



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ELEKTRONOVÝ RAYTRACER

ELECTRON RAYTRACER

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JAN SUCHÁNEK

VEDOUCÍ PRÁCE

SUPERVISOR

Doc. Ing. MARTIN ČADÍK, Ph.D.

BRNO 2018

Zadání diplomové práce



21460

Student: **Suchánek Jan, Bc.**
Program: Informační technologie Obor: Počítačová grafika a multimédia
Název: **Elektronový ray tracer**
Ray Tracing for Electron Microscope
Kategorie: Počítačová grafika

Zadání:

1. Seznamte se s algoritmy sledování paprsků (ray tracing) s důrazem na nelineární metody.
2. Vytipujte metodu vhodnou pro simulaci sledování paprsků v elektronovém mikroskopu a popište její vlastnosti.
3. Navrhněte a implementujte systém pro simulaci sledování paprsků v elektronovém mikroskopu.
4. S navrženým systémem experimentujte a diskutujte možnosti budoucího vývoje.

Literatura:

- dodá vedoucí práce

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3 zadání

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Čadík Martin, doc. Ing., Ph.D.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 22. května 2019

Datum schválení: 2. listopadu 2018

Abstrakt

Tato práce se věnuje simulaci elektronového mikroskopu. Spojuje poznatky z metod sledování paprsků a realistického zobrazování s fyzikálními poznatky z oblasti elektronové mikroskopie. Cílem práce je vytvořit funkční a co nejvěrnější simulátor skenovacího elektronového mikroskopu s kvalitním výstupem v úměrně rychlém čase. Přínos práce spočívá v získání dostatečného množství obrazového materiálu pro další zpracování, především v oblasti strojového učení, ale i pro usnadnění vývoje aplikací mikroskopu pro post-processing. V současné době se obrázky získávají z reálného systému a ručně anotují.

Abstract

This thesis is pointed on simulation of electron microscope. It uses knowledge from ray-tracing methods and physically based rendering with physics of electron microscopy. The main object of this thesis is to create realistic simulator for electron microscope which can generate satisfying realistic images. This output can be used for another research in range of machine learning or microscope application development. This method should replace long and more expensive way of getting test images from real system for those research activities.

Klíčová slova

Elektronový ray tracer, sledování paprsku, Elektronový mikroskop, simulátor, PBRT, Elektronová mikroskopie

Keywords

Electron ray tracing, ray-tracing, Electron microscope, Electron simulation, PBRT, Electron microscopy

Citace

SUCHÁNEK, Jan. *Elektronový raytracer*. Brno, 2018. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Doc. Ing. Martin Čadík, Ph.D.

Elektronový raytracer

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana docenta Martina Čadíka. Další informace mi poskytli Lukáš Hubner a Libor Novák. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jan Suchánek
21. května 2019

Poděkování

Tímto bych chtěl poděkovat vedoucímu práce panu docentu Martinu Čadíkovi za trpělivost a pomoc při vedení diplomové práce. Dále pak pánům Lukáši Hübnerovi a Liboru Novákovi za cenné rady a informace o elektronové mikroskopii a přiblížení funkcionality skenovacích elektronových mikroskopů

Obsah

1	Úvod	3
1.1	Motivace	3
1.2	Skenovací elektronový mikroskop	3
1.3	Transmisní elektronový mikroskop	4
2	Skenovací elektronový mikroskop	6
2.1	Komponenty mikroskopu	6
2.2	Princip elektronového mikroskopu	9
2.2.1	Sekundární elektrony	10
2.2.2	Zpětně rozptýlené elektrony	12
2.2.3	Interakce eletronu s povrchem	13
3	Realistické zobrazování	15
3.1	Existující implementace	15
3.2	Hardwarová akcelerace výpočtů	17
3.3	Sledování nelineárních paprsků	18
3.4	PBR _{TV3}	18
3.4.1	Komponenty	19
3.4.2	Upravitelnost PBR _{TV3}	21
4	Návrh aplikace a implementace	22
4.1	Analogie mezi raytracingem a simulací	22
4.2	Simulátor	22
4.2.1	Abstrakce	22
4.3	Simulace	25
4.4	Implementace	26
4.5	Ukázka výstupu	28
5	Testování a srovnání	29
5.1	Návrh testování	29
5.2	Testovací data	29
5.2.1	Metriky pro srovnání výstupů	30
5.3	Srovnání	31
5.3.1	PathTracing a ElektronTracing	31
5.3.2	Reálný systém a ElectronTracing	31
5.3.3	Vliv počtu vzorků na pixel	40
5.4	Výkon aplikace	41

6 Závěr	43
6.1 Vlastnosti implementovaného rozšíření systému PBRTv3	43
6.2 Hodnocení práce ve firmě	44
Literatura	45
A Obsah přiloženého paměťového CD	47
B Manuál	48
B.1 Instalace a sestavení	48
B.2 Použití	48
B.2.1 Tvorba modelu	48
B.2.2 spuštění	48
C Ukázka vstupního souboru	49
D Obrazové přílohy	51
D.1 Systém TLD	51
D.2 Systém ICE	53
D.3 Systém ETD	55

Kapitola 1

Úvod

V rámci této diplomové práce se zabývám tématem elektronové mikroskopie a její simulací v co nejvěrnější formě a v co nejrychlejší čas. Přínos práce by měl být právě v možnosti nasimulování obrazových dat vycházejících z mikroskopu. V moderních metodách zpracování obrazu se totiž často pro detekci objektů, rozpoznávání a pojmenování i v případě obarvení, využívá stále častěji neuronových sítí a strojového učení. Pro natrénování neuronových sítí je potřeba mít dostatečně velkou zásobu vhodných anotovaných dat, která lze v případě elektronové mikroskopie jen obtížně získat.

Získávání obrázků z reálného mikroskopového systému obnáší spoustu úkonů a nezbytných nastavení, které se navíc liší podle různých druhů mikroskopových komponent. V simulaci lze toto zjednodušit na pouhé zadávání parametrů programu – simulátoru.

Další výhodou simulace je pak možnost jednoznačně určit a tím i částečně automatizovat anotaci výsledného obrázku a tím zjednodušit a urychlit tvorbu datových sad pro trénování neuronových sítí.

1.1 Motivace

Elektronový mikroskop má široké spektrum využití. Používá se jednak v bio-medicínské oblasti, jednak v materiálovém inženýrstvím. Detaily o molekulární struktuře materiálu se také hodí pro nanotechnologie. Nemalou průmyslovou oblastí, vlastně největší, kde se mikroskopy používají, je výroba polovodičových součástek.

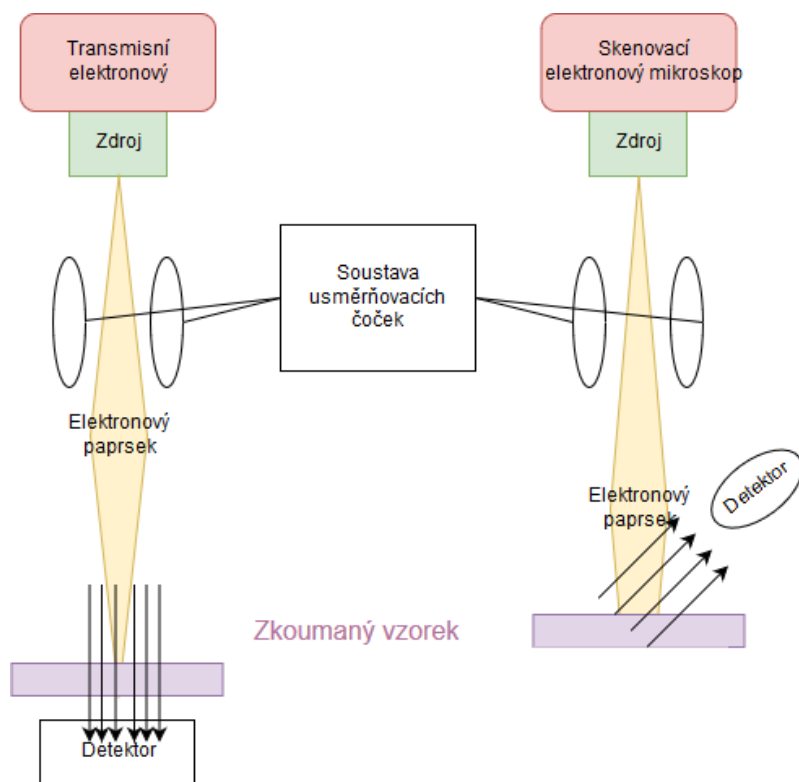
V obou oblastech nachází využití dva základní typy elektronových mikroskopů. Typy se liší způsobem snímání, rychlostí, pořizovací cenou a v neposlední řadě také možným zvětšením a dostupným rozlišením.

1.2 Skenovací elektronový mikroskop

Hlavní část elektronového mikroskopu je urychlovací trubice, v angličtině zvaná Gun, odvozeno od toho, že zařízení „střílí“ elektronový paprsek směrem ke zkoumanému vzorku. Střílení je v úvozkách, protože ve skutečnosti dochází k urychlení toku elektronů ze zhaiveného vlákna, obdobně jako v případě obyčejné žárovky, pomocí cívek a jeho usměrnění na požadované souřadnice pomocí vychylovacích cívek. Princip se velice podobá starým CRT obrazovkám.

Skenování pak probíhá pixel po pixelu, kdy určitou dobu svítíme na jedno místo vzorku, toto místo pak odpovídá právě jednomu pixelu ve výsledném rastru. Intenzita výsledného

pixelu je dána počtem elektronů zachycených v detektoru. Čím více elektronů bylo zaznamenáno, tím je daný pixel světlější. Rozdíl mezi skenovacím a transmisním elektronovým mikroskopem je vidět na obrázku 1.1.

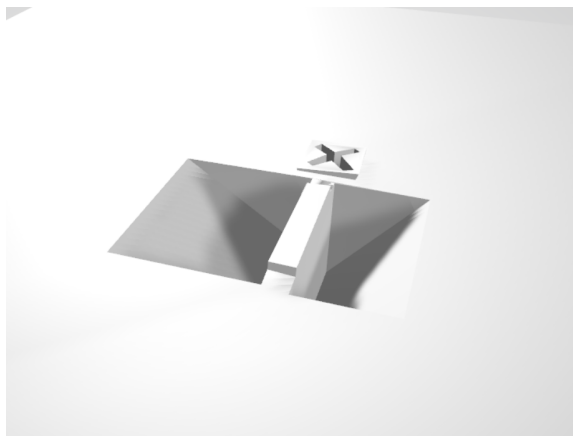


Obrázek 1.1: Princip elektronových mikroskopů

1.3 Transmisní elektronový mikroskop

Transmisní elektronový mikroskop je dražší a náročnější varianta elektronového mikroskopu. Oproti skenovacím mikroskopům přináší výhody v podobě možnosti detailnějšího a většího přiblížení vzorku. To si ale vybírá svou „daň“. Kromě již zmíněné vyšší ceny se jedná i o relativně velký kus hardwaru, mikroskop by se dal přirovnat k šatní skříni. Kromě váhy, která je podstatně vyšší.

Další a podstatnější nevýhoda spočívá v potřebné předpřípravě zkoumaných materiálů a vzorků. Zkoumaný preparát totiž musí být velice tenký, aby se dal elektronovým paprskem „prosvítit“. Proces přípravy mívá různé podoby, většinou se ale využívá skenovací elektronový mikroskop, kdy se zdroj elektronů vymění za iontový zdroj. Tímto mikroskopem se pak na konkrétní místa mohou vypalovat díry, fyzicky vyrazit a přesunout molekuly vzorku. Připravený vzorek může pro ilustraci vypadat jako na obrázku 1.2.



Obrázek 1.2: 3D model preparátu.

