntable scénách; u obecnějších pohybů (dronový let, interiéry s komplexní geometrií) je úspěšnost nižší — detektor ale tyto případy zachytí. Spolehlivě škáluje do ~200 kamer, výše s výrazně delším časem běhu.

### JEDNODUŠE ŘEČENO

Apple-podobné silné stránky (App Store kompatibilní, rychlé pro orbity) s navíc diagnostickými hodnotami. Funguje obzvláště dobře, když obcházíš objekt jako u Object Capture. U složitějších záběrů (dronový let nebo obývací pokoj) RadiancKit automaticky pozná, že to nebude dobré, a přepne se na globální metodu. Označeno „Beta“, protože je stále v testování — standardní doporučení zůstává Apple Photogrammetry pro jednoduché skeny objektů a Workspace Import (Q3 nebo Q6) pro náročné outdoorové sety.

## I Q5 — Native SfM (globální)



Volá se automaticky, když inkrementální mapper (Q4) spustí collapse detektor (méně než 60 % vstupních snímků zaregistrováno nebo poměr points-per-camera pod 13). Ručně lze vynutit pouze přes CLI `--sfm-mapper global`. V Inspektoru není globální metoda dostupná přes samostatný picker — aplikace sama rozhoduje, kdy se přepnout.



Globální varianta nativní pipeline. Nejprve extrakce features + matching jako u Q4, poté odhad relativní pozice pro všechny ověřené páry, následované rotation averaging (synchronizuje všechny rotace kamer ve světovém souřadnicovém systému) a translation averaging (LSQR-založené na matrix-free sparse formulaci, aby se zabránilo přetečení integeru u velkých počtů kamer). V principu škáluje na ~5 000 kamer, v praxi je kvalita degradovaná nad několika sty kamerami — měření Phase 3.8 §5 akceptační brány na K-1351 dalo finalLoss 0.07 místo cílových 0.0115. Považováno za „fallback tier“: nasazuje se, když inkrementální mapper degeneruje, ale samo se znovu nekontroluje na kvalitu.

### JEDNODUŠE ŘEČENO

Plán B pro nativní engine. Volá se automaticky, když rychlejší inkrementální cesta selže. Dodává použitelný výsledek, ale u velmi velkých nebo obtížných scén obvykle není tak přesný jako to, co dostaneš z Metashape nebo externí instalace COLMAP. Pokud se Native stane tvým standardním workflow, vyplatí se v takových případech zacházka přes Workspace Import (Q3 nebo Q6).

## I Q6 — Import Metashape / COLMAP textového Workspace (fáze Q7)



File menu → „Import COLMAP / Metashape Workspace...“ (Cmd+⇧+I). Drag-and-drop složky s `sparse/0/cameras.{bin,txt}` a `images/`.



Automaticky detekuje, zda složka vybraná přes drag-and-drop nebo Open Panel odpovídá jednomu ze tří layoutů COLMAP workspace (`sparse/0/`, `sparse/`, nebo `root`) a zda je rekonstrukce binární (`cameras.bin`) nebo textová (`cameras.txt`). Binární cesta používá COLMAP binární parser, textová cesta ETH3D loader — oba produkují stejný model výsledku SfM a zbytek pipeline (import obrázků, spuštění MCMC tréninku) je agnostický ke zdroji. Obrázky se otevírají přes sandboxový bookmark systém aplikace jako security-scoped, takže import funguje i v App Store verzi. Speciálně určeno pro případ „Metashape export bez nutnosti přepočítat rekonstrukci“. Detekce zmíněná v položce File menu varuje v logu aplikace, pokud vybraná složka není rozpoznatelný workspace.

### JEDNODUŠE ŘEČENO

Speciálně funkce pro uživatele Metashape. Pokud máš licenci pro Metashape nebo RealityCapture a tam jsi udělal rekonstrukci kamer, můžeš složku s exportem jednoduše přetáhnout sem a okamžitě spustit trénink. U velkých scén šetří několik hodin výpočtu, protože RadianceKit pak SfM sám nedělá.

## Který backend kdy?

Scénář	Doporučený backend
Sken objektu, 50–200 fotografií	Q1 Apple Photogrammetry
Velký outdoor / dron / >500 obrázků	Q6 Workspace Import (počítat v Metashape nebo COLMAP, poté načíst)
Existuje export z Metashape/RealityCapture	Q6 Import (SfM není potřeba)
ETH3D / akademický COLMAP textový set	Q3 COLMAP textový import
Striktně App Store kompatibilní + orbit scéna	Q4 Native inkrementální
Q4 selže	Q5 Native globální (automaticky)
ETH3D benchmark data	Q3 (autotest precomputed)

## Rychlé srovnání

Bac- kend	App Store	Sand- box	Externí binárka	Nejlepší použití	Max ~ka- mer
Q1 Apple PG	✓	✓	—	Orbit-ob- jekt	~300
Q2 COL- MAP Bi- nary	✗ (jen de- veloper build)	—	colmap/glomap	Velký out- door	~5 000
Q3 COL- MAP tex- tový im- port	✓	✓	—	Bench rigy	~1 500
Q4 Nati- ve inkre- mentální	✓	✓	—	Orbit-ob- jekt	~200
Q5 Na- tive glo- bální	✓	✓	—	Q4 fall- back	~1 351
Q6 Works- pace Im- port	✓	✓	—	Reuse Me- tashape	dle zdroje

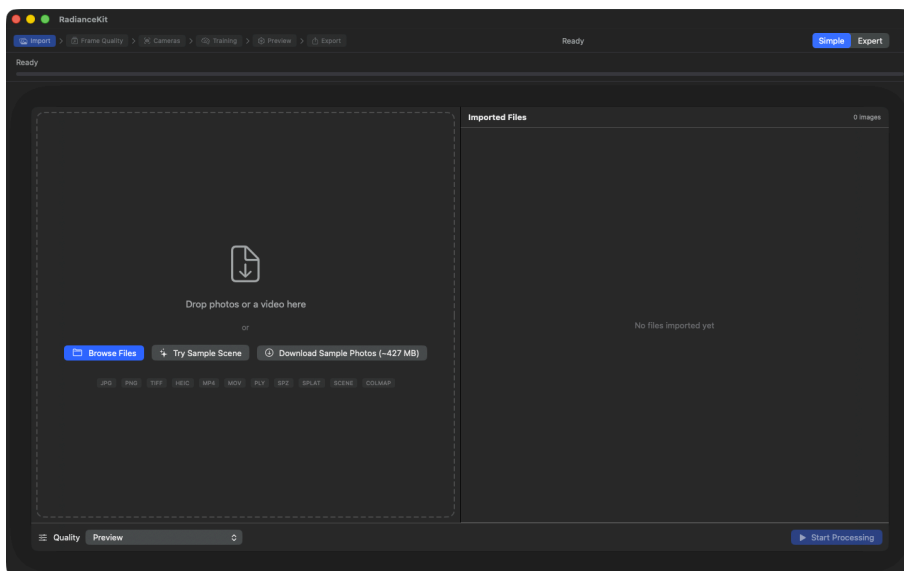
## KAPITOLA

## Kapitola 10 — Režim pro začátečníky

Režim pro začátečníky (Simple Mode, Cmd+1) je řízený workflow pro všechny, kteří poprvé rekonstruují 3D Gaussian Splatting scénu. Místo toho, aby zobrazoval sidebar plný polí Inspektoru, vede tě aplikace čtyřmi kroky: nejprve importovat obrázky nebo video a vybrat kvalitativní předvolbu, poté běží zpracování (SfM + trénink), poté lze hotovou scénu posoudit v 3D náhledu, a na konec se exportuje do požadovaného formátu. Úzká progress lišta na horním okraji okna kdykoli ukazuje, ve kterém kroku se nacházíš.

Ve srovnání s Expert Mode (Cmd+2), který zobrazuje všechny ovládací prvky najednou, Režim pro začátečníky skrývá nepoužité volby, dává validační varování při příliš málo nebo špatných obrázcích a na každém kroku nabízí pouze tlačítka, která mají v aktuálním stavu smysl. Kdykoli můžeš přepínat mezi Začátečník a Expert mode (Cmd+1 / Cmd+2), celý stav — importované obrázky, zvolená předvolba, právě běžící trénink, hotový mrak bodů — zůstává zachován a je v druhém režimu okamžitě k dispozici.

### Z1 — Import (vybrat obrázky a předvolbu)



Obrázek 35: Simple Mode krok 1 — prázdná drop zone před importem obrázků, crumb-trail nahoře (Import → Frame Quality → Cameras → Training → Preview → Export), format pills JPG/PNG/TIFF/HEIC/MP4/MOV/PLY/SPZ/SPLAT/SCENE/COLMAP

**CO JE NA OBRÁZKU** Crumb-trail (Import aktivní) ukazuje čtyřkrokový workflow. Levá drop zone se třemi CTA: „Browse Files“, (NSOpenPanel), „Try Sample Scene“ (bundled demo), „Download Sample Photos (~427 MB)“, (Mip-NeRF360 flowers subset). Format pills pod

tím listují všechny akceptované typy souborů. Vpravo „Imported Files“ s counterem „0 images“, a empty stavem „No files imported yet“. Dole quality picker (default: Preview) a „Start Processing“, (deaktivováno, dokud nejsou obrázky).

První krok spočívá v tom dát aplikaci obrazový materiál. Přes drag-and-drop do velkého přerušovaného pole uprostřed, přes tlačítko „Browse Files“, nebo přes klik na dodanou sample scénu. Vpravo se objevuje seznam všech importovaných obrázků s rozlišením a velikostí souboru; dole v plovoucí nástrojové liště volíš kvalitativní předvolbu a startuješ s „Start Processing“ pipeline. Validační varování (červené při < 3 nebo < 10 obrázcích, oranžové při 10–19) ukazují, zda aplikace očekává smysluplnou rekonstrukci, nebo ne.

### C-01 ProgressIndicator (zobrazení kroků)

#### KDE

Nahoře nad workflow, vždy viditelné.

#### TECHNICKY

Zobrazuje horizontální progress lištu přes celou pipeline (Frame-Quality → SfM → trénink) se stage alokací: Frame-Quality zabírá 0–5 % (fáze 3.11, velmi krátká), SfM zabírá 0–30 % lišty, trénink 30–100 %. Vedle status text a fází pojmenovaná procentní indikace („SfM 41 %“, „Training 12 500/20 000“), aby uživatel nečetl zdánlivý krok zpět „41 % SfM → 25 % training“, jako chybu — lišta ukazuje celkový postup pipeline, ne sub-stage. ETA výpočet začíná, jakmile je naměřeno dost tréninkového tempa (typicky po prvních 100 iteracích). Tatáž indikace se používá i v Expert Mode nad Inspektorem.

#### JEDNODUŠE ŘEČENO

Úzká lišta úplně nahoře je tvá mapa workflow. Řekne ti nejen, co aplikace momentálně dělá (vyrovnává kamery, běží trénink, ...), ale i kolik už dohromady udělala. Rozdělení je záměrně takové, že výpočet kamer zabírá první třetinu lišty a samotný trénink poslední dvě třetiny — jinak by to působilo, jako by se postup po SfM náhle vrátil k nule. Můžeš se tedy v klidu zaklonit, pohled na lištu stačí, aby ses viděl hrubou etapu. Text vedle ti říká, zda jsi právě v SfM stupni (např. „SfM 41 %“,) nebo v tréninku (např. „Training 12 500/20 000“), aby čísla nepůsobila matoucí. Pokud nedostáváš ETA zobrazené, je trénink prostě ještě moc mladý — aplikace odhaduje až poté, co naměřila dost tempa.

### C-03 DropZoneView (drag-and-drop oblast)



Levá strana kroku Import, velký přerušovaný obdélník se symbolem. V Režimu pro začátečníky se zobrazuje s labelem „Drop photos or a video here“.



Drop oblast, která krátce nadhodí symbol a obarvuje pozadí, jakmile drag items visí nad polem. Akceptuje JPG, PNG, TIFF, HEIC, MP4, MOV, PLY, SPZ, .splat, .radiancecene bundly a adresáře. Drop routing podle typu: obrázky se sbírají a předávají se řazené, videa spouštějí frame sampling cestu, splat soubory přímo otevírají náhled, scene bundly se načítají. Adresáře se enumerují a všechny obsažené obrázky se importují. Security-scoped bookmarky pro sandbox kompatibilní přístup se správně přijímají a uvolňují. Nepodporované koncovky se zobrazují jako warning banner po 5 sekund.

#### JEDNODUŠE ŘEČENO

Velké přerušované pole je hlavní ovládání prvního kroku. Stačí přetáhnout fotografie nebo video dovnitř, nebo celou složku — aplikace bere vše, co zná, a ignoruje zbytek. Pokud pole zmoudrá a symbol krátce nadhodne, aplikace drag rozpoznala. Pušť a import okamžitě startuje: obrázky putují do seznamu vpravo, videa automaticky spustí krok frame sampling, a již natrénované soubory `.ply` / `.spz` / `.splat` přímo otevírají náhled. Pokud formát vůbec neseď (např. PDF nebo BMP), objeví se krátká poznámka na horním okraji — aplikace nepolyká neznamý materiál tiše.

### C-05 Tlačítko Browse Files



Uvnitř drop zone, prominentní tlačítko.



Tlačítko, které otevírá macOS souborový dialog s vícenásobným výběrem a typy souborů JPG, PNG, TIFF, MP4, MOV, složky plus aplikací vlastní scene formát. Výsledné URL jsou security-scoped a předávají se přes stejné import cesty jako drag-and-drop. Pokud uživatel vybere složku, je enumerována rekurzivně za obrázky.

#### JEDNODUŠE ŘEČENO

Pokud ti drag-and-drop nevyhovuje, klikni jednoduše toto tlačítko a naviguj v macOS souborovém dialogu ke svým fotkám. Můžeš vybrat více souborů najednou (Cmd-klik na jednotlivé obrázky) nebo vybrat celou složku — aplikace pak složku rekurzivně prohledá za všemi podporovanými typy obrazu. To je obzvláště praktické, pokud tvé záběry leží vnořené v podsložkách (např. „shoot-day1/“, „shoot-day2/“) — jeden klik na hlavní složku stačí. Funkčně tlačítko dělá přesně to, co i drag-and-drop; vyber prostě cestu, která je pro tebe pohodlnější.

**C-06** Tlačítko Try Sample Scene

Uvnitř drop zone, viditelné pouze pokud bundle aplikace obsahuje sample scénu a ještě nebyly importovány obrázky/splaty.

**TECHNICKY**

Objevuje se pouze pokud (a) je v bundlu aplikace přítomna `sample-scene.splat`, `.spz` nebo `.ply` A (b) ještě nebyly importovány obrázky/video a neexistuje mrak bodů. Při kliknutí načte hotový mrak bodů (preferenčně nejmenší formát — `.splat` ~3 MB, `.spz` ~1.4 MB, fallback `.ply`) a po 400 ms nastaví napevno kódované hodnoty kamery z originálních metadat květinové scény pro esteticky smysluplnou výchozí perspektivu.

**JEDNODUŠE ŘEČENO**

Pokud aplikaci poprvé startuješ a chceš se prostě podívat, co na konci vyleze — klikni sem. Otevře to hotově natrénovanou květinovou scénu, kterou můžeš okamžitě otáčet a exportovat, aniž by aplikace musela počítat. Kamera je přednastavena na esteticky smysluplnou výchozí perspektivu, takže hned vidíš něco hezkého. Perfektní pro bezrizikové vyzkoušení 3D ovládání a export kroku, než se pustíš na vlastní záběry. Jakmile importuješ vlastní obrázky, tlačítko automaticky zmizí — zobrazuje se pouze, dokud je projekt úplně prázdný.

**C-07** Tlačítko Download Sample Photos

Uvnitř drop zone, vedle „Try Sample Scene“; stejné podmínky viditelnosti.

**TECHNICKY**

Spouští download (repo [github.com/bkindler/radiancekit-sample-photos](https://github.com/bkindler/radiancekit-sample-photos)), který stahuje cca 427 MB 960 plně rozlišených framů a krmí je do aplikace. Během downloadu je tlačítko deaktivováno. Postup se objevuje v horní progress baru jako „Downloading X %“, v vlastním stage, protože toto stage drží svou vlastní 0–100 % škálu a s pozdější SfM stage se nepřekrývá.

**JEDNODUŠE ŘEČENO**

Stejně jako sample scéna, jen s výchozími fotkami místo s hotevým výsledkem. Tak můžeš jednou nechat projít celou pipeline a vidět, jak dlouho SfM a trénink na tvém Macu opravdu trvají. Download je velký (asi půl DVD = 427 MB), děje se ale jen jednou — poté jsou fotky lokálně a můžeš pipeline libovolně často restartovat s různými předvolbami. Během downloadu horní progress lišta ukazuje aktuální stav downloadu v procentech, takže můžeš odhadnout, kdy začne. Tip: vezmi k tomu raději rychlou WiFi nebo kabel — 427 MB se jinak protáhne.

## C-09 Quality Presets Picker




KDE

Plovoucí dolní nástrojová lišta Import overlay, vlevo vedle Start tlačítka.



TECHNICKY

Ovládací prvek s labelem „Quality“, seskupuje dostupné předvolby podle kategorií (Classic / MCMC / Custom). Built-in předvolby jsou seskupené podle kategorie; section headery jsou napevno kódované. Custom předvolby viditelné pouze pokud nějaké existují. Locked state: předvolby, které neleží na free seznamu (Quick + Preview), dostávají suffix „“ k názvu, pokud uživatel nemá zakoupeno; při výběru picker skočí zpět na Preview a automaticky otevírá Purchase Sheet. Při výběru se předvolba aplikuje, což nahrazuje celou tréninkovou konfiguraci.