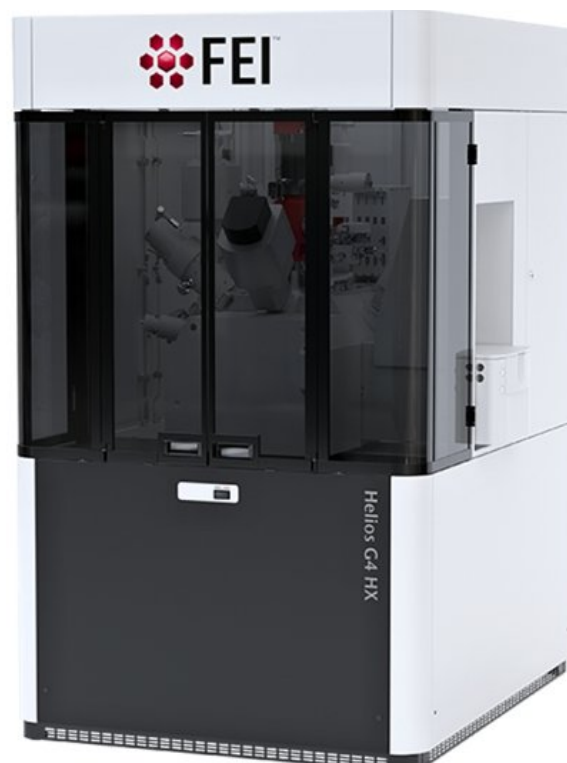
Přepážka v prohlubni je budoucí vzorek pro TEM mikroskop. Křížek na povrchu je pomocný prvek pro detekci obrazu, aby se mohlo provést strojové vylomení vzorku z materiálu. Křížek bývá z jiného materiálu, čímž se na snímku odliší i intenzitou, z důvodu dalšího usnadnění detekce daného vzorku.

Kapitola 2

Skenovací elektronový mikroskop

V této kapitole bych rád představil skenovací elektronový mikroskop. V sekci 2.1 jsou popsány jednotlivé komponenty, v sekci 2.2 je popsána princip a fungování elektronového mikroskopu. Následující sekce pak přibližují detailněji vlastnosti elektronových paprsků uvnitř systému. Na základě informací obsažených v této kapitole je pak navržen a implementován jednoduchý simulátor.

2.1 Komponenty mikroskopu



Obrázek 2.1: Fotografie mikroskopu

Elektronový mikroskop je zařízení v zásadě jednoduché. Když se podíváme do historie, první elektronový mikroskop byl sestaven v roce 1931. Sice se jednalo o transmisní systém,

ale od něj už zbývalo jen málo k sestavení skenovacího elektronového mikroskopu, který je někdy zván jako rastrovací.

V případě transmisního elektronu se jednalo o využití znalostí z různých oblastí fyziky a jejich využití za jedním účelem. V následujících podsekcích jsou tedy popsány jednotlivé komponenty, které stojí za vznikem výsledného obrázku.

Zdroj napětí

Pro tvorbu elektronového paprsku je nutné mít dostatečně silný a stabilní zdroj elektrického napětí. Generátor elektronů totiž funguje na podobném principu jako klasická žárovka, tedy paprsek elektronů se generuje žhavením wolframového vlákna nebo vlákna z jiného materiálu o podobných vlastnostech.

Alternativně lze použít i novější způsob generování elektronů. Namísto žhaveného vlákna se instaluje zdroj s hrotem z wolframu nebo jiného, opět chemicky podobného prvku, tento typ zdroje je znám pod zkratkou FEG z anglického názvu „Field Emission Gun“.

Zdroj elektronů

V angličtině se nazývá jednoduše „gun“ a jedná se o soustavu elektromagnetických čoček pro usměrnění toku elektronového paprsku z žhaveného vlákna nebo formovaného hrotu, směrem ke zkoumanému vzorku.

Elektronová čočka je cívka, která generuje optimální elektromagnetické pole. Optimální myšleno ve smyslu použití pro usměrnění elektronového paprsku a jeho zaostření na požadované místo. Pomocí soustavy takových čoček-cívek lze docílit různého zaostření stejně tak i pohybovat paprskem podle potřeby v definovaném rastru.

Kromě zbraně emitující elektrony existuje i zbraň, která emituje větší a těžší ionty. Tato varianta se používá k „řezání“ a tvarování zkoumaného materiálu. Praktické využití se naskýtá například při tvorbě preparátů pro transmisní elektronový mikroskop.

Stolek

Důležitý a mechanicky složitý prvek sloužící pro držení vzorku. Umožňuje pochyb ve všech směrech os x, y a z. Stejně tak jako rotaci. Pohyby jsou prováděny s velice velkou přesností v řádech mikrometrů a maximální stabilitou.

Stolky kromě pohybu umožňují i aplikačně specifické operace, na míru připravené různých výzkumným odvětvím. V případě biologie se jedná například o chlazení, či úplné zmrazení vzorků, kdy je potřeba zachovat buněčnou integritu vzorku a voda v kapalném skupenství by přinesla destrukci vzorku. Na druhou stranu se využívá i zahřívání, kdy pro materiálové inženýrství je zajímavé a důležité vědět, jak se vzorek chová při vyšších teplotách (zahřívání mikročipu a podobně).

Detektor

Zařízení pro zachytávání elektronů přicházejících ze vzorku. Zajímavostí detektorů je, že generují elektromagnetické pole, kterým k sobě elektrony přitahují. Protože rozlišujeme dva druhy elektronů, které při dopadu paprsku vznikají a každý druh má značně odlišnou energii (udávána v elektronvoltech, nepatří do soustavy jednotek SI), potřebujeme volit sílu elektromagnetického pole podle toho, který druh elektronů chceme sledovat.

Zvláštní případy nastávají u systémů s více detektory. Zřídka kdy dochází k situaci, kdy se využívají skutečně oba zaráz, ale i možnost volně mezi nimi přepínat přináší zajímavé výsledky.

Při snímání má detektor dáno, po jakou dobu má pro daný bod vzorku detekovat elektrony. Pro odstínění šumu také obsahuje práh, pod který přijaté elektrony neviduje. Detektory také přijímají různé odražené elektrony z ostatních bodů vzorku. K odrazům dochází i od samotného zařízení mikroskopu – například přímo ze zbraně, vsunuté kamery, případně druhého detektoru, je-li přítomen.

Kamera

Elektronový mikroskop bývá vybaven optickou kamerou, která snímá vnitřní prostor systému. Lze jeho pomocí vizuálně sledovat polohu vzorku a kontrolovat jeho stav. V mikroskopu bývá zpravidla kamer více, pro možnost snímání vnitřního prostoru mikroskopu z více úhlů a tím získání dalších informací o procesu snímání. Vizuální kontrola prostoru by bez kamer byla velice obtížná z důvodu, že uvnitř systému je potřeba udržovat poměrně kvalitní vzduchoprázdno.

Vakuová pumpa

Jak už bylo zmíněno u kamer, v elektronovém mikroskopu je vytvořeno a udržováno kvalitní vakuum. Z toho důvodu je součástí mikroskopu i vakuumová pumpa, respektive systém pump různých vlastností, které jsou vhodně kombinované, aby dobře pracovaly jako celek.

Čistota vnitřního prostoru je pro konečný výstup mikroskopu podstatná veličina, ale vzduchoprázdno uvnitř hraje roli i v celkové funkčnosti. Bez vakua by zkrátka mikroskop nemohl fungovat z fyzikálního hlediska.

Pro zdroj elektronů je podstatné, aby v místě kde dochází ke žhavení wolframového vlákna, nebyl přítomen kyslík. Důvod je prostý, podobně jako u žárovky jde o zamezení hoření, které by v tomto případě způsobilo daleko větší škody, než u žárovky. V komoře pro zkoumané materiály je vakuum také důležité. Zde hlavně z důvodu čistoty výsledného snímku. Jakákoliv nečistota, sebemenší částička prachu v atmosféře, by byla příčinou v lepším případě zašumění výstupu, v horším případě by došlo ke kompletnímu zastínění vzorku touto nepatřičnou nečistotou.

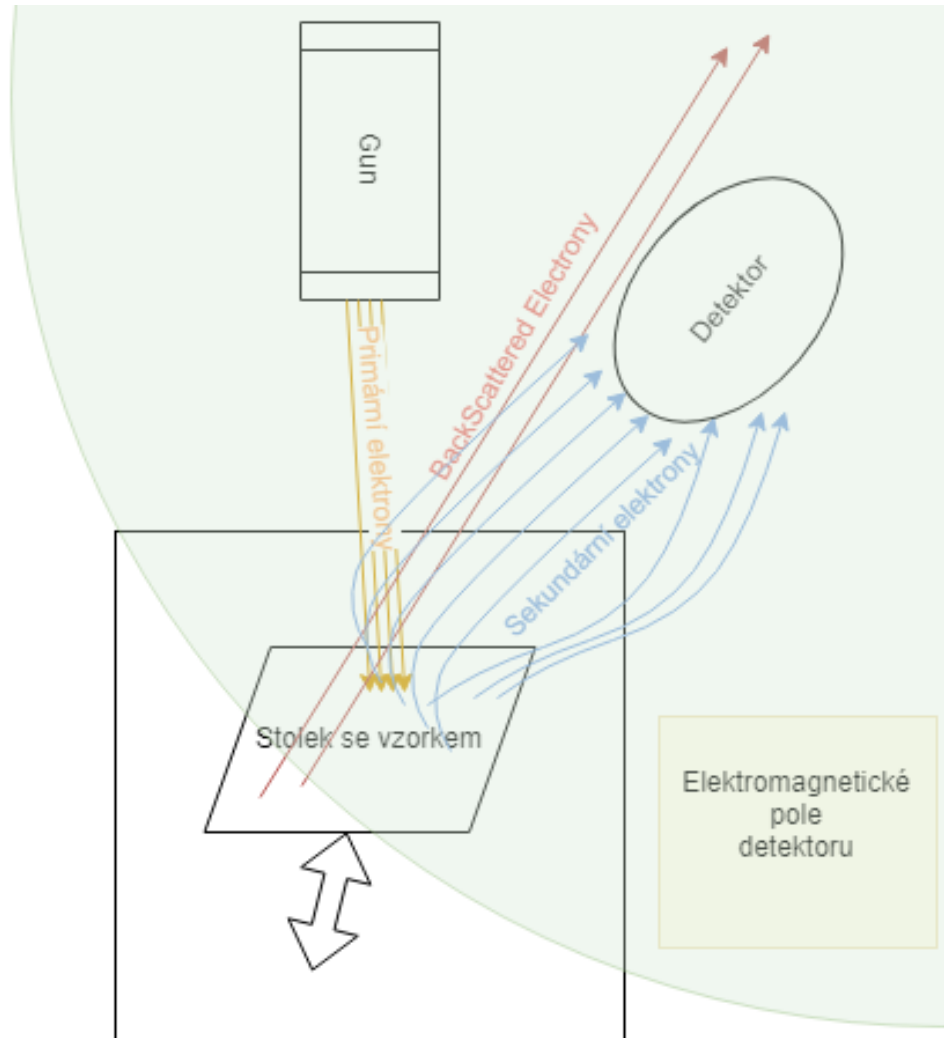
Řídící PC

Všechny komponenty mikroskopu je potřeba nějakým způsobem ovládat. Většinou se spolu s mikroskopem, jako součást dodává vlastní nové PC, které je ve správě dodavatele mikroskopu, čímž je zaručena jistá stabilita zařízení.

Počítač má na starosti vlastní ovládání systému, kontroluje všechny součásti mikroskopu a umožňuje s nimi provádět požadované akce. Počítač bývá odpojen od internetu, s jednotlivými komponentami mikroskopu komunikuje pomocí ethernetové sítě, která poskytuje dostatečnou rychlost spojení a stabilitu.

2.2 Princip elektronového mikroskopu

V této sekci je detailněji popsán způsob získávání obrazu z elektronového mikroskopu. Je ukázáno, jak spolu jednotlivé komponenty zmíněné výše souvisí a k čemu vlastně slouží.



Obrázek 2.2: Schéma mikroskopu

Rastrovací elektronový mikroskop získává informace o prvku pomocí elektronových paprsků. Na výsledný obrázek má vliv několik parametrů, před každým snímáním si operátor určuje, jakého přiblížení chce dosáhnout, dále se nastavuje doba, po kterou jsou paprsky vysílány na jednu konkrétní plošku, což zásadně ovlivňuje intenzitu výsledného pixelu nebo skupiny pixelů, pokud je nastavena větší oblast dopadu paprsku. Je důležité vědět, že snímání probíhá sekvenčně, bod po bodu, řádek po řádku. Lze přirovnat ke staré technologii CRT televizorů, kde pomocí elektrod je vychylován paprsek světelného zdroje do rastru obrazovky.

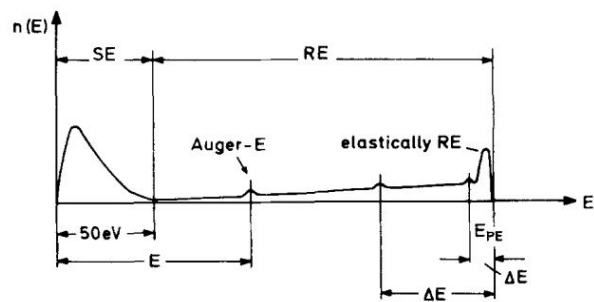
Proces snímání je možné sledovat živě v operátorské aplikaci, případně i za chodu aplikace, v takovém případě pak dojde k resetu vykreslování a vše započne od začátku, s novými parametry.