### JEDNODUŠE ŘEČENO

Zde vybíráš, jak přesně a jak dlouho má aplikace počítat. „Quick“, a „Preview“ jsou použitelné bez nákupu a dodávají během několika minut první výsledek — ideální pro test, zda jsou tvé obrázky vůbec smysluplné. „Balanced“, a „Quality“ potřebují plnou verzi a dodávají výrazně čistší modely, trvají ale hodiny místo minut. MCMC je jiná strategie, která si vystačí s méně Gaussovými splaty — dobrá, pokud chceš model později kompaktně exportovat nebo vystavit na web. Premium předvolby poznáš podle malých ikonů zámku u názvu; pokud klikneš na jednu bez licence, picker skočí zpět na Preview a kupní sheet se automaticky otevře. Pravidlo: vždy startuj s Preview, podívej se na výsledek a pak se rozhodni, zda se delší běh vyplatí.

## C-10 Tlačítko Start Processing



KDE

Plovoucí dolní nástrojová lišta Import overlay, vpravo vedle preset pickeru.



TECHNICKY

Tlačítko, které zůstává šedé, dokud nejsou importovány ani obrázky, ani video. Při kliknutí spouští pipeline a přepíná stage machine do pořadí Frame-Quality → SfM → trénink. Tlačítko samo nemá další status; běžící zpracování se zobrazuje místo toho jako samostatná zpracovací obrazovka.

### JEDNODUŠE ŘEČENO

„Začneme“, knoflík. Dokud je šedý, chybí ještě vstupní obrázky nebo video. Jakmile jsi fotky vtáhl, stává se aktivním a klikáš na něj, abys spustil SfM a trénink po sobě. Od tam aplikace přebírá celý průběh a automaticky přistaneš na zpracovací obrazovce (Z2). Nemusíš nic dalšího klikat — až po konci tréninku aplikace přepíná zpět do náhledu (Z3). Pokud si to ještě jednou rozmyslíš, můžeš i poté kdykoli přes Cancel přerušit.

## C-11 Video Sampling Slider



Pravý seznam obrázků, viditelný pouze pokud bylo importováno video (místo obrázků).

### TECHNICKY

Slider 0.5 fps – 30 fps v 0.5 krocích. Při změně se aktualizuje hustota framů a navíc se z hustoty a délky videa počítá počet cílových framů (minimálně 10). Slider leží mimo seznam obrázků, protože položky seznamu by blokovaly mouse eventy slideru. Pod sliderem stojí vypočtené cílové framy („247 frames,“) a délka videa („1m23s video“). Tooltip varuje: „Doubling the density doubles the number of frames and increases SfM time by ~100 %.“

### JEDNODUŠE ŘEČENO

Pokud jsi místo fotek importoval video, tento slider rozhoduje, kolik jednotlivých obrázků má aplikace z videa vytáhnout. Více obrázků = lepší kvalita, ale lineárně více výpočetního času. Pro 30sekundové orbit video je 5 fps (150 obrázků) dobrý začátek; u 1minutových záběrů často stačí 3 fps. Pod regulátorem aplikace živě ukazuje, kolik framů při aktuálním nastavení vyjde — tak vidíš okamžitě, zda jsi v smysluplném rozsahu zhruba 100–300 obrázků. Pokud výsledek bude špatný, posuň regulátor doprava a zkus znovu; zdvojnásobení frame rate ale zhruba zdvojnásobí i SfM dobu.

## C-12 Tlačítko Clear All



Pravý seznam obrázků, vpravo dole; viditelné pouze pokud byly importovány obrázky.

### TECHNICKY

Červené tlačítko. Klik otevírá potvrzovací dialog s titulem „Clear all imported files?“, a zprávou „N images will be removed.“. Potvrzení vyprázdní všechny importované obrázky/video, staging adresáře, mrak bodů, status tréninku, výsledek SfM a všechny cache; stage skáče zpět na Import. Na Cancel zůstane vše zachováno. Dialog je konfigurován jako nedestruktivní default cesta (destruktivní tlačítko červeně označeno).

### JEDNODUŠE ŘEČENO

Pokud chceš úplně začít znovu, klikni sem. Potvrzovací otázka se objevuje, protože smazání zahodí všechny aktuální importy včetně případně už spočítaných kamer a tréninkových výsledků — nemůžeš to vrátit. Smysluplné, pokud chceš zvolený obrazový materiál úplně vyměnit nebo chceš zbavit se starého projektu, než spustíš nový. Pozor: vyjmout jednotlivý obrázek se děje přes seznam vpravo (viz další bod), ne přes toto tlačítko. Tvé soubory na disku se přitom nemažou — aplikace zapomíná jen své reference.

**C-13 File List ForEach (mazání jednotlivého obrazu)**

Pravý seznam obrázků, každá položka.

**TECHNICKY**

Seznam přes importované obrázky se swipe-to-delete. Na obrázek řádek s ikonou, názvem souboru, rozlišením („1920 x 1080“) a velikostí souboru (formátováno KB/MB). Rozlišení pochází z metadata cache, která se asynchronně plní z headerů obrazu, aby se UI neblokovalo. Akce mazání nabízí macOS typický swipe delete (trackpad swipe doleva na řádku) plus klávesnicí Delete při selektované řádce. Poznámka: rozšířená image delete cesta s explicitním minus tlačítkem, Backspace a Cmd-Z pro vrácení akce byla doplněna *pouze v Expert Mode* v Project Navigatoru — v Režimu pro začátečníky zůstává swipe delete.

**JEDNODUŠE ŘEČENO**

Seznam vpravo ukazuje každý importovaný obrázek s rozlišením a velikostí souboru — praktické pro rychlé zjištění, zda jsi nesměchal vysoce rozlišený s nízkou rozlišeným materiálem. Pro vyjmutí jednotlivého obrázku ho prsty na trackpadu posuň doleva — jako v iOS Mail — nebo ho vyber a stiskni Delete. Aplikace samotný soubor nemaže; jen ho vyjímá z aktuálního projektu. Pokud potřebuješ skutečné minus tlačítko nebo Cmd-Z vrácení akce, přepni do Expert Mode (Cmd+2), tam to v Project Navigatoru existuje. V Režimu pro začátečníky zůstává záměrně u jednoduchého swipe vzoru.

**C-15 Validační varování (3stupňový tier)**

Pod seznamem obrázků, nad tlačítkem Clear All.

**TECHNICKY**

Tři po sobě jdoucí prahy založené na počtu importovaných obrázků (aktivní pouze pokud obrázky existují a žádné video): - < 3 obrázky: červený banner (red octagon), text „At least 3 images are required. Camera alignment cannot be computed from fewer images.“, - 3–9 obrázků: červený 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 obrázků: oranžový banner (warning triangle), text „Workable, but quality usually improves with 20+ images and good coverage around the scene.“ Od 20 obrázků banner mizí. Prahové hodnoty jsou napevno zakódovány a založeny na empirických 560+ tréninkových experimentech.

**JEDNODUŠE ŘEČENO**

Aplikace se dívá, kolik obrázků jsi importoval, a dává ti barevné posouzení. Červená znamená: to s vysokou pravděpodobností nic nebude — buď SfM nemůže vypočítat kamery nebo trénink overfittuje na příliš málo materiálu. Oranžová znamená: mohlo by to klapnout, ale nepočítej s top kvalitou, protože algoritmus mezi obrázky málo přesahu najde. Žádný banner znamená: dobré podmínky, máš dost materiálu. Pokud chceš skutečně čisté modely, namiř alespoň 30–50 rovnoměrně rozložených záběrů kolem svého motivu — rád i výrazně víc u venkovních scén nebo velkých místností. Můžeš navzdory varování startovat, ale neptej se překvapeně, pokud SfM bez komentáře přeruší nebo model vypadá děravě.

**C-16 COLMAP Workspace Detection****KDE**

Při dropu složky — žádné viditelné tlačítko, nýbrž detekční logika.

**TECHNICKY**

Při dropu adresáře se kontroluje, zda v něm leží jeden ze tří kanonických workspace layoutů: `sparse/0/cameras.bin`, `sparse/cameras.bin` nebo přímo `cameras.bin` v rootu. Pokud sedí, standardní image enumerace se přeruší a místo toho otevírá modální alert, který ptá uživatele, zda použít existující rekonstrukci nebo obrázky znovu poslat přes Apple Photogrammetry. Stejná cesta i pro textové formátové workspaces (`cameras.txt`) a ETH3D experty. Viz Kapitola 9 backend Q6 pro detaily. Působí v Režimu pro začátečníky stejně jako v Expert Mode.