Obdobně jako v CRT monitorech dochází tedy ke generování paprsku, v případě elektronového mikroskopu se jedná o paprsek uvolněných elektronů a ty jsou soustavou elektromagnetických cívek soustředěny na požadovaný bod. Interakcí mezi paprskem a materiálem dochází k nabití místa dopadu energií, která se hromadí a způsobí, že část elektronů je emitována zpět. V zásadě rozlišujeme dva druhy elektronů, „odražených“ po dopadu primárního paprsku ze zdroje. Jsou to **sekundární elektrony (SE)** a **zpětně rozptýlené elektrony (BSE - BackScattered Electrons)**. Oba druhy mají významný vliv na podobu výsledného obrazu, je tedy nutné co nejpřesněji popsat jejich trajektorie a vlastnosti. Oba druhy elektronů jsou popsány v následujících podsekcích [2.2.1](#) a [2.2.2](#).

Oba druhy emitovaných SE a BSE pak v mikroskopu detekuje detektor, který můžeme popsat akceptační funkcí. Tato funkce popisuje, jaké elektrony o jaké energii je schopen detekovat. Samotná detekce probíhá tak, že po předem specifikovanou dobu snímání (byla nastavena už dříve v uživatelské aplikaci) detektor počítá, kolik elektronů ho zasáhne. Dochází zde k nějakému prahování, jsou přijaty elektrony, které překročí nějaký práh, od kterého detektor považuje zásah za validní. Výsledný čítač po uplynulé době pak určuje intenzitu výsledného pixelu v obrázku. Po dokončeném snímání jednoho bodu, případně oblasti, se zdrojový paprsek přechýlí na novou pozici a vše se opakuje, dokud není sejmut celý rastr. Zjednodušené schéma je k vidění na obrázku č. [2.2](#). Z funkcionality vyplývá, že celý proces snímání je relativně časově náročný. Například pro sejmutí rastru o rozlišení 800x600 bodů a při době snímání 1ms na jeden obrazový bod, se dostaneme na dobu pořízení tohoto obrázku 8 minut za předpokladu, že zanedbáme režii přesunu paprsku z bodu na bod následující a dalších nutných operací. Urychlení v současné době moc není reálné, mikroskopy už jsou tak relativně drahá zařízení a prosté použití dvou zdrojů a dvou detektorů by neumožnilo paralelní snímání dvou oblastí, protože by se navzájem rušily. Jediný přínos by mohl být pro překryv režijních časů při přesunutí paprsku na další místo snímání, což by se finančně pravděpodobně nevyplatilo. V tento moment má smysl simulace, která umožní paralelismus a v případě dost realistického výstupu by mohla být ušetřena spousta času i prostředků.

2.2.1 Sekundární elektrony

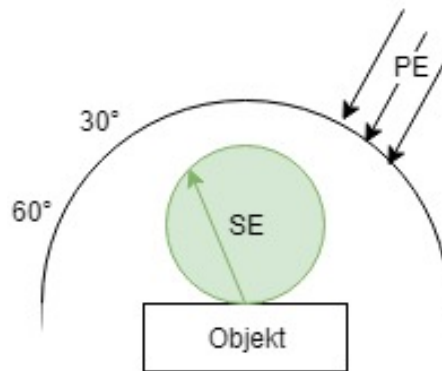
Tato podsekcce vychází z článku [\[12\]](#) a poskytuje informace o sekundárních elektronech a jejich experimentálně zjištěných vlastnostech. Sekundární elektrony jsou elektrony vemitované po dopadu primárního paprsku na povrch. Jsou charakterizovány nižší energií. Experimentálně bylo zjištěno, že pro primární elektronový paprsek s energií 100eV jsou sekundární elektrony emitovány s energií do 50eV. Energie těchto sekundárních elektronů pak odpovídá rozložení na obrázku [2.3](#). Bylo zjištěno, že v podstatě nezáleží na energii primárního paprsku. Jediná závislost, kterou můžeme v naměřených hodnotách pozorovat je na druhu materiálu. Pro kovové, vodivé prvky dosahuje energie sekundárních elektronů vyšších hodnot, než pro izolanty.



Obrázek 2.3: Energie SE, převzato z [12]

Směr emise

Úhlová distribuce paprsků sekundárních elektronů kopíruje kosinovou distribuci. Je podstatné vědět, že směr emise nijak nezávisí na úhlu dopadu primárního elektronu, jak je znázorněno na obrázku 2.4



Obrázek 2.4: Úhlová distribuce vyzáření SE, převzato z [12]

Úhrn emise sekundárních elektronů

Pod tímto termínem, do češtiny lze „yield“ přeložit jako „výtěžek“, můžeme nalézt informaci o tom, kolik sekundárních elektronů získáme na jeden elektron primární. V odborných materiálech je značen řeckým písmenem δ . Experimentálně byly zjištěny hodnoty pro jednotlivé prvky i některé často používané sloučeniny. Kromě experimentálních měření lze hodnoty i dopočítat, většinou se udávají v tabulkách, pro hodnoty energií používaných v skenovacích mikroskopech. Pro příklad uvádím některé hodnoty u známějších prvků v tabulce 2.5.

Atomové číslo	Atom	Energie	δ
6	C	100-200	0.5-0.6
14	Si	250-300	0.9-1.1
82	Pb	500-700	1.0-1.3
26	Fe	400	1.1-1.3
78	Pt	700-750	1.35-1.7
47	Ag	700-800	1
79	Au	700-875	1.2-1.6

Obrázek 2.5: Úhlová distribuce vyzáření SE, zdroj: [12]

2.2.2 Zpětně rozptýlené elektrony

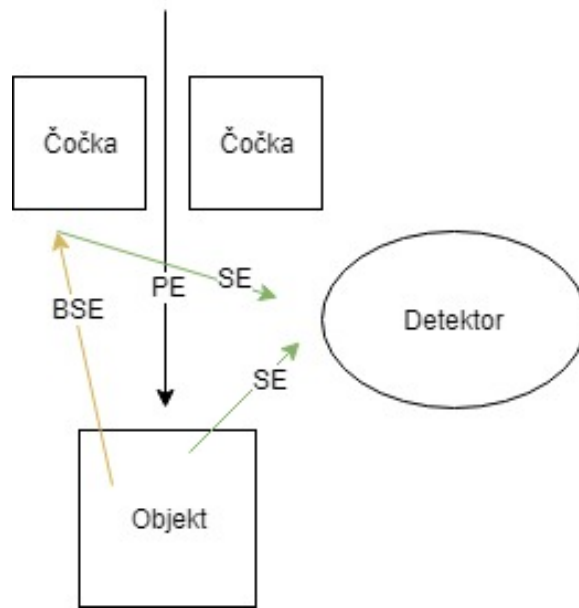
Zpětně rozptýlené elektrony jsou elektrony, které se objevují po dopadu primárního paprsku, podobně jako sekundární elektrony. Používáme pro ni zkratku BSE, která odpovídá anglickému termínu „BackScattered Electron“. Mezi BSE řadíme elektrony, které s sebou nesou vyšší energii. V článku [12] jsou to elektrony, které pro primární paprsek s energií v intervalu od 100eV do 1keV nabývají hodnoty vyšší než 50eV a menší než energie původního primárního elektronového paprsku.

Obdobně jako u sekundárních elektronů existuje pro zpětně rozptýlené elektrony koeficient, který udává počet takových elektronů. Pro značení se používá řecké písmeno η .

Vliv zpětně rozptýlených elektronů na sekundární elektrony

Sekundární elektrony nejsou generovány pouze z primárních paprsků. Když se podíváme na obrázek 2.6. Z něj vyplývá, že sekundární elektrony jsou emitovány v místech kontaktu paprsku o dostatečné energii s povrchem materiálu do hloubky λ , v angličtině zvana „escape depth“. Tato hloubka je velmi malá a detaily za touto hloubkou nejsou v elektronovém mikroskopu možné spatřit.

Podmínkou pro generování sekundárních elektronů je dostatečná energie zdrojového paprsku. Tuto hranici bezpečně překračují primární elektrony, ale stejně tak i zpětně rozptýlené elektrony, které při průchodu materiálem a přes hraniční hloubku emitování generuje další sekundární elektrony.

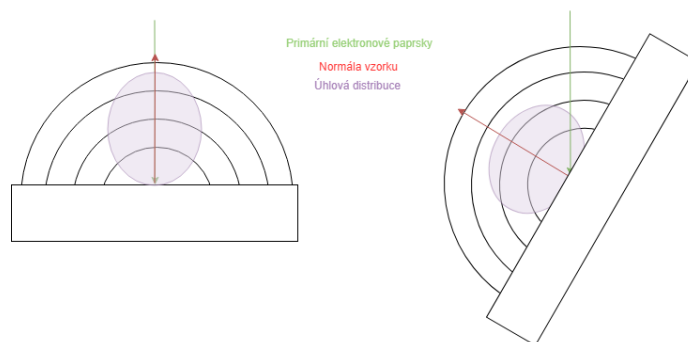


Obrázek 2.6: Schéma míst generování jednotlivých druhů elektronů

2.2.3 Interakce elektronu s povrchem

Při dopadu elektronu na povrch sledovaného vzorku dochází k proniknutí urychleného primárního elektronu skrz povrch. Podle hustoty sledovaného materiálu a jeho krystalické mřížky se dostaneme do různé hloubky materiálu. Část energie primárního paprsku se pohltí a ztratí, k nějakým ztrátám dojde vždy. Zbytek neztracené energie se přemění na výše zmíněné dva druhy signálů.

Pro BackScattered elektrony bylo experimentálně zjištěno, že jejich největší část je emitována ve směru normálového vektoru povrchu vzorku v místě dopadu[6], což se značně liší od vlastností odrazu světla. Ovšem pro prvky s nižším protonovým číslem je maximální hodnota vyzáření směrem, který téměř odpovídá odrazu. Toto bylo zjištěno experimentálně. Znázornění úhlové distribuce zpětně rozptýlených elektronů je načrtnuto na obrázku 2.7. Oblast distribuce je specifická pro každý chemický prvek.



Obrázek 2.7: Úhlová distribuce vyzáření BSE

Sekundární elektrony jsou pak elektrony, které z povrchu materiálu vypudila energie získaná dopadem paprsku primárních elektronů. Pro každý takový paprsek lze získat i několik sekundárních elektronů, což bylo potvrzeno experimentálním měřením [6, s. 22]. Nadřadou stranu, hodnota výtěžku sekundárních elektronů, z anglického „yield“, lze aproximovat pomocí analytické rovnice 2.1.

$$\delta = \delta_0 \sec \phi \tag{2.1}$$

Kde δ_0 je výtěžek elektronů pro vzorek s nulovým vychýlením. Úhlová distribuce sekundárních elektronů je pak dána Lambertovým kosinovým zákonem. V tomto případě už nezáleží na náklonu vzorku. Více než na náklonu záleží na sekundárních elektronech, které vyzáří zpětně rozptýlené elektrony BSE. Takto emitované sekundární elektrony jsou závislé na hloubce průchodu primárního paprsku do materiálu. [6, s. 23]

Kapitola 3

Realistické zobrazování

Tato kapitola se věnuje problematice realistického zobrazování, oblasti počítačové grafiky, kdy se tvůrci snaží co nejpřesněji vykreslit model scény. Ideální situace je, kdy člověk není v konečném důsledku schopen rozeznat, kdy se jedná o fotografii a kdy o „pouhý“ generovaný obrázek – render.

Realistické vykreslování scény je vykreslovací metoda vycházející z co nejpřesnějšího matematicko-fyzikálního modelu světa. Simulujeme skutečné vlastnosti světla, abychom docílili co nejvěrnějšího znázornění světla, stínu a ostatních optických jevů, jako například kaustik.

Realistické zobrazování je oblíbený problém, který je často řešen v kruzích počítačové grafiky. Je popsáno mnoho metod realistického vykreslování, které popisují, jak nejlépe simulovat fyzikální vlastnosti světla a světa, aby bylo dosaženo co nejlépejších výsledků. K tomu vznikají i hotové implementace vykreslovacích systémů, které využívají znalosti popsané v těchto článcích. Realistické vykreslovací systémy obvykle využívají pokročilejší metody sledování paprsků a vznikají s různými cíli: pro kvalitu výstupu, rychlost výpočtů – snaha o výpočty v reálném čase – a v neposlední řadě pro výukové účely v oblasti vykreslování. Poslední, výukové, implementace většinou popisují do detailu způsob implementace a představují možnosti upravení, pomalejší pak bývá samotný výpočet, ale ani tak se nejedná o přehnaně dlouhou dobu. Některé hotové implementace jsou popsány v následující sekci [3.1](#).