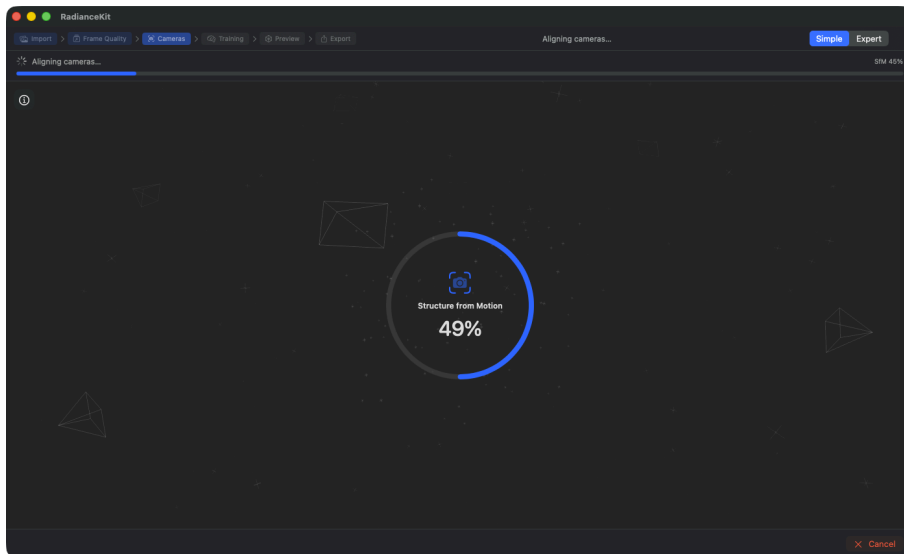
**JEDNODUŠE ŘEČENO**

Pokud jsi už pracoval s MetaShape, RealityCapture nebo COLMAP a tam jsi nechal výpočet kamer běžet, můžeš exportní složku jednoduše přetáhnout sem. RadianceKit podle obsahu automaticky pozná, že jde o COLMAP workspace (kontroluje `sparse/0/`, `cameras.bin` a spol.), a ptá se tě, zda převzít hotový výpočet nebo sám počítat znovu. Převzít šetří u velkých scén hodiny čekání, protože se SfM kompletně přeskakuje — trénink startuje okamžitě. Také textové formátové workspaces (`cameras.txt`) a ETH3D experty se rozpoznávají. Tato funkce je dostupná v Režimu pro začátečníky stejně jako v Expert Mode; více detailů stojí v Kapitole 9 pod backend Q6.

**Kdy do dalšího kroku?**

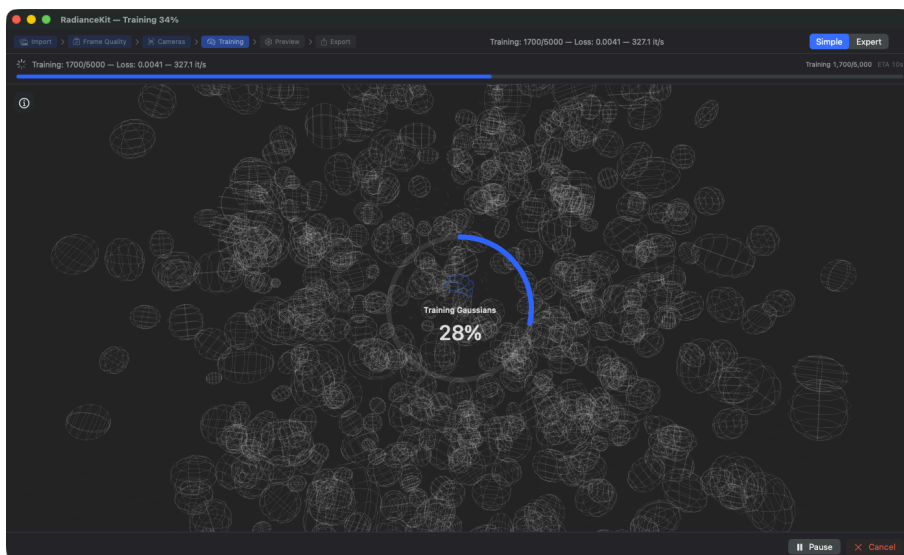
Můžeš stisknout Start Processing, jakmile (a) je importován alespoň jeden obrázek nebo video a (b) je validation banner oranžový nebo zmizel. Při červeném banneru tě aplikace sice nechá startovat, ale s vysokou pravděpodobností můžeš zpracování hned přerušit. Doporučeno: alespoň 20 obrázků, ostrých, s výrazným přesahem mezi po sobě následujícími záběry, všechny zhruba ze stejné vzdálenosti od motivu. Před startem zvol předvolbu, která sedí tvému časovému rozpočtu — u 30 obrázků a Quick předvolby jsi za pár minut hotov, u Quality to trvá spíše 1–2 hodiny.

## Z2 — Zpracování (SfM + trénink)



Obrázek 36: Z2 SfM fáze — stage ikona „Structure from Motion“, s 41 % ve velkém kruhu, horní statusová lišta na „SfM 25%“, cancel tlačítko vpravo dole

**SfM fáze (vyrovnávají se kamery):** Velký progress kruh ukazuje sub-stage postup (zde 41 % běžící Apple Photogrammetry session). Status text „Aligning cameras...“ vlevo nahoře. Crumb-trail značí „Cameras“ jako aktivní stupeň. Horní statusová lišta ukazuje celkový pipeline postup (25 %) — SfM zabírá první polovinu lišty. Plovoucí wire-frame kamery v pozadí naznačují, že se odhadují pozice.



Obrázek 37: Z2 tréninková fáze — stage ikona „Training Gaussians“, s 6 %, live metriky nahoře (Training: 400/5000 — Loss: 0.1642 — 138.7 it/s), ETA 33s, Pause/Cancel dole

**Tréninková fáze (Gaussiany se optimalizují):** Sub-stage ikona se mění na „Training Gaussians“, procento počítá iterace ze zvolené předvolby (zde 400 / 5 000 pro Preview předvolbu = 8 % stage). Live metric řádek ukazuje loss hodnotu (0.1642), iterace za sekundu (138.7 it/s) a ETA (33 s). Pipeline celkový postup šplhá z 50 % na 100 % během

této fáze. Pause tlačítko (místo Cancel-Only v SfM fázi) umožňuje Resume později; Cancel zahazuje tréninkový výsledek a vrací se na Z1.

Jakmile pipeline běží, aplikace skrývá Import overlay a zobrazuje celoplošnou zpracovací obrazovku. Uprostřed běží velký progress kruh (220 × 220 pixelů) se stage ikonou, status textem a procentem; v pozadí jemná splat animace symbolicky vizualizuje běžící výpočet. Vlevo nahoře lze zobrazit info panel, který ukazuje live metriky z tréninku a SfM. Dole jsou Pause/Resume, Cancel a v případě chyby Retry tlačítka.

### C-18 SplatTrainingView (animace na pozadí)



KDE

Celoplošné pozadí za progress kruhem, skryté při přerušení nebo chybě.



TECHNICKY

Dekorativní animace, která podle pipeline postupu (0...1) renderuje rostoucí počet malých animovaných splat částic. Zdrojem je vypočítaná postupová hodnota, která mapuje SfM fáze na 0–0.2 a trénink na 0.2–1.0 (Frame-Quality na 0–0.05). Splaty se tím viditelně „staví“, zatímco trénink běží. Výhradně dekorativní — zobrazení neukazuje skutečné mezi-výsledky aktuálního tréninku (to by byl Live Preview v Expert Mode). Při Cancel nebo Failure se skrývá a viditelný zůstává jen status kruh.



JEDNODUŠE ŘEČENO

V pozadí běží malá animace z tančících bodů, aby obrazovka během výpočtu nepůsobila tak prázdně. To není tvůj skutečný 3D model — ten uvidíš až po tréninku v kroku Z3. Animace má ale stejnou tonalitu, takže podle přibližného stupně zhuštění odečteš, jak daleko je trénink pokročilý. Zpočátku jsou viditelné jen málo bodů, ke konci se pozadí výrazně hustěji plní — pěkný vizuální indikátor navíc k procentu v kruhu. Pokud tě animace ruší (např. protože chceš v pozadí vedle pracovat), můžeš přepnout do Expert Mode, kde odpadá.

**C-19 Velký progress kruh**

Uprostřed zpracovací obrazovky, 220 × 220 pixelů.

**TECHNICKY**

Dva přes sebe renderované prstence: vně tlumený track ring, uvnitř vyplněný postup ring s akcentovým nebo červeným stroke (červený při chybě). Uvnitř kruhu stage ikona (mozek pro trénink, kamera pro SfM, film pro video frame extrakci, sparkles pro frame quality), stage titul a živě animované procento v 32 bodovém rounded fontu. Ikona jemně pulsuje, dokud je zpracování aktivní. Zobrazení interpoluje na 30Hz timeru jemně směrem k aktuálnímu skutečnému postupu — s constant-creep (0.0003/frame) plus proporčním podílem (4 % gapu) a soft-ceiling, který nastavuje na 80 % dalšího očekávaného milestone (pro SfM z napevno kódované milestone tabulky). Tak působí postup plynule, i když skutečné SfM updaty přicházejí jen každých pár sekund.

**JEDNODUŠE ŘEČENO**

Velký kruh uprostřed je tvá hlavní indikace, zatímco aplikace počítá. Plní se jemně, i když skutečné výpočtové updaty přicházejí jen každých pár sekund — to ti dává pocit, že se něco děje, místo abys minuty zírala na zamrzlé procento. Symbol uprostřed se mění podle toho, zda se právě extrahují framy (film ikona), vyrovnávají kamery (kamera ikona) nebo trénují Gaussiany (mozek ikona). Procento se vztahuje na aktuální dílčí krok — celkovou pipeline vidíš v úzké liště úplně nahoře. Při chybě se ring obarvuje červeně místo modré a ikona už nepulsuje, takže okamžitě poznáš, že se něco pokazilo.

**C-22 Info tlačítko (zobrazit metriky)**

Vlevo nahoře na zpracovací obrazovce, 32 × 32 pixelů.