3.1 Existující implementace

V této sekci jsou popsány hotové systémy, které využívají metod sledování paprsků při vykreslování. Je to směs různých řešení využívající i hardwareové optimalizace, komerční i zdarma distribuované. Stručně je i zmíněn systém PBRTv3, kterému je pak vyhrazena i celá další sekce [3.4](#).

Mitsuba renderer

Na svých stránkách [\[3\]](#) uvádí, že Mitsuba renderer je „vykreslovací systém, založený na principech PBRT“. Rovněž se zde uvádí, že implementuje experimentální metody a sám se hlásí k orientaci na výukovou oblast o realistickém vykreslování. Mitsuba renderer je psán v jazyce C++.

Implementované metody

- Ambient Occlusion
- Global Illumination
- Monte Carlo pathtracing
- Bi-directional Pathtracing
- Photon Mapping
- Instant radiosity

System pro optimalizaci a urychlení výpočtu využívá paralelních výpočtů, kdy si sám vytváří vhodně velké clustery, které zpracovávají samostatné procesy. Důraz je také kladen na paměťovou náročnost – systém by měl zvládnout zpracovat třicet milionů trojúhelníků. Pro urychlení výpočtu na procesoru intel jsou využívány instrukce SSE2. Také jsou využívány optimalizační struktury pro dělení scény – KD strom včetně ořezávání.

Pro zadání dat se interně používá soubor formátu XML, ale pro snazší práci s trojrozměrnými modely je vytvořen nástroj pro převod COLLADA souborů do interního formátu a je plánováno rozšíření i pro další modelovací nástroje. Rovněž jsou v systému implementovány různé druhy materiálů a jejich odrazových (scattering) funkcí. Výstup systému je na obrázku 3.1.



Obrázek 3.1: Výstup z MITSUBA rendereru

Nvidia OptiX

Ray tracing engine [8] od společnosti Nvidia, který poskytuje nástroje pro implementaci výkonnostně náročných ray-tracing metod optimalizovaných pro výpočet na grafické kartě. Akcelerace na grafické kartě přináší opravdu mnoho možností urychlení výpočtu na GPU, ale to, že se jedná o vývojový kit od komerční společnosti s sebou nese i nevýhodu v podobě závislosti na grafické kartě od Nvidie. Minimální požadavky pro běh jsou grafické karty postavené na architektuře Maxwell z roku 2014, C++ překladač a CUDA toolkit. Operační

systemy jsou podporovány i Windows i Linux. Dalším požadavkem je poslední aktualizovaná verze grafického ovladače.

K frameworku jsou poskytnuty příklady použití, včetně kernelů a modelů a online dokumentace přístupná na adrese [7]. Dokumentace byla využita pro vygenerování vzorových obrázků.

Klíčové vlastnosti Nvidia OptiX

- Programovatelná Ray-Tracing pipeline, akcelerovaná na GPU
- Single-ray programový shader s využitím C++
- Optimalizováno pro aktuální GPU hardware
- Automatický přístup do paměti
- Využití umělé inteligence pro vykreslování
- Akcelerace výpočtu na RT jádrech
- Zdarma pro nekomerční užití.

Intel Embree/OSPRay

Intel Embree [2] je „kolekce výkonných kernelů pro sledování paprsků“ vyvinutých společností Intel a šířená je pod svobodnou licenci Apache 2.0 a otevřenými zdrojovými kódy. Podobně jako OptiX využívá specifické GPU a technologie od Nvidie, je Embree postaveno nad architekturou procesorů Intel. Optimálně využívá jejich specifických instrukcí, od SSE přes všechny verze AVX, po poslední verzi AVX-512. Vše ve frameworku Embree je optimalizováno pro snadnou aplikaci ve vlastních vykreslovacích systémech užívající metody sledování paprsků. V dřívějších verzích byl poskytován i modelový engine „Embree example renderer“, který už ale v dnešní době není podporován, protože přešel do samostatného projektu Intel OSPRay. S využitím Embree je pak postaveno mnoho komerčních a známých vykreslovacích engineů:

- Corona Renderer
- FluidRay RT
- Visual Designer 3D
- Glise
- Autodesk Maya 2014

3.2 Hardwarová akcelerace výpočtů

V poslední době se hledají další způsoby, jak optimalizovat realistické vykreslování natolik, aby se dala scéna realisticky vykreslovat v reálném čase. Takový výpočetní výkon by přinesl mnoho nových možností do herní počítačové grafiky, kdy je potřeba scénu překreslovat přibližně šedesátkrát za sekundu. Z toho důvodu jsou vyvíjeny nové způsoby optimalizace: Nvidia a její grafické karty s RTX částí, k níž se přidala i společnost Intel s GPU Xe [14].

Nvidia RTX

V poslední řadě grafických karet od Nvidie, GeForce 20xx, se nově objevují specializované čipy, pro urychlení ray-tracingových metod. Přístup k těmto čipům je již možný přes všechny grafické vývojové kity, Vulkan API, Microsoft DXR a výše zmíněná Nvidia Optix. Aktuálně technologie není v takovém stavu, aby dokázala realisticky, ray-tracingem, vykreslit celou scénu v reálném čase. Z toho důvodu se využívá kombinace standardní vykreslovací „pipeline“ a detaily a efekty, které je lepší a efektivnější počítat formou sledování paprsků, jsou akcelerovány pomocí těchto nových čipů. Tyto a další informace lze dohledat na stránkách Nvidie o technologii RTX [9].

3.3 Sledování nelineárních paprsků

Protože v mikroskopech jsou trasy paprsků nelineární, je vhodné hledat inspiraci v grafickém vykreslování a v člancích zabývajících se sledováním nelineárních paprsků při vykreslování speciálních efektů, jako jsou například vykreslení krajiny pod vodou. Dále pak atmosferické jevy, nejznámější je asi fata morgána, kde nelineární lom světla způsobují rozdílné teploty vzduchu v různých vrstvách atmosféry. Počítačová simulace tohoto a dalších jevů je popsána v článku [13]

Ve zmíněném článku je popsán algoritmus pro výpočet globálního osvětlení „Volume photon mapping“, rozšířený o výpočet hodnoty tak zvaného „Inelastic Scattering“, tedy rozptylu, kdy při absorpci příchozích částic, v případě světla fotonů, dochází k emisi částic s jinou energií, než se kterou přiletěly. Energie emitovaných částic pak může být i větší i menší.

V článku [13] ve třetí kapitole autoři popisují, jakými způsoby lze jevy vznikající nelineární trajektorií částic simulovat. Zakřivení drah se totiž obtížně simuluje, je výpočetně nepoměrně náročnější, než v případě rovných přímek. Jsou zde zmíněny tři metody: a to považovat paprsky pro výpočet světla a stínů, tedy mezi místem kolize a světelným zdrojem, za přímky i v tomto případě nebo mít předem nachystanou texturu, kde jsou všechny světelné jevy zachyceny a vyčteny. Obě řešení ale nevycházejí z fyzikálního modelu, tedy přináší drobné odchylky ve výsledcích. třetí řešení popsáno v článku pak zkouší voxelizovat scénu a nahrubo si předpočítat místa a směr dopadu světla z každého světelného zdroje pro každý vytvořený voxel. Ani jednu ze zmíněných metod nepovažují autoři Diego Gutierrez a kolektiv za dostatečně přesnou, odpovídající reálným fyzikálním vlastnostem. Proto přišli s vlastním rozšířením algoritmu.

3.4 PBRTv3

PBRTv3[10] je vykreslovací systém, který implementuje teorii fyzikálně inspirovaného vykreslování. Jedná se o výukovou implementaci teorie popsané v knížce Physically based rendering. Tento renderer je značně upravovatelný a podporuje snadné doplnění nových metod a nových prvků, které se využívají pro vykreslování. Od různých metod vzorkování přes všechny typy kamer (ortografické a perspektivní jako hlavní využívané). Také jsou implementovány různé osvětlovací metody, což je nejpodstatnější část všech vykreslovacích systémů. V PBRTv3 jsou implementovány metody, jako přímé osvětlení(direct lighting), sledování cesty (pathtracing), obousměrné sledování cesty (bidirectional pathtracing) i ve variantě s užitím algoritmu Metropolis-Hastings.

3.4.1 Komponenty

Vstupní soubor

PBRT využívá vlastní formát pro definici vykreslované scény. Má koncovku „.pbrt“ a kromě popisu modelu v podobě vertexů, indexů a normál pro stěny obsahuje i informace pro PBRT specifické, například jakým způsobem má systém vzorkovat, jakou zvolit kameru, kde bude kamera umístěná. Také jsou zde definována všechna světla ve scéně, různých typů. V neposlední řadě je také možná volba různých vykreslovacích metod.

PBRTv3 si samo kontroluje formát vstupního souboru pomocí parseru. V něm je určeno, které komponenty musí být definovány, to znamená kontroluje, jestli je přítomna kamera, a alespoň jedno světlo. Dále je ve vstupním souboru definována i podoba výstupního renderu, jeho rozlišení, jméno a formát. Pro každou komponentu jsou tedy k dispozici různé parametry, zde také probíhá kontrola, jestli je vše podstatné nastaveno. U volitelných parametrů se pak používají výchozí nejčastější hodnoty. Pokud je definován parametr, který komponenta nepoužívá, je vypsáno varování a takový parametr se ignoruje. Jako příklad lze uvést parametr FOV (Field of view), který bychom se snažili nastavit ortografické kameře.

Sampler

Vzorkování je podstatná akce pro generování výsledného obrazu. Spojitou funkci obrázku je potřeba reprezentovat diskřetně a je tedy potřeba ji vhodně navzorkovat. Za tímto účelem vzniklo několik metod, které by měly přinést požadované výsledky. Cílem vzorkování je ušetření výpočetních prostředků, anti-aliasing. Vhodné vzorkování přispívá i ke kvalitě výsledného obrázku, kdy vzorkovací algoritmus vhodně vybírá oblasti, které je potřeba lépe a přesněji zvýraznit (plochy s objekty) a naopak zjednodušit výpočty v místech, kde detaily nejsou tak významné (jednolitě plochy stejné barvy).

V PBRTv3 je implementováno základní vrstvené vzorkování (stratified sampling), kdy se obraz rozloží na mřížku a podle různých metodik ze každý dílek vzorkuje. Metoda má horší vlastnosti výstupu, proto se moc často nepoužívá. Další implementované metody jsou Haltonové vzorkování a Sobolovo vzorkování.

Světla

Pro vykreslení scény je nejpodstatnější světlo, bez světelného zdroje je celá scéna vykreslená jako jednolitá černá plocha.

PBRTv3 poskytuje rozhraní pro implementaci světel, které lze doimplementovat z vlastní iniciativy, jak je popsáno na stránce [4]. Klíčová metoda světla je `Sample_Li()`, kde se vypočítává příspěvek světla k výsledné barvě a intenzitě a také vektor W_i , který udává směr, odkud na daný bod světlo svítí.

Druhy světel v PBRTv3 jsou nachystány nejčastěji používané typy světel. Každé světlo s sebou nese informaci o tom, jakého je typu a podle toho je s ním zacházeno. Je poskytnuté bodové světlo pro napodobení reflektorů, distantní nebo směrová (directional) světla, která působí ze stejného směru na libovolný bod scény. Pak plochá světla, která mají charakteristiky podobné stropním svítidlům – světelná plocha v určité vzdálenosti. Poslední druh je pak ploché světlo v nekonečné vzdálenosti, které se používá pro simulaci slunečního světla.

Materiály

Pro vykreslení je důležité znát fyzikální vlastnosti povrchu, který je zobrazován. Například velký rozdíl je třeba v propustnosti světla (sklo versus jiné). Proto je potřeba pro každý materiál dodefinovat způsoby odrazu. Většinou je daná vlastnost popsána pomocí BRDF. (Bidirectional Reflectance Distribution Function).

Pro simulátor mikroskopu by se vlastnosti materiálu daly využít pro obdobnou funkci, totiž popsat způsob odrazu elektronů, respektive elektronového paprsku, od materiálu. Je tedy třeba dodefinovat obdobu BRDF podle fyzikálních vlastností elektronů. Zde je také prostor pro vytvoření vícero druhů materiálů. Dá se říci, že každý chemický prvek má různou strukturu atomů, tudíž je nutno pro téměř každý použitý materiál doimplementovat další funkce.