**TECHNICKY**

Jednoduché tlačítko s material pozadím. Přepíná info panel zapnuto/vypnuto. Ikona se mění mezi info-kruh-outline a info-kruh-vyplněným, když je aktivní. Jemná zobrazovací animace. V tooltipu „Show detailed processing metrics“.

**JEDNODUŠE ŘEČENO**

Standardně je obrazovka záměrně uklizená — pouze velký progress kruh, víc zatím nevidíš. Pokud jako technicky zaměřený uživatel chceš přesněji vědět, co se děje (která iterace, jak vysoký loss, kolik Gaussianů), klikni na ikonu i vlevo nahoře. Malý panel se rozkládá dole a ukazuje všechny live hodnoty. Další klik ho opět skryje. Nastavení není persistentní — při každém novém tréninkovém běhu je panel nejprve opět skrytý, což je záměrně, aby začátečníky to nepoděsilo.

**C-23 Info panel (live metriky)**

Vlevo dole na zpracovací obrazovce, viditelné jen pokud `showProcessingInfo == true`.

**TECHNICKY**

Dvouloupcový panel s ultra-thin material pozadím. Levý sloupec: stage specifické info řádky — pro SfM status text a procento; pro trénink iterace, kombinovaný loss, L1 loss, D-SSIM loss, Gaussian count (oranžově obarveno), Speed (it/s), Elapsed time, vypočítaná ETA, SH degree a learning rate. Pravý sloupec: status text, time info string, inline loss chart (viz C-28) a discoverability nudge (viz C-32). Všechny hodnoty se čtou ze status tréninku, který se při každém tréninkovém ticku aktualizuje.

**JEDNODUŠE ŘEČENO**

Info panel ukazuje všechny live hodnoty, které by v Expert Mode trvale stály v inspector sidebaru: aktuální iterace, loss hodnota (menší = lepší), počet Gaussianů, rychlost, odhadovaný zbývající čas, SH degree a learning rate. Na pravé straně navíc běží malilinká loss křivka, která ti na první pohled prozradí, zda trénink jde správným směrem. Pokud trénink působí zaseknutě, pomáhá pohled sem — loss, který už neklesá, nebo ETA, která už neklesá, naznačují problémy. Pokud loss exploduje (náhle se stane obrovským) nebo ukazuje NaN, trénink se stal nestabilním a Cancel + Retry nebo přechod na jinou předvolbu má smysl.

**C-25 Pause/Resume tlačítko**

Spodní navigační lišta, viditelné pouze během tréninkové fáze (NE během SfM) a dokud zpracování běží.

**TECHNICKY**

Bordered tlačítko. Podle statusu volá Pause nebo Resume. Label se mění mezi „Pause„ (s pause ikonou) a „Resume“ (play ikona). Během SfM kroku se tlačítko nezobrazuje, protože Apple Photogrammetry nezná pause sémantiku. Pause stav uchovává iteraci, Gaussian status a optimizer momentum kompletně — Resume pokračuje tam, kde se předtím zastavilo.

**JEDNODUŠE ŘEČENO**

Zatímco trénink běží, můžeš ho kdykoli zastavit a později pokračovat. Smysluplné, pokud chceš mezitím dělat na Macu něco jiného, co potřebuje hodně GPU — např. video střih, herní test nebo render export z jiné aplikace. Klikni Pause, dělej svou věc, klikni Resume, trénink běží přesně tam, kde byl. Counter iterací, počet Gaussianů a optimizer momentum přitom zůstávají plně zachovány, pause state tě nestojí nic na kvalitě. Během SfM fáze není Pause dostupné — Apple Photogrammetry nezná funkci zastavení, tam musíš v nouzi pracovat s Cancel.

**C-26 Cancel tlačítko**

Spodní navigační lišta, viditelné dokud zpracování běží (SfM nebo trénink).

**TECHNICKY**

Červené bordered tlačítko. Otevírá potvrzovací dialog s titulem „Stop and discard progress?“, tlačítka „Discard Progress“ (destruktivní) a „Keep Running,“ (Cancel). Při potvrzení se nastaví cancel flag, tréninkový task se ukončí, SfM subprocess se v případě potřeby ukončí a do JSONL logu se zapíše summary řádek se status přerušení. Na rozdíl od Pause se tréninkové buffery a status zahazují.

**JEDNODUŠE ŘEČENO**

Knoflík přerušení. Na rozdíl od Pause je definitivní — pokud poté chceš nový start, zpracování běží od začátku, všechny už natrénované iterace jsou ztracené. Smysluplné, pokud ses spletl v předvolbě, trénink běží příliš pomalu nebo aplikace zjevně produkuje odpadkové výsledky a nechceš čekat. Před skutečným přerušením se aplikace ještě jednou ptá přes potvrzovací dialog, abys nešel z omylu o hodiny výpočetního času. Pokud chceš jen krátce přerušit, vezmi raději Pause.

**C-27 Retry tlačítko**

Spodní navigační lišta, viditelné pokud pipeline selhala (SfM status startuje s „SfM failed,“ nebo trénink je ve stavu chyby).

**TECHNICKY**

Akcentové tlačítko. Spouští celou pipeline znovu. Před startem se kontroluje, zda jsou ještě importovány obrázky/video. Předchozí chybové logy zůstávají v JSONL adresáři zachovány; nový run zapisuje nový log soubor s aktuálním timestampem.

**JEDNODUŠE ŘEČENO**

Pokud SfM nebo trénink přeruší s chybovou zprávou, můžeš to tady zkusit znovu. Občas to pomáhá, protože mnoho kroků (RANSAC, densifikace) má náhodnostní podíly a druhý pokus může být úspěšný, kde první selhal. Celá pipeline pak běží znovu od začátku — SfM a trénink, v čerstvém JSONL log souboru. Pokud i druhý pokus selže, většinou jsou problémem vstupní obrázky (příliš málo, příliš málo přesahu, motion blur, špatné světlo); pak se vrať s Back a vyměň materiál. Tip: paralelně podívej do tréninkových logů (Help → Open Training Logs), tam stojí podrobněji, kde to konkrétně zaseklo.

**C-28** Inline loss chart

V info panelu, pravý sloupec, viditelné pouze během tréninku s neprázdnou historií průběhu.

## TECHNICKY

Kompaktní kreslicí oblast (40 pixelů vysoká), kreslí loss history jako 1-pixel čáru v akcentové barvě. Data se filtrují na finite hodnoty (NaN ochrana pro nestabilní tréninky). Min/Max se počítají přes celou history — chart auto-zoomuje tedy na hodnotový rozsah. Poslední loss hodnota stojí vpravo nahoře nad chartem. History sama se v App stavu staví při každém tréninkovém ticku (typicky každých 100 iterací).

## JEDNODUŠE ŘEČENO

Malilinká loss křivka, která ti na první pohled ukazuje, zda trénink „konverguje“, (čára klesá doprava) nebo zda visí nebo exploduje (čára plochá nebo stoupá). U zdravého tréninku čára klesá zpočátku strmě a poté se vyploštuje — to je očekávaný průběh, podobný půlící křivce. Chart automaticky zoomuje na aktuální hodnotový rozsah, takže i malá zlepšení ke konci tréninku zůstávají viditelná. Pokud čára náhle vyletí nahoru nebo zamrzne, je to dobrý signál, že se něco děje špatně — buď je materiál problematický nebo by byla jiná předvolba lépe vhodná. Chart najdeš v info panelu, který nahoře vlevo otevíráš ikonou i.

**C-32** Discoverability nudge (Expert Mode hint)

V info panelu, pravý sloupec dole, viditelné pouze během tréninku A v Režimu pro začátečníky.

## TECHNICKY

Malý řádek s oko ikonou a caption textem „Switch to Expert Mode (⌘2) for live splat preview“, v zdrženlivém tónu a 10 bodovém písmu. Žádný interaktivní prvek, jen hint. Nereaguje na klik — uživatel musí skutečně stisknout Cmd+2 nebo kliknout menu Mode → Expert Mode.

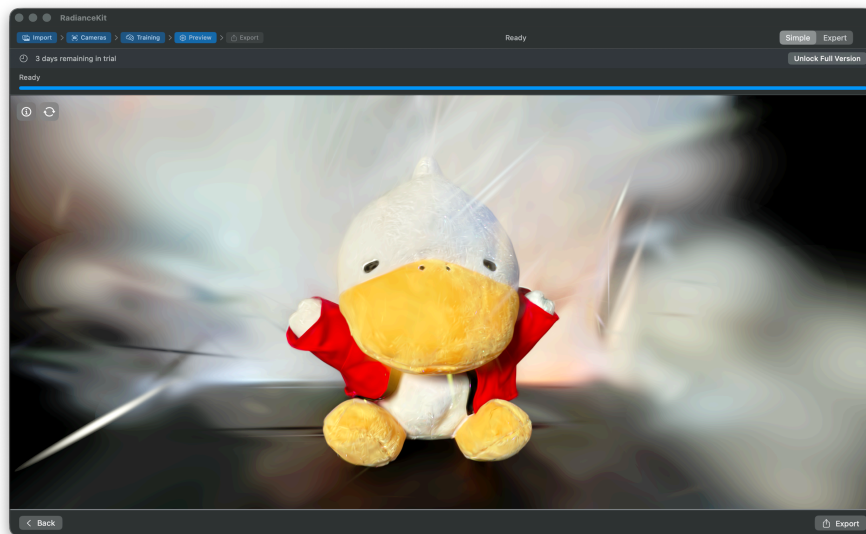
## JEDNODUŠE ŘEČENO

Decentní hint, že v Expert Mode je během tréninku aktuální mezi-verze tvého 3D modelu vidět živě v náhledu. V Režimu pro začátečníky se to záměrně skrývá, aby UI zůstalo klidné — ale mnoho uživatelů vůbec neví, že tato funkce existuje, takže zde jemně upozorňujeme. Stiskni Cmd+2 a trénink běží na pozadí dál, zatímco u toho můžeš sledovat, jak se tvůj model před tvýma očima skládá. To je také dobrý nástroj pro odhadnutí už po několika tisících iteracích, zda výsledek něco bude, nebo zda raději přerušit a začít znovu. Cmd+1 tě kdykoli vrátí do zobrazení pro začátečníky.