Kamera

Systém poskytuje tři základní druhy kamer využívané pro vykreslovací metody. Je naimplementována perspektivní kamera, kopírující perspektivu reálného světa, pak také kamera enviromentální.

V neposlední řadě je naimplementována i kamera ortogonální, která zachovává rovnoběžnost přímků ve scéně.

Vzhledem k tomu, že při mapování elektronového mikroskopu na PBRT odpovídá kamera zdroji elektronových paprsků a tyto paprsky dopadají kolmo na zkoumaný vzorek, nejvíce se k tomu blíží právě ortogonální kamera, která vysílá paprsky kolmo z kamery. I zde je ovšem prostor pro vlastní implementaci, která by umožnila vzít v potaz i další proměnné. V reálném systému elektronového mikroskopu například dochází k jevu, kdy se paprsky odrazí od zkoumaného vzorku do kamery a od ní teprve do detektoru.

Integrátor

Nejdůležitější součást, kde je implementována renderovací metoda. S využitím dat z kamery a světél, myšleno, kde se co nachází a jaké mají vlastnosti, dopočítá výslednou barevnou hodnotu pixelu. Pro každý světelný paprsek se provádí výpočet právě zde. Komponenta je zvaná jako „Integrátor“ proto, že jejím cílem je vypočítat rozložení záření světla v prostředí. [4]

Obdobně jako ostatní komponenty, i integrátor nabízí možnost vlastní implementace. Stačí implementovat rozhraní integrátoru, které vypadá následovně:

```
class Integrator {
public:
    <<Integrator Interface>>
    virtual ~Integrator();
    virtual void Render(const Scene &scene) = 0;
};
\\ zdroj: http://www.pbr-book.org/3ed-2018/Introduction/
\\ pbrt_System_Overview.html#Integrator
```

Jediné, co musí integrátor implementovat je metoda `Render()`, která provede výpočet rovnice a výsledky uloží do filmu, kde se kombinují výsledky jednotlivých paprsků. Každá metoda výpočtu osvětlení pak má vlastní integrátor, který kromě metody `render` přidává

ještě další akce, které jsou typické pro implementovanou metodu. Integrátor pro sledování cesty například implementuje

3.4.2 Upravitelnost PBRTv3

PBRT nabízí velmi mnoho možností, jak si lze systém upravit k obrazu svému. Samotní autoři tuto možnost prezentují na oficiálních stránkách projektu[5]. Nabízí celkem jednoduché a srozumitelné api, kde se dají přidat vlastní v c++ implementované moduly a nimi pak rozšířit množinu již implementovaných metod, druhů světel a pod. Nahradit nebo doimplementovat lze v podstatě každou část PBRT.

Protože k PBRT existuje velice podrobná dokumentace v podobě výukové knížky, jsou dodatečné úpravy možné poměrně snadno a navíc lze dobře pochopit, jak celý systém funguje, proč je něco implementováno, jak je to implementováno a lze provádět opravdu hluboké zásahy.

Přidání modulu do API

Nejsnazší postup přidání nového prvku PBRT, například vlastního integrátoru, je inspirace v již implementovaných součástech. Pro přehlednost se může hodit dodržovat strukturu souborů v PBRT, případně si vytvořit vlastní složku na všechny přidané komponenty. Já se rozhodl pro první variantu a dodržoval jsem adresářovou strukturu PBRT.

Vytvoří se nový zdrojový a hlavičkový soubor pro nově implementovaný modul. Tento modul musí splňovat nějaké základní parametry a implementovat rozhraní, které je definováno v PBRT. Například pro integrátor je třeba implementovat metodu „Li“, u ostatních komponent, je to obdobné, je třeba naimplementovat všechny metody, které PBRT provolává pro danou funkcionalitu.

V neposlední řadě je také potřeba zaregistrovat novou komponentu v PBRT API. Pro verzi tři je API definováno v souborech `/src/core/api.*`. Zde také probíhá načítání a analýza vstupního PBRT souboru. Zde je potřeba doimplementovat volání konstruktorů nových modulů, které odpovídají hodnotám požadovaných ve vstupním souboru. Například, při volbě integrátoru volíme "path" pro metodu pathtracing. Pro nově implementovaný modul je vhodné použít charakteristické jméno, například "electron" pro elektronovou simulaci. Rozpoznání metody je pak pouhé porovnání řetězců.

Kapitola 4

Návrh aplikace a implementace

V této kapitole je popsán postup návrhu aplikace v sekcích 4.1 až 4.3. Následující sekce 4.4 popisuje implementační detaily a zvláštnosti. Jsou zde zmíněny i přijaté kompromisy vynucené použitím frameworku PBRTv3.

4.1 Analogie mezi raytracingem a simulací

Při návrhu simulátoru můžeme vycházet z fotorealistické obasti počítačové grafiky a jejich vykreslovacích metod. Jednotlivé prvky z pathtracingu lze namapovat a připodobnit ke komponentám elektronového mikroskopu. Pro příklad, kamera při vykreslování generuje paprsky – elektronová zbraň generuje elektronové paprsky. V obou případech lze paprsky sledovat a vypočítat jejich trasu. Když budeme pokračovat, při vykreslování řešíme, zda-li je bod dopadu zastíněný či nikoliv. U elektronového mikroskopu zjišťujeme, jestli se elektron z bodu dopadu dostane do detektoru. V obou případech na tom závisí barva výsledného pixelu.

Kapitola sama pro sebe je sledování nelineárních trajektorií paprsků. Tento problém se okrajově řeší pro simulaci speciálních efektů, jak je popsáno v sekci 3.3. Oproti tomu v elektronovém mikroskopu jsou všechny trajektorie paprsků nelineární – vlivem působení elektromagnetických polí jednotlivých komponent. Proto je potřeba najít dostatečně přesnou a rychlou metodu aproximace této nelineární trajektorie.

4.2 Simulátor

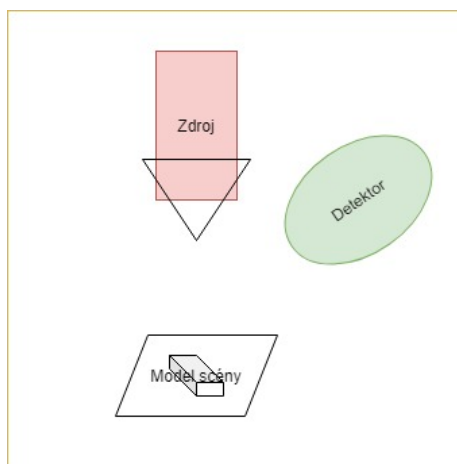
Tato sekce popisuje způsob tvoření programového simulátoru činnosti mikroskopu. Vstupem programu bude model zkoumaného vzorku, u kterého se určí materiálové vlastnosti a hlavně tvar.

Další důležitou součástí je rozmístění a typ komponentů použitých v mikroskopu. Simulátor bude umožňovat nastavovat pozice prvků a jejich vlastnosti. Rovněž by měl poskytovat i více metod výpočtu výsledného obrázku – v komerčním prostředí by se používala nejkvalitnější, ale zde je prostor pro experimentování a srovnání různých metod přístupu k problému.

4.2.1 Abstrakce

Nejpodstatnější část tvorby simulátoru je vhodně zvolená abstrakce simulovaného systému. Výhodou simulace elektronového mikroskopu je, že pro celkem realistické výstupy můžeme

hodně věcí zjednodušit. Jak v samotném výpočtu tak i u parametrů a vlastností jednotlivých komponentů. Abstrahovaný systém je znázorněn na obrázku 4.1. Jedná se o zjednodušený skenovací mikroskop s detektorem sekundárních elektronů umístěným přímo ve zdroji elektronového paprsku. Je to stejná situace, jako by ve vykreslované scéně pomocí sledování paprsku bylo světlo umístěno přímo v kameře.



Obrázek 4.1: Abstrakce mikroskopu

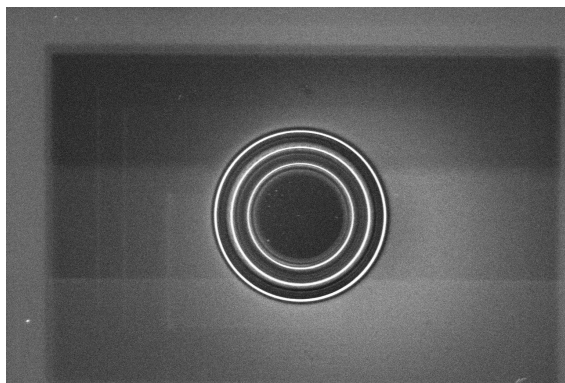
V následujících podsekcích jsou popsány jednotlivé komponenty a detailněji i provedené zjednodušení pro usnadnění výpočtů, pochopitelně s dopadem na podobu výsledného obrázku.

Detektor

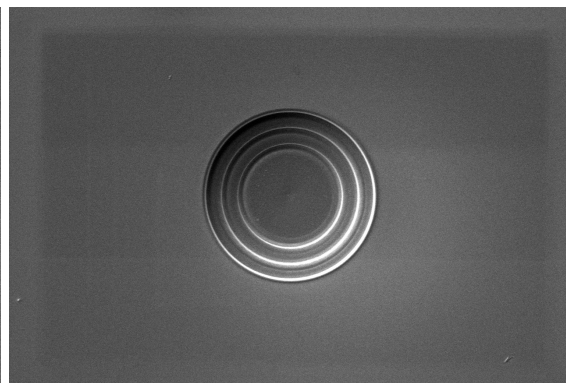
Detektor v reálném systému je zařízení, které slouží k detekci odražených elektronů ze vzorku. Má určitou charakteristiku, která udává, jaké elektrony přijímá, tak zvanou akceptační funkci, která zobrazuje závislost na energii elektronu a jeho zachycení detektorem. Můžeme rozlišit dva druhy detektorů, jeden zachytává elektrony s nižší energií, druhý s vyšší. Existují i kombinované, každý ze zmíněných přináší jisté výhody i nevýhody.

Kromě detekování elektronů, které letí směrem do detektoru, dochází i k ovlivňování trajektorie elektronů samotným elektromagnetickým polem detektoru. Ten pak zachytává i elektrony, které by za normálních okolností neměly šanci do detektoru doputovat. Proto můžeme na výstupních obrázcích z mikroskopu vidět jevy, kdy na první pohled jasně zastíněná oblast zobrazena velice jasně, jako kdyby byla přímo pod světlem.

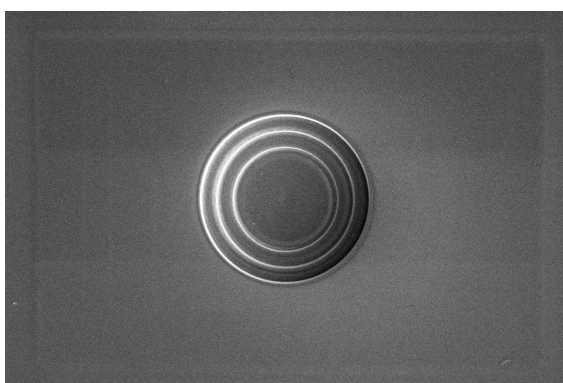
V reálném systému se vyskytují dvě třídy elektronových detektorů – jeden druh snímá zpětně odražené elektrony, druhý obvyklejší pak snímá sekundární elektrony. Zde v dokumentu bych rád představil tři druhy detektorů sekundárních elektronů, které se značí zkratkami ETD, ICE a TLD. Detekce elektronů je u všech tří typů přináší obdobné výsledky. V reálných systémech je hlavní rozdíl tvořen v tom, že každý z detektorů je na odlišném místě, jak můžeme vidět při srovnání obrázků 4.2, 4.3 a 4.4.



Obrázek 4.2: Detektor TLD



Obrázek 4.3: Detektor ICE



Obrázek 4.4: Detektor ETD

Všechny tři můžeme tedy celkem úspěšně simulovat stejným způsobem s tím, že pro různý detektor změním akorát pozici detektoru. Tak bude docíleno obdobných výsledných obrázků, jako na srovnání výše.

I samotnou detekci lze značně zjednodušit, kdy díky tomu, že elektronové detektory, „vtahují“ letící elektrony svým elektromagnetickým polem. Zjednodušení spočívá v zanedbání energií elektronů. Vezmeme v potaz pouze trajektorii a řešíme, jestli je elektron vtažen či nikoliv. S jistotou pak lze tvrdit, že když se elektron dostane nad nejvyšší bod zkoumaného vzorku (a detektor je umístěn nad povrchem, ve skutečných systémech je tato podmínka splněna) lze tento elektron považovat za detekovaný.