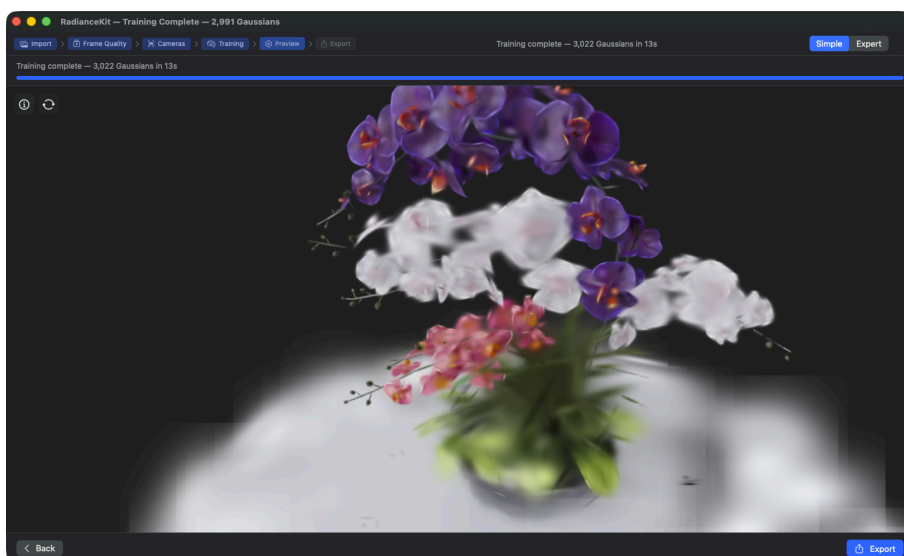
## Kdy do dalšího kroku?

Aplikace automaticky přepíná do Z3 (náhled), jakmile je trénink úspěšně dokončen — nemusíš nic klikat. Spodní navigační lišta se pak mění z Pause/Cancel na Back tlačítko (zpět na Import) a Export tlačítko (vpřed na Export). V případě chyby (červená chybová zpráva, stage ikona je X) se objevuje místo toho Retry a musíš se rozhodnout, zda startuješ znovu nebo se s Back vracíš na Import a měníš obrazový materiál.

## Z3 — Náhled (otáčet 3D modelem)



Obrázek 38: Simple Mode preview krok s 3D viewerem



Obrázek 39: Z3 náhled po dokončení tréninku — scéna rekonstruována, hlavička ukazuje „Training complete — 3,022 Gaussians in 13s,, Back a Export tlačítka dole

**CO JE NA OBRÁZKU** Crumb-trail značí „Preview,, jako aktivní stupeň. Celoplošný 3D náhled renderuje hotově natrénovanou scénu. Header status lišta: „Training complete — 3

022 Gaussians in 13 s“ — udává finální Gaussian count a tréninkový čas. Drag v náhledu rotuje kameru (Yaw/Pitch); scroll wheel zoomuje podél view direction. „Back,“ tlačítko (vlevo dole) vrací na Z2 pro Resume nebo re-run; „Export“ tlačítko (vpravo dole, primary) naviguje dál na Z4.

Po dokončení tréninku aplikace automaticky přistává v náhledu. Zde vidíš svůj hotový Gaussian Splatting model v fullscreen Metal náhledu a můžeš jej myší a trackpadem otáčet, zoomovat a posouvat. Na horní straně náhledu leží malý overlay s kamerovým ovládáním a info — auto rotace, tréninková statistika, reset knoflík. Před dalším krokem (Export) se nabízí, aby ses na model podíval z různých úhlů, abys se ujistil, že je rekonstrukce čistá.

### C-36 SplatViewportView (3D hlavní pohled)



Fullscreen pozadí kroku náhled.

#### TECHNICKY

Metal-based 3D náhled, který renderuje hotový mrak bodů. Renderer je vlastní ForwardPass rasterizér RadianceKitu — tentýž, který už zobrazuje splaty během tréninku — takže jde o opravdové WYSIWYG (co se trénuje, se přesně tak zobrazuje i exportuje). Tile-based rendering pipeline s Order Independent Transparency. Pokud renderer nelze inicializovat (např. protože Metal na systému není dostupný), objevuje se místo toho černé pozadí s textem „Metal not available,“. Pohled ignoruje safe-area, takže model sahá až k hraně okna.

#### JEDNODUŠE ŘEČENO

Hlavní viewport. Zde vidíš svůj hotový 3D model rekonstruovaný z tvých fotek, renderovaný na GPU v reálném čase. Klikni a táhni levou myší pro otáčení. Scroll kolečko nebo trackpad gesto dvěma prsty pro zoomování. Pravá myš nebo Cmd+drag pro posouvání. Model sestává z desítek tisíců semi-transparentních 3D elipsoidů („Gaussianů“), které tvoří scénu fotorealisticky rekonstruují — každý jeden má pozici, orientaci, tvar a barvu, které se trénink naučil. V vzácném případě, že tvůj Mac nepodporuje Metal, vidíš místo toho černé pozadí s hint zprávou — RadianceKit potřebuje nutně Metal-schopnou GPU.

**C-37** CameraControlsOverlay (ovládací overlay)

Nad náhledem, plovoucí.

**TECHNICKY**

Kompaktní UI overlay s tlačítky pro auto rotaci (turntable), reset camera, výběr pozadí (Gray/Black/White), save screenshot, toggle info panel. Váže se na parametry kamery (vzdálenost, azimut, elevace, target, FOV) a ovládá auto turntable. Během tréninku (pokud uživatel v Expert Mode chce vidět běžící náhled) overlay zobrazuje navíc kompaktní řádek tréninkového statusu.

**JEDNODUŠE ŘEČENO**

Malá plovoucí lišta nad modelem. Zde spouštíš auto rotaci (model se točí sám, dobré pro screenshoty a krátká demo), kameru přes reset vracíš na výchozí pozici (pokud ses ztratil), měníš pozadí (šedé pro neutrální, černé pro maximální kontrast, bílé pro světlé modely) a děláš přímo screenshoty, které se ukládají pod /Pictures. Praktické, pokud chceš ukázat určitý detail z přesně daného úhlu, aniž bys extra exportoval celý model. Auto rotace je také dobrý test, zda model vypadá ze všech stran stejně dobře, nebo zda existuje „špinavá strana“, která vznikla chybějícími záběry.

**C-38** Export tlačítko (navigační lišta)

Spodní navigační lišta v Z3.

**TECHNICKY**

Akcentové tlačítko s labelem „Export,“ a share ikonou. Klik spouští přepnutí na Z4. Předtím nadřazený pohled kontroluje, zda je plná verze odemčená — pokud ne, místo Export stage se zobrazuje uzamčený view (viz U-06).

**JEDNODUŠE ŘEČENO**

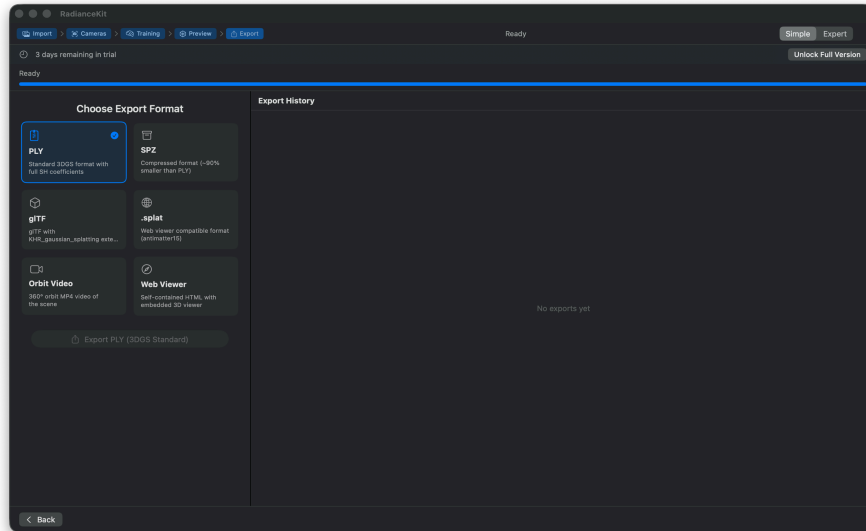
Pokud jsi s výsledkem spokojen, klikni Export a přistaneš v posledním kroku, kde vybíráš formát a ukládáš. Bez koupené plné verze přistaneš místo toho na obrazovkovém zámku s unlock hintem a tlačítkem koupit — aplikace ti plnou verzi nechce nasaďit, ale export je jednou z premium funkcí. Jakmile nákup dokončíš, aplikace běží přímo v odemčeném stavu dál a přistaneš v obvyklé Export stage. Pokud si to přesto rozmyslíš, přes Back tlačítko se vracíš do náhledu a můžeš dál otáčet modelem.