Zdroj elektronů

V teoretické části jsou zmíněny dva zdroje elektronů, jejich skladba a metoda, jakým způsobem jsou paprsky směřovány a koncentrovány pomocí elektromagnetických čoček na jeden výstupní bod ze zdroje elektronů. Odtud pak víceméně po přímce letí směrem ke sledovanému vzorku. Pro tvorbu obrazu nás zajímá právě tento odchozí paprsek, to znamená, že se nemusíme zabírat jednotlivými čočkami a vzít v potaz pouze pozici elektronové zbraně a směru, který se primární paprsek ubírá.

Z toho důvodu můžeme zanedbat, je-li použité v systému wolframové vlákno nebo hrot, stejně tak i počet usměrňovacích čoček. V simulátoru tedy bude zdroj reprezentovat pozice a směr. V reálném systému nelze pozici zdroje jednoduše měnit, operativně lze pohybovat jen stolcem, popsáním v následující podsekcí.

Pohyblivý stolek

Stolek se v mikroskopu stará o pohyb zkoumaného vzorku. Základní modely umožňují posun v osách x a y , pokročilejší varianta dovoluje i pohyb po ose z pro přiblížení nebo oddálení vzorku. Nejpokročilejší stolek pak poskytuje i rotace podél všech tří os. Existují i varianty, kdy možnosti rotace jsou omezeny na pouze dvě osy.

Simulace pohyblivého stolku není velký problém. Pohyby zkoumaného vzorku lze napodobit jednoduchými transformacemi scény.

Zkoumaný vzorek

Zkoumaný vzorek je trojrozměrná struktura. Zde se nabízí dvě varianty reprezentace vzorku. Buď pomocí voxelového modelu nebo pomocí trojúhelníkového modelu. Protože více fyzikálních vykreslovacích systémů pracuje s trojúhelníkovými modely, bude snazší pro simulaci využít právě ten. I když voxelový model poskytuje více možností pro modelování detailů – vzorky pro elektronové mikroskopy mají často zrnitou strukturu, rovinných hladkých ploch se objevuje minimum, protože přiblížením jsou všechny strukturální vlastnosti jasně viditelné. Nevýhoda je nutnost tvorby voxelového modelu, proto pro simulátor použijeme trojúhelníkový model, za cenu nižší přesnosti, ale jednodušší a rychlejší tvorby pomocí libovolného modelovacího softwaru.

4.3 Simulace

Samotná simulace je pak nejpodstatnější část generování nového obrazového výstupu. Řízení simulace je rozhodující pro spojení všech prvků, implementují se zde metody pro sledování nelineárních paprsků.

Jednodušší řešení

Pro nástin a ověření správnosti řešení lze vytvořit velice jednoduchou simulaci, která zanedbává několik nepodstatných fyzikálních detailů skenovacího elektronového mikroskopu. Například paprsek mikroskopu – v reálném systému se jedná o svazek o určitém průměru, parametricky nastavitelném. Tento paprsek můžeme zjednodušit a průměr zcela zanedbat. Také některé vlastnosti detektoru lze zjednodušit za minimální ztrátu kvality ve výsledném obraze.

Největším zjednodušením ale je zanedbání tak zvaných „BackScatter“ elektronů. Tento druh elektronů, generovaný při dopadu primárních elektronů, přináší převážně informaci o struktuře materiálu zkoumaného vzorku. Například z jakého prvku se skládá. Toto jsem se rozhodl pro demonstraci funkčnosti zanedbat, jelikož nás v prvotní fázi zajímal pouze tvar povrchu zkoumaného vzorku.

Zpřesnění simulace

Největší možnost a prostor pro zpřesnění je věrněji simulovat například cestu elektronového paprsku. Celá cesta paprsku by se dala krocovat směrem k detektoru a průběžně kontrolovat a přepočítávat jeho trasa. Protože je trasa ovlivněná i elektromagnetickým polem, cestuje elektron po křivce a nikoliv po přímce, jako v případě světla.

Mimo cesty lze simulovat i vliv „BackScattered“ elektronů. Ty totiž při vytvoření a opětovném nárazu do povrchu vzorku mohou vytvořit další sekundární elektrony a tyto

se v konečném důsledku promítnou do právě zkoumaného bodu. V reálném systému je to celkem běžný jev, tudíž je celkem vhodné a realistické ho reprodukovat v simulátoru.

4.4 Implementace

V rámci diplomové práce byla vytvořena jednoduchá implementace simulace. Rozhodl jsem se simulátor vytvořit s pomocí systému PBRTv3, díky jeho možné upravitelnosti a také perfektním doplňkovým materiálem v podobě knížky o realistickém zobrazování a její on-line verzi. Jednotlivé prvky simulátoru jsou popsány níže, odpovídají navržené abstrakci v předchozí sekci. Také jsou zde popsány jednotlivé úpravy a rozšíření PBRTv3.

Detektor

Detektor je implementován v souborech `src/lights/electronDetector.*`. Umístění je zvoleno s ohledem na adresářovou strukturu PBRTv3 – pro tento systém je detektor implementován jako světelný zdroj. Z toho plyne, že elektronový detektor musí implementovat rozhraní pro světla.

Co detektor obsahuje navíc, je metoda pro určení, zda-li letící elektron byl daným detektorem zachycen. Jedná se o metodu `bool ElectronDetector::DetectorHit(Ray)`, která přijímá informaci o letícím paprsku a vrátí informaci o zásahu, tato metoda je volána při samotné simulaci a slouží jako obálka pro vnitřní implementaci detektoru. Uvnitř metody je volána funkce `magneticFieldFunc(float r, Ray ray)`. Toto řešení je zvoleno z důvodu rozšířitelnosti o další varianty detekcí. Aktuálně jsou implementovány dvě, náhodná funkce, která přijme elektron s padesátiprocentní pravěpodobností. Druhá metoda přijme všechny elektrony, které se dostanou do kladné výšky ve scéně. Pro další metody detekce je potřeba naimplementovat další statickou funkci a předat ji konstruktoru detektoru. Aktuální implementace neumožňuje přepnutí funkce pomocí vstupního souboru, využívá pouze porovnání pozice elektronu a modelu.

Zdroj elektronů

Vlastní implementace elektronové pušky je v souboru `src/cameras/electronGun.*`. Zdroj elektronů odpovídá běžné ortografické kameře, která je v PBRTv3 implementovaná a proto z ní také dědí. V rámci simulace je ale udělána příprava pro použití elektronových paprsků, které jsou doimplementovány přímo v jádru PBRTv3, v souboru `src/core/geometry.h`. Paprsky s sebou nesou hodnotu o energii, která by mohla posloužit pro další zpřesnění výsledků simulace. V reálném systému se totiž energie primárního paprsku mírně odchyľuje kolem střední hodnoty. Toto by měla obstarávat právě vlastní implementace generátoru paprsku. Bohužel PBRTv3 očekává, že všechny hodnoty pro generování paprsků jsou konstantní a proto nelze jednoduše docílit generování náhodných hodnot pro elektronový paprsek. Toto omezení je logické, protože výpočet umožňuje dobře paralelizovat a zaručuje tím generování stále stejných obrázků bez náhodných změn. V případě simulátoru se to ale úplně nehodí.

Integrátor

Implementace integrátoru pro výpočet intenzity pro konkrétní paprsek je nejzajímavější část celé práce. Pro umožnění práce s elektronovými paprsky včetně energie, jsem se rozhodl upravit PBRTv3 hlouběji, než je poskytnuté rozhraní pro implementaci vlastního integrátoru. Rozhodl jsem se odloučit bázovou třídu pro světelné a elektronové integrátory.

V souboru `src/core/integrator.h` je doplněna nová bazová třída pro elektronové integrátory.

Konkrétní modelová implementace je mezi integrátory v souboru `src/intergrators/electron.*`. V něm je nejpodstatnější metoda `ElectronIntegrator::Li()`. V této implementaci je metoda přetížená, jedna varianta počítá s paprsky z běžné kamery. Toto bylo využíváno pro testy a debuggování s perspektivní kamerou, pro lepší představu o tvaru modelu.

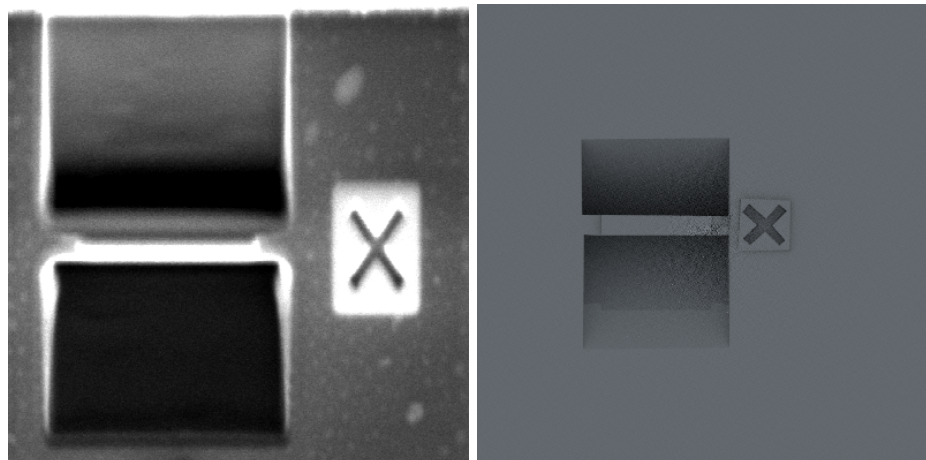
Materiály

Pro simulaci je potřeba do systému zanést i vlastní funkci distribuce paprsků. Elektronový paprsek při dopadu a emisi netvoří dvě složky jako běžný světelný. Spekulární složka paprsku v podstatě neexistuje, odraz elektronového paprsku je téměř dokonale difúzní a kopíruje kosinové rozložení. Materiály v PBRTv3 pokrývají běžné typy materiálů – kovové, plastová a podobně. V elektronovém simulátoru to není možné takhle zobecnit, protože vlastnosti a výsledná intenzita se liší pro různé druhy materiálů, jak je vidět na obrázku 4.5.

Implementace materiálů je doplněna podle adresářové struktury PBRTv3 na cestu `src/materials/silicon.*` respektive `src/materials/platinum.*`. Aktuálně jsou vytvořeny dva materiály použité v testovací sadě dat a v modelovém obrázku 4.5. Volba materiálu se provádí ve vstupním souboru pro systém PBRT. V něm jsou od sebe odděleny jednotlivé „meshe“, tedy sítě, které tvoří části výsledné scény nebo jednoho modelu. Pro každou lze specifikovat vlastní materiál jako parametr ve vstupním souboru.

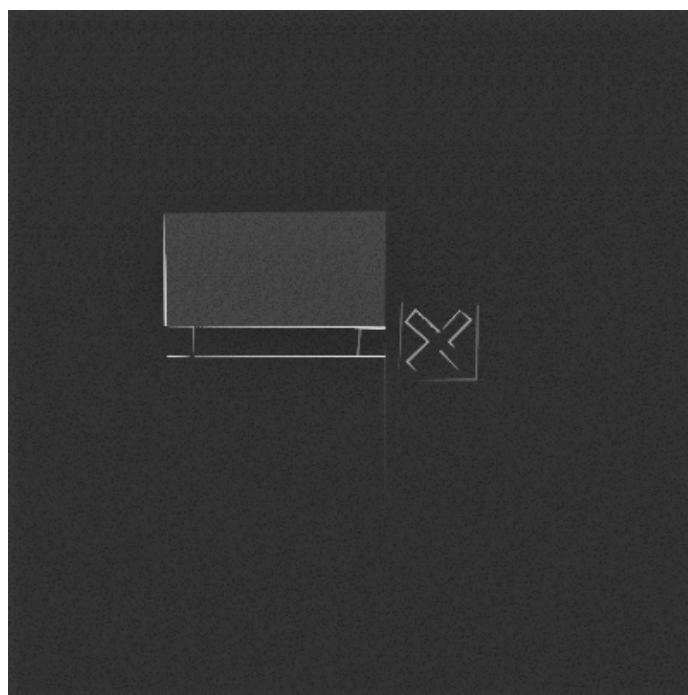
Přidání dalších materiálů je také jednoduché, stačí sledovat návod pro přidání části do api, vybrat si vhodný název materiálu. Nabízí se implementace materiálů pro každý chemický prvek zvlášť. Pro nové materiály je nutné dospecifikovat parametry, které jsou dále používány ve výpočtu. Je také nutné, pro fungování spolu s elektronovým integrátorem, aby implementoval potřebnou metodu `GetSEYield()`, která vrací počet sekundárních elektronů emitovaných na základě úhlu dopadu. Všechny materiály by tak měly dědit od třídy `MicroscopeMaterial` na cestě `src/materials/microscopeMaterial.*`.