**Kdy do dalšího kroku?**

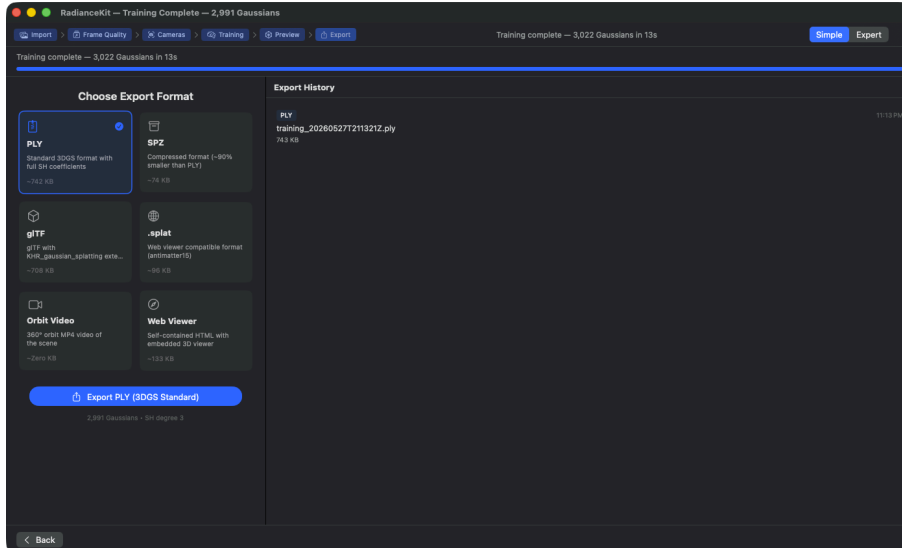
Před exportem model jednou kompletně otoč a zkontroluj: jsou přítomné všechny oblasti, které jsi pokryl ve vstupních obrazech? Existují plovoucí „floatery,“ (volně ve vzduchu visící Gauss splat mraky)? Působí pozadí/obloha čistě nebo rozmazaně? Závažné prob-

lémy se dají opravit pouze novým tréninkem — buď s více obrázky, jinou předvolbou, nebo v Expert Mode s Floater Reduction nastaveními.

## Z4 — Export (vybrat formát a uložit)



Obrázek 40: Simple Mode export krok s format kartami



Obrázek 41: Z4 export karty — 6 formátů (PLY 742 KB vybráno, SPZ 74 KB, gTf 708 KB, .splat 96 KB, Orbit Video, Web Viewer 133 KB), export history sidebar vpravo s již exportovaným PLY

**CO JE NA OBRÁZKU** Crumb-trail značí „Export,“ jako aktivní stupeň. Levé kartové mřížka „Choose Export Format“ se všemi šesti volbami: PLY (standardní 3DGS, 742 KB, s plnými SH koeficienty — zde předem vybráno s modrým zatržením), SPZ (komprimovaný 3DGS formát, ~90 % menší než PLY, 74 KB), gTf (s KHR\_gaussian\_splating rozšířením, 708 KB), .splat (kompatibilní s web viewerem přes antimatter15, 96 KB),

Orbit Video (360° MP4 scény, live výpočet velikosti), Web Viewer (samostatné HTML s vloženým 3D viewerem, 133 KB). Údaje o velikosti se počítají živě z aktuálního Gaussian countu a formátového overheadu. Vpravo „Export History“, listuje již dokončené exporty s formát pillou, názvem souboru a timestampem — klik reveals ve Finderu. Primary-CTA vlevo dole: „Export PLY (3DGS Standard)“ s Gaussian podtitulem „2,991 Gaussians · SH degree 3“.

V posledním kroku vybíráš z 6 export formátů (PLY, SPZ, glTF, .splat, Orbit Video, Web Viewer) přes dvousloupcovou kartovou mřížku, klikáš Export a vybíráš místo uložení v macOS dialogu. Vpravo běží history všech dosavadních exportů — při výběru karty se pod každou kartou okamžitě zobrazuje odhadovaná velikost souboru, takže např. preferuješ SPZ, pokud chceš na web (malé), a PLY, pokud chceš importovat do jiného softwaru (SuperSplat, Postshot, Blender přes plugin) (velké a kompletní).

### C-39 Dvousloupcová Format Grid



KDE

Levá hlavní strana kroku export.



TECHNICKY

Kartové rastr se dvěma flexibilními sloupci a 12 bodovým odstupem. Iteruje přes formáty nabízené v Režimu pro začátečníky — filtrovaná podmnožina plného seznamu formátů, která obsahuje jen 6 nejdůležitějších formátů: PLY, SPZ, glTF, .splat, Orbit Video, Web Viewer. Compressed-PLY a SOG se nabízejí POUZE v Expert Mode.



JEDNODUŠE ŘEČENO

Kartové rastr s 6 formáty, které jsou v Režimu pro začátečníky relevantní: PLY (standardní formát pro jiné 3D nástroje), SPZ (komprimovaná varianta pro web), glTF (oficiální Web3D standard), .splat (pro antimatter15 web viewer), Orbit Video (hotové MP4 k ukázkám) a Web Viewer (samostatný HTML soubor s vloženým 3D přehrávačem). Tím pokrýváš 90 % použití. Pokud potřebuješ jeden z méně běžných formátů (Compressed PLY nebo SOG pro extrémní kompresi), přepni do Expert Mode, tam je dostupných všech 8 formátů. Kompaktní výběr zde je záměrný, aby začátečníky nezahltila rozmanitost.

**C-40** Format Card tlačítko

Každá karta v gridu.

**TECHNICKY**

Jednoduché tlačítko s kartovým layoutem: ikona (např. document zipper pro PLY, archive box pro SPZ, video ikona pro orbit video) nahoře, format name jako headline, popisová caption (2 řádky zkráceno), pod tím odhadovaná velikost souboru (live počítaná z formátu, Gaussian countu a SH degree a formátovaná jako KB/MB). Při kliknutí se formát vybírá. Selektovaná karta dostává akcentové pozadí, akcentový border a zatržení ikonu vpravo nahoře. Tooltip je formátový popis.

**JEDNODUŠE ŘEČENO**

Karta na formát. Klikni jednu a označí se akcentovou barvou a zatržením, a export tlačítko pod tím přizpůsobuje svůj text („Export PLY“, „Export SPZ“ atd.). Každá karta ukazuje vhodný symbol, název, dvouřádkové krátké vysvětlení a odhadovanou velikost souboru u tvého aktuálního tréninkového výsledku. Velikost ti pomáhá smysluplně volit — pokud chceš výsledek poslat mailem, vezmi nejmenší variantu (obvykle SPZ nebo .splat); pokud chceš pokračovat v jiném 3D softwaru, vezmi tu s nejlepší kompatibilitou (typicky PLY). Při hoveru nad kartou tooltip ukazuje podrobnější popis, pokud ti přijde nejasný rozdíl mezi formáty.

**C-41** Video Duration slider

Pod format gridem, viditelné jen pokud je vybrán video formát (Orbit Video nebo Social Video).

**TECHNICKY**

Slider 3–30 sekund v 1sekundových krocích, váže se na délku videa v App stavu. Maximální šířka 300 pixelů. Zobrazuje se pouze pokud je vybrán video formát. U ne-video formátů se slider z pohledu úplně odstraňuje — žádné mrtvé místo.

**JEDNODUŠE ŘEČENO**

Pokud vybereš orbit video jako export, tady určuješ délku. 3 sekundy = velmi rychlé otáčení, 30 sekund = pomalé klidné otáčení kolem tvého modelu. Pro social media reels (Instagram, TikTok) je obvykle 6–10 sekund ideální — dost dlouhé na ukázání modelu, dost krátké aby diváci neodskákali. U prezentací nebo portfolio videí klidně 15–20 sekund. Slider se objevuje pouze pokud je vybrán video formát; u souborových formátů jako PLY nebo SPZ by byl nesmyslný a je skrytý.

**C-42 Export tlačítko**

Pod format gridem (a pod duration sliderem, pokud video vybráno).



Velké akcentové tlačítko. Label: „Export {Format-Name}“, share ikona. Při kliknutí se otevírá macOS save dialog s formátem odpovídající koncovkou a default filename „scene.{ext}“; při potvrzení se export píše na zvolenou URL. Deaktivováno, pokud není přítomen tréninkový výsledek nebo už běží export.