4.5 Ukázka výstupu



Obrázek 4.5: Snímek z mikroskopu Obrázek 4.6: Neupravené PBRTv3

Na výstupu simulace jsou patrné světlé hrany modelu, kde dochází vlivem úhlu dopadu k uvolňování největšího množství sekundárních elektronů. Protože výstup simulátoru není úplně přesný, v obrázcích jsou velké rozdíly. Bílý čtverec kolem křížku na fotce z mikroskopu je způsobený různým materiálem oproti pozadí. Další rozdíl je v pozici detektoru, čímž dochází k menšímu zvýraznění hran ve spodní části obrázku ze simulátoru [4.7](#).



Obrázek 4.7: Výstup simulátoru

Kapitola 5

Testování a srovnání

Tato kapitola představuje experimentální část práce s implementovaným systémem. Sekce 5.1 popisuje způsoby testování, použité metriky a zdůvodnění, proč byly použity právě tyto metody. Sekce 5.2 popisuje akvizice testovacích dat pro tvorbu modelů, vytvoření simulovaných výstupů a přípravu pro srovnání s reálnými výstupy. Zbylé sekce se pak jednotlivě věnují konkrétním experimentům a jejich evaluaci.

5.1 Návrh testování

Pro výslednou diplomovou práci by bylo vhodné vytvořit metriku porovnání generovaných a skutečných snímků. Empirické srovnání uživatelem může být v první fázi užitečné pro ověření správnosti řešení. Konečné slovo by pak měl mít reálný uživatel mikroskopu, který by mohl vyzkoušet práci s koncovými uživatelskými aplikacemi, kdy v jednom případě by aplikaci využíval nad výstupem z generátoru a v druhém případě napřímo s reálným systémem. Cílem by bylo dosáhnout co nejpodobnější uživatelské zkušenosti v případě použití jednoho i druhého způsobu.

V neposlední řadě by měl být výstup simulátoru testován i pro trénování plánovaných neuronových sítí, které by měly usnadnit detekci objektů, kazů či jiných anomálií. Úspěšnost těchto sítí v praxi by pak mohla být dalším kritériem ukazujícím kvalitu výstupu ze simulátoru.

Stejně jako v případě prvotní verze simulátoru by se pohled uživatele na přímé srovnání výstupu mohl hodit i v pozdní fázi implementace. Podobně jako v případě realistického vykreslování by se mohl udělat test, kdy by se mělo určit, zda-li je obrázek generovaný nebo pochází ze skutečného systému. Nevýhodou ale je, že v případě snímků z elektronového mikroskopu, není množina potenciálních testerů tak velká, jako v případě vykreslování, protože lidí, kteří vědí, jak by měl správně snímek z elektronového mikroskopu vypadat, není tak velké množství.

5.2 Testovací data

Pro potřeby diplomové práce byly poskytnuty výstupy z elektronového mikroskopu pro křemíkovou destičku s vyhloubenými útvary. K nim jsou připraveny i trojúhelníkové modely. Data jsou přiložena na cestě `testData/images`. V adresářových podsložkách jsou umístěny snímky roztríděné podle detektoru, který byl použit ke snímání. Obrázky byly pořízeny pro každý model vždy s vychýlením o úhel 0° , 10° , 30° a 45° . Vychýlení vzniklo rotací kolem

osy x pohyblivého stolku. Více úhlů pohledu bylo pořízeno pro možnost porovnání chování hran na základě otožení preparátu.

Kromě křemíkových destiček z různými modely byl vytvořen i jeden speciální model pro ověření a demonstraci funkcionality simulace pro objekty s více druhy prvků. K tomu dobře posloužil model připravovaného preparátu pro transmisní elektronový mikroskop. Na něj se naleptává destička z prvku, která má výrazně vyšší intenzitu při pozorování pod mikroskopem.

5.2.1 Metriky pro srovnání výstupů

V této podsececi jsou přiblíženy použité metriky pro srovnání kvality obrázků. Jsou vybrány metodiky pro porovnání na základě fyzikálních vlastností signálu. Protože ty se vždy nehodí, je použita i metoda pro porovnání na základě strukturální podobnosti.

Informace o metodách srovnání obrazu byly čerpány z diplomové práce [1], implementace v srovnávacích metod pomocí pythonu byla převzata z [11].

PSNR

Špičkový poměr signálu a šumu je nejčastější metrika pro hodnocení kvality obrazu.

$$PSNR = 10 * \log \frac{m^2}{MSE} \quad (5.1)$$

Rovnice 5.1 se používá pro výpočet PSNR, kde MSE je střední kvadratická chyba, počítaná pomocí rovnice 5.2. m^2 je pak druhá mocnina maximální hodnoty signálu – pixelu.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|X_{i,j} - Y_{i,j}\|^2 \quad (5.2)$$

Výpočet MSE tedy provedeme tak, že sečteme druhou mocninu rozdílu hodnoty pixelu pro každou dvojici. Znamená to tedy, že oba obrázky musí mít stejné rozměry.

SSIM

Metoda měření podobnosti mezi dvěma obrazy, která vychází z vlastností lidského zraku. V rámci metody se porovnává jas, kontrast a korelaci. To znamená, že ve výsledku hraje větší roli defekt na jednoznačně jednoduše ploše než v místě, které má na každém pixelu různou intenzitu. Výpočet SSIM provedeme pomocí rovnice 5.3

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\sigma_x^2 + \sigma_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (5.3)$$

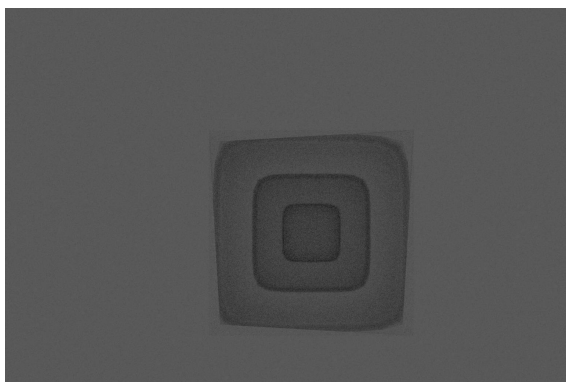
Kde $C_i = (K_i L)^2$ kde L nese maximální hodnotu pixelu a K_i jsou konstantní hodnoty blížíící se nule.

5.3 Srovnání

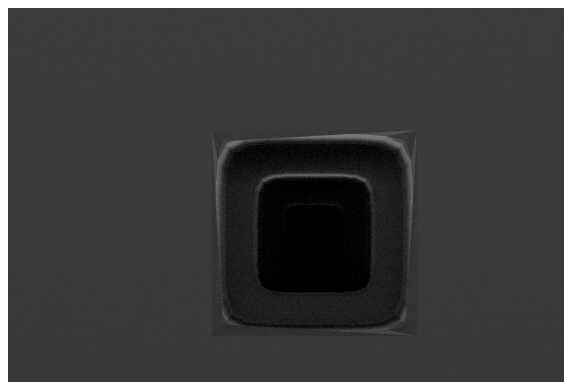
Tato sekce se věnuje srovnání a experimentům s implementovaným systémem. Podsekcce 5.3.1 ukazuje výstupy renderované metodou sledování cestu paprsku světla a elektronů, podsekcce 5.3.2 pak ukazuje srovnání mezi reálným a simulovaným mikroskopem.

5.3.1 PathTracing a ElektronTracing

Rozdíly mezi Path Tracingem a Electron Tracingem je jasný už z fyzikálního popisu. Vnímání výstupu je u elektronového mikroskopu trochu obtížnější, protože člověk nemá s čím porovnávat, jak dobře výstup vlastně vypadá. U path tracingu lze vycházet ze zkušeností z reálného světa a dokážeme určit, je-li některý ze stínů zle vykreslený.



Obrázek 5.1: PathTracing



Obrázek 5.2: ElektronTracing

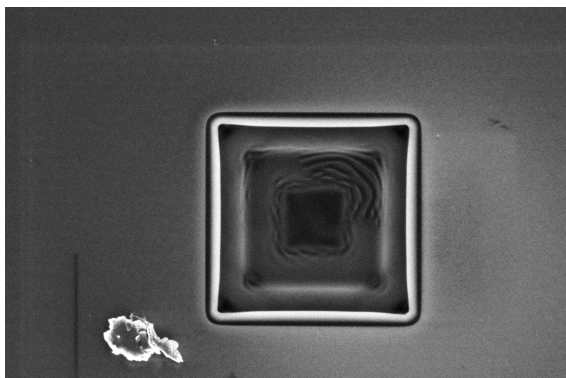
U elektronového mikroskopu je třeba znát přibližnou podobu modelu scény. Co je na každém obrázku zřetelné jsou hrany mezi plochami. Je také důležité znát, která oblast vzorku je z jakého materiálu.

5.3.2 Reálný systém a ElectronTracing

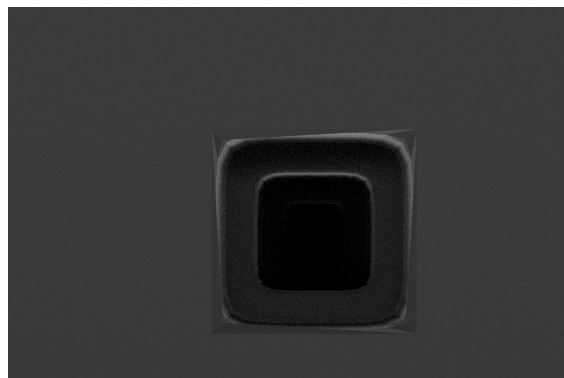
Jak bylo řečeno dříve, byla vytvořena sada testovacích dat. Na následujících stránkách jsem vytvořil vizuální srovnání pro jednotlivé obrázky – generované a pořizené. Pod každým modelem se nachází komentář a informace o tom, proč nejsou vždy výstupy simulátoru úplně ideální.

Dále se pod obrázky nachází informace o výsledcích provedených metrik a případně další zajímavé pohledy. Rozlišení porovnávaných obrázků je 1536 bodů na šířku a 1024 na výšku. Obrázky se simulátoru jsou vzorkovány osmi vzorky na pixel. Detektor elektronů je umístěn v místě zdroje – jedná se o simulaci systému s TLD detektorem. Z mikroskopu s TLD detektorem pochází i referenční obrázky. Výstupy jsou k vidění na následující stránce.