**JEDNODUŠE ŘEČENO**

Klikni, vyber místo uložení v macOS dialogu, hotovo — aplikace zapíše soubor ve zvoleném formátu na vybrané místo. Default název je „scene.{koncovka}“, (např. „scene.ply“ nebo „scene.spz“), který můžeš v dialogu libovolně změnit, než uložíš. Tlačítko je šedé, dokud není přítomen tréninkový výsledek (zde by se to nikdy nemělo stát, protože bys jinak vůbec nebyl v export kroku) nebo už běží jiný export. Jakmile export běží, pod ním se objevuje indikace postupu; aplikace zůstává ovladatelná, takže můžeš už připravovat další export.

**C-43 Export Progress Bar**

Pod export tlačítkem, viditelné jen dokud běží export.



Indikace postupu s maximální šířkou 300 pixelů, pod tím caption „Exporting... N %“. Hodnota běží od 0 do 1 a aktualizuje se během zápisu — u PLY v chunkách 10 000 Gaussianů, u SPZ jednorázově po kvantizaci, u Orbit Video v frame intervalech.

**JEDNODUŠE ŘEČENO**

Během exportu zde vidíš postup jako úzkou lištu plus procento. PLY je obvykle během sekund hotové, protože soubor se prostě binárně zapisuje. SPZ trvá trochu déle, protože data se přitom kvantizují a komprimují. Orbit Video je časově nejnáročnější export — zde se každý frame renderuje znovu; podle rozlišení a délky to může trvat minutu i déle. Během exportu aplikace zůstává ovladatelná, takže můžeš už připravovat další formát nebo v náhledu klikat dál.

**C-44** Export Error Display

Pod progress barem, viditelné pouze pokud při posledním exportu došlo k chybě.

## TECHNICKY

Červený řádek s warning ikonou a chybovým textem. Červená 8 % pozadí opacity, zaoblené rohy. Maximální šířka 400 pixelů. Časté příčiny chyb: SOG očekává `cwebp` v system PATH (ne App Store kompatibilní); chyby zápisu při plné kapacitě disku; sandbox chyby u cílů uložení mimo povolenou oblast.

## JEDNODUŠE ŘEČENO

Pokud export nevyjde, objeví se zde červeně krátký jasný popis problému. Většinou je příčina jasná — žádné místo na disku, žádná práva zápisu pro cílovou složku, nebo cíl mimo sandboxem povolené oblasti. Speciálně u SOG formátu se stává, že `cwebp` v systému chybí; v tomto případě není SOG použitelné a musíš se uchýlit k SPZ. Pokud je chybová zpráva nejasná, podívej se do log adresáře (Help → Open Training Logs), tam stojí podrobněji, co se pokazilo. V pochybnostech pomáhá prostě vybrat jiné místo uložení — např. plochu.

**C-46** Export History List

Pravá strana export kroku.

## TECHNICKY

Seznam přes export historii (persistentně uložená jako JSON v UserDefaults, udržovaná po každém úspěšném exportu). Každý řádek ukazuje format badge (malý, akcentové barvy), timestamp (HH:mm), název souboru (1 řádek zkrácen) a formátovanou velikost souboru. Klik na řádek otevírá Finder s selektovaným souborem. Empty state: „No exports yet“.

## JEDNODUŠE ŘEČENO

Seznam tvých dosavadních exportů — formát, čas, název souboru, velikost, v chronologickém pořadí. Klikni na řádek a soubor se ve Finderu zvýrazněně zobrazí, aniž bys musel sám navigovat složkami. Praktické, pokud hodinu později ještě jednou potřebuješ poslední export a už nevíš, kam jsi ho uložil — history si to pamatuje. Pokud jsi nikdy nic neexportoval, stojí zde přátelské hint „No exports yet“. Seznam přežívá restart aplikace, protože je uložen v UserDefaults.

## C-48 History kontextové menu (pravý klik)



Pravý klik na history řádek.



Kontextové menu na každé položce seznamu se dvěma akcemi: „Reveal in Finder„ (otevívá Finder s selektovaným souborem, jako jednoduchý klik) a „Copy Path“ (kladá kompletní cestu souboru jako text do schránky). Druhé je užitečné pro drag-and-drop do jiných aplikací nebo pro předání na příkazovou řádku.

### JEDNODUŠE ŘEČENO

Pravý klik na history záznam otevírá malé menu se dvěma akcemi. „Reveal in Finder„ dělá totéž jako normální klik — otevírá Finder se selektovaným souborem, takže ho okamžitě vidíš. „Copy Path“ pokládá kompletní cestu souboru do schránky, takže ji můžeš vložit např. v Terminal příkazech, do jiných aplikací nebo do poznámky. Obzvláště praktické, pokud chceš export předat někomu jinému nebo ho otevřít v jiném programu, který pracuje s zadáním cesty. Funkčně malý, ale užitečný detail, který sází na macOS typické ovládací vzory.

## Kdy je workflow ukončen?

Po úspěšném exportu máš svůj 3D model jako soubor na disku a history ukazuje nový záznam. Není „Done„ tlačítko — můžeš libovolně mnoho exportů v různých formátech připojit, aniž bys musel znovu trénovat. Pokud chceš zpět do náhledu (např. abys znovu ověřil perspektivu kamery), použij Back tlačítko ve spodní navigační liště. Pokud chceš začít úplně novou scénu, jdi přes Back až k Z1 a tam použij Clear All, nebo File → New Project (Cmd+⇧+N).

## Přepnutí do Expert Mode

Stiskni kdykoli Cmd+2 nebo vyber Mode → Expert Mode ( M8 ). Celý stav zůstává zachován: importované obrázky, zvolená předvolba, běžící nebo hotový trénink, hotový mrak bodů, export history, dokonce aktuální stage. V Expert Mode se místo čtyřkrokové stage zobrazuje plný inspector sidebar se všemi ~150 ovládacími poli. Zejména: Project Navigator (viz Kapitulu 2) nabízí rozšířené obrazové operace (minus tlačítko, Backspace-delete, Cmd-Z-undo, Quick Look náhled), live preview v náhledu během tréninku, a všechny loss, MCMC, densifikační a Mip-Splatting parametry. Cmd+1 přepíná zpět do Režimu pro začátečníky — také to neztrácí stav.

## Časté otázky

**Proč zůstává mé Start Processing tlačítko šedé?**

Ještě jsi neimportoval obrázky ani video. Přetáhni alespoň jeden soubor do drop zone nebo použij „Browse Files“. Jakmile seznam obrázků vpravo obsahuje alespoň jeden záznam, tlačítko se stává aktivním. (Při jen 1–2 obrázcích sice startuje, ale SfM přímo přeruší s chybou — viz červený validation banner.)

### **Proč je mé Export tlačítko uzamčeno?**

V Režimu pro začátečníky jsou dva stupně: (a) Pokud tréninková pipeline ještě není hotová a nemáš mrak bodů, tlačítko je deaktivováno — musíš nejprve dokončit Z2. (b) Pokud jsi plnou verzi ještě nekoupil (`PurchaseManager.hasAccess == false`), vidíš místo export stage uzamykací view s ikonou zámku a tlačítkem „Unlock Full Version“, které otevírá Purchase Sheet. Quick a Preview předvolby umožňují trénink zdarma, ale export je premium.

### **Proč nemohu vybrat předvolbu?**

Můžeš vybrat — ale pokud klikneš na premium předvolbu (Balanced, Quality, MCMC varianty) bez koupené plné verze, picker automaticky skáče zpět na Preview a Purchase Sheet se otevírá. Quick a Preview jsou jediné zdarma použitelné předvolby.

### **Proč je má drop zone prázdná a přerušovaně-šedá, i když do ní táhnu obrázky?**

Pravděpodobně UTI type mismatch. Aplikace akceptuje JPG, PNG, TIFF, HEIC, MP4, MOV plus aplikací vlastní splat formáty. Jiné obrazové formáty (BMP, GIF, WebP, RAW formáty) se NEROZPOZNÁVAJÍ. Pokud jsi si jistý, že tvůj typ obrazu by měl být obsažen, zkontroluj koncovku názvu souboru — aplikace jde primárně podle extension, ne podle obsahu souboru.

### **Proč SfM trvá tak dlouho, i když mám jen 30 obrázků?**

Apple Photogrammetry neškáluje lineárně — u některých konstelací obrazu (interiéry s komplexními texturami, motion blur, špatné světlo) potřebuje výrazně déle, než počet obrázků naznačuje. Pokud SfM po 10+ minut u 30 obrázků stále visí, přeruš a zkus to znovu s lepším materiálem, nebo přepni do Expert Mode a vyzkoušej COLMAP/Native SfM (`Cmd+2` → Inspector → Camera Alignment).

### **Kde najdu své tréninkové logy?**

Help → Open Training Logs (`Cmd+⇧+L`). To otevírá `~/Documents/RadianceKit/Logs/`. Každá tréninková session zapisuje vlastní JSONL soubor s timestampem v názvu souboru — první řádek je konfigurace, poté následuje progress řádek každých 100 iterací, poslední řádek je summary s final loss a success flagem.



## TIRÁŽ

*Vysazeno písmem SF Pro · Kód v SF Mono · Typst  
0.14 · 22. June 2026*

© 2026 Bjoern Kindler · Bischofshofener Str. 9, 82008 Unterhaching, Německo

Vytvořeno s ❤ v Unterhachingu