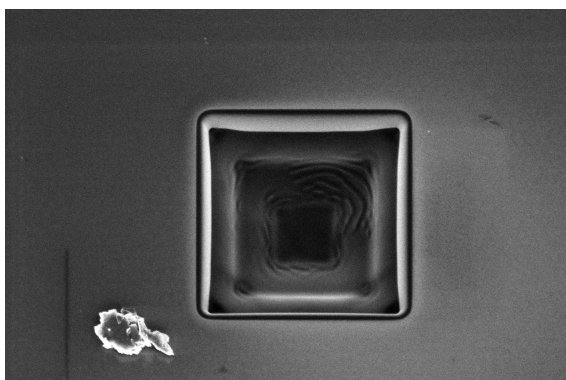
Víceúrovňové čtverce



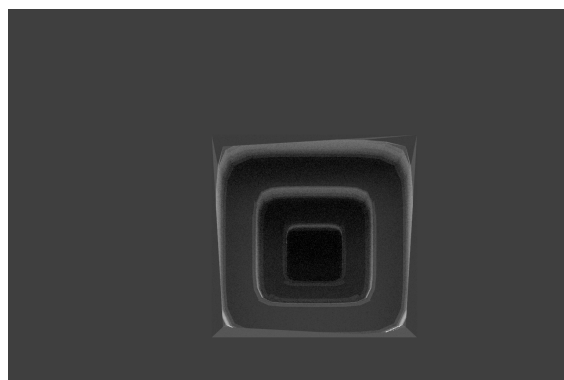
Obrázek 5.3: Náklon 0°



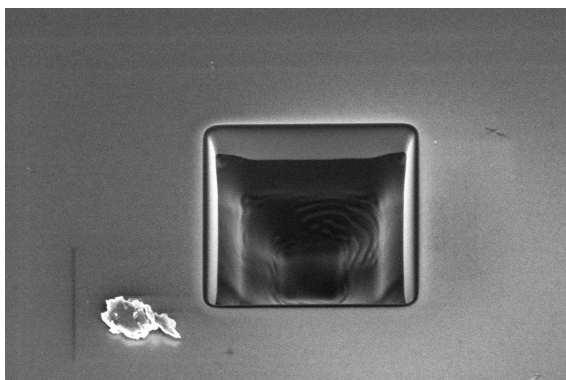
Obrázek 5.4: Výstup ze simulátoru



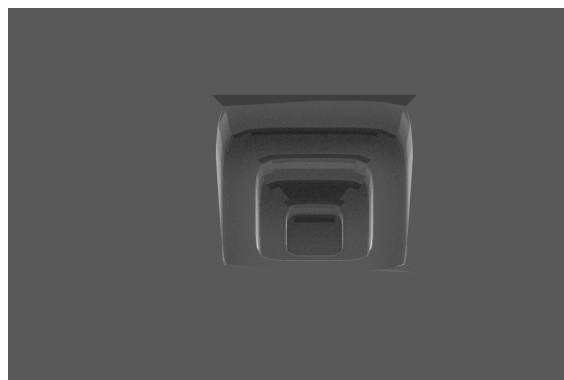
Obrázek 5.5: Náklon 10°



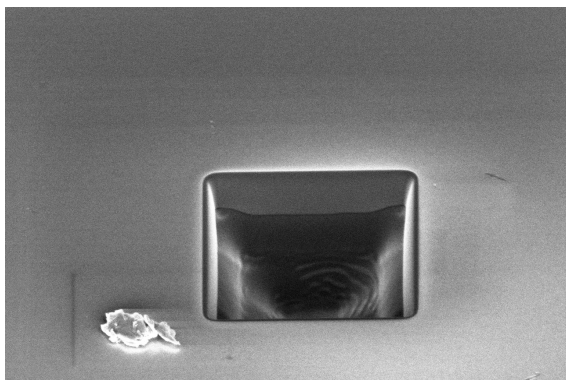
Obrázek 5.6: Výstup ze simulátoru



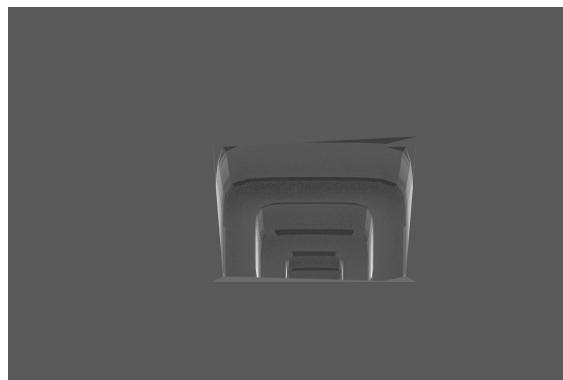
Obrázek 5.7: Náklon 30°



Obrázek 5.8: Výstup ze simulátoru



Obrázek 5.9: Náklon 45°



Obrázek 5.10: Výstup ze simulátoru

Zpočátku byla simulace testována proti relativně složité struktuře vzorku. Tomu odpovídá i složitost výsledného modelu, která v konečném důsledku stejně není dostatečná. Výsledné simulované obrázky dobře zobrazují dva typické jevy a to vyšší jas pro pixely u hran, kde vlivem úhlu dopadu dochází ke zvýšené emisi sekundárních elektronů a pak také celkový jas simulovaných obrázků, kdy s roustoucím úhlem náklonu tento jas roste. To je způsobeno tím, že se opět mění úhel dopadu na plochu materiálu a sekundárních elektronů se uvolňuje násobně více.

Na výstupech z mikroskopu jsou viditelné některé defekty. První, co padne do oka, je bílá skvrna nepravidelného tvaru u levého spodního rohu na všech snímcích. Jedná se o kaz. Další defekt je viditelný uvnitř druhého zanořeného čtverce. Jedná se o pozůstatek kopání do sledované křemíkové destičky. Původní snaha bylo vytvořit ostřejší hrany, jak je vidět na obrázcích se simulátoru.

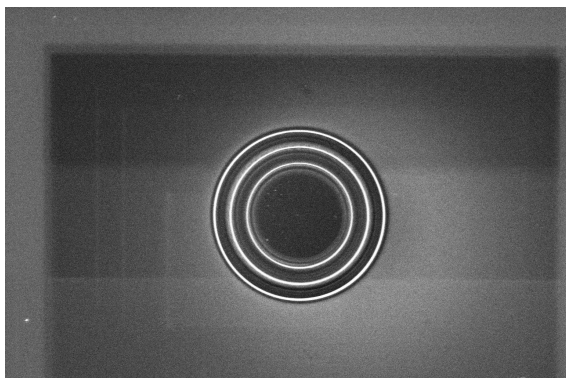
Výsledky měření

Výsledek není dostatečně uspokojivý, promítla se v něm nedostatečná přesnost vstupního modelu. V tabulce je přidána i doba vykreslení, je vidět, jak při zvyšujícím se náklonu vzorku stoupá počet sekundárních elektronů. Každý tento elektron je třeba vyhodnotit a tím se doba výpočtu prodlužuje.

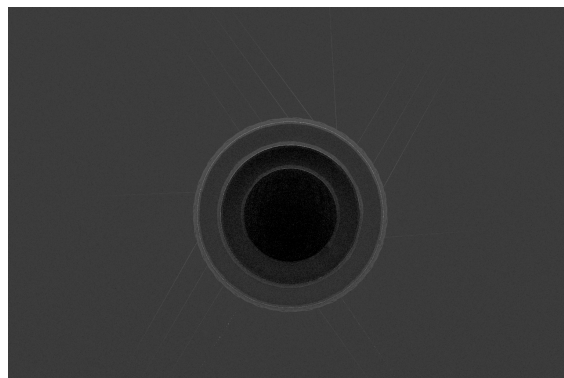
Úhel náklonu(°)	PSNR	SSIM	MSE	Doba generování
0	14.126	0.263	2514.879	~10s
10	16.173	0.302	1569.594	~20s
30	15.461	0.268	1849.136	~60s
45	13.398	0.248	2973.838	~60s

Tabulka 5.1: Výsledky měření pro vnořené čtverce

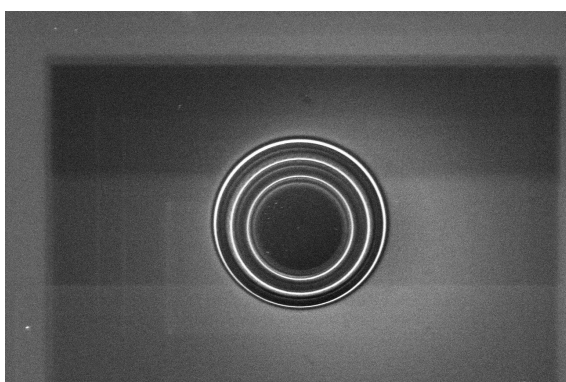
Víceúrovňové Kružnice



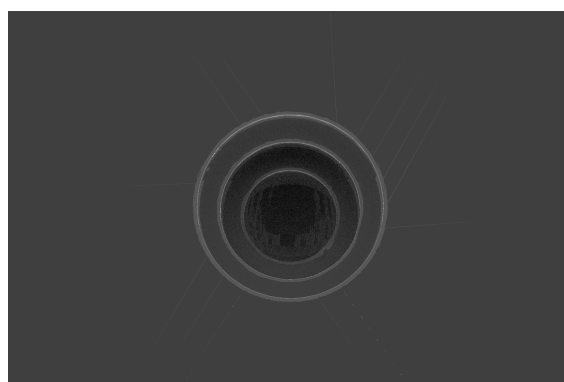
Obrázek 5.11: Náklon 0°



Obrázek 5.12: Výstup ze simulátoru



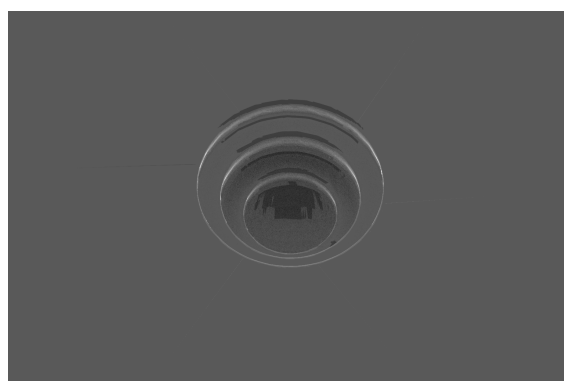
Obrázek 5.13: Náklon 10°



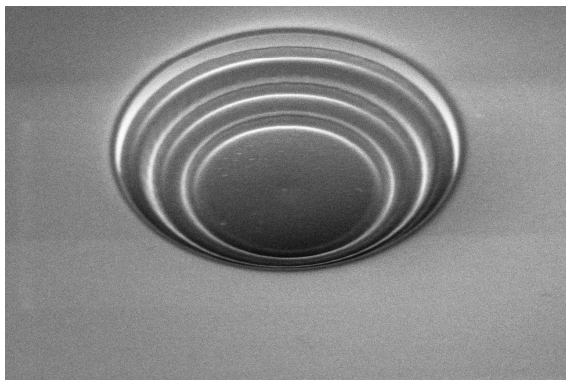
Obrázek 5.14: Výstup ze simulátoru



Obrázek 5.15: Náklon 30°



Obrázek 5.16: Výstup ze simulátoru



Obrázek 5.17: Náklon 45°



Obrázek 5.18: Výstup ze simulátoru

Druhý pokus se složitým modelem nedopadl o mnoho lépe. V tomto případě je zkoumána skupina soustředných kružnic, kdy každá další vnořená je o kousek hlubší. Na snímcích z mikroskopu jsou tři zřetelné horizontální pásy, které značí, že podklad, do kterého se kruhy hloubily, nebyl úplně rovný. Tato vlastnost není v modelu zanesena, proto v simulaci viditelná není. Také to bude jedna z příčin nízkého výsledku měření.

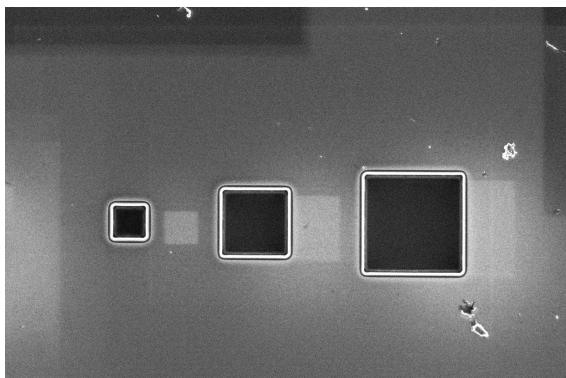
Další defekt je viditelný lehce na obrázku s třicetiprocentním náklonem a nejvíce se projeví u posledního obrázku 5.18. Tato chyba je způsobena rovněž nedokonalostmi modelu.

Výsledky měření

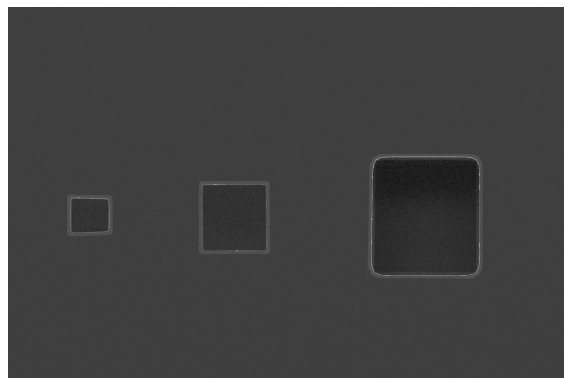
Úhel náklonu(°)	PSNR	SSIM	MSE	Doba generování
0	14.387	0.179	2368.204	~10s
10	15.864	0.197	1685.184	~17s
30	17.391	0.163	1185.684	~42s
45	12.733	0.170	3465.397	~53s

Tabulka 5.2: Výsledky měření pro soustředné kruhy

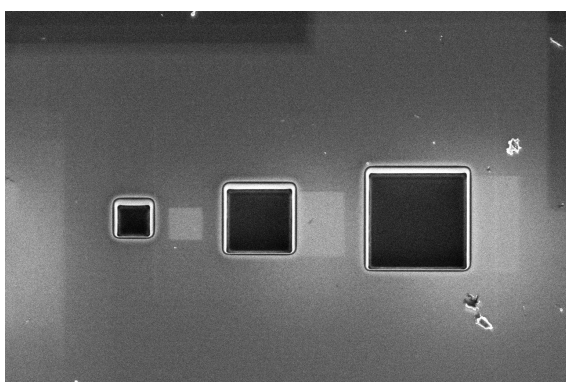
Tři velké čtverce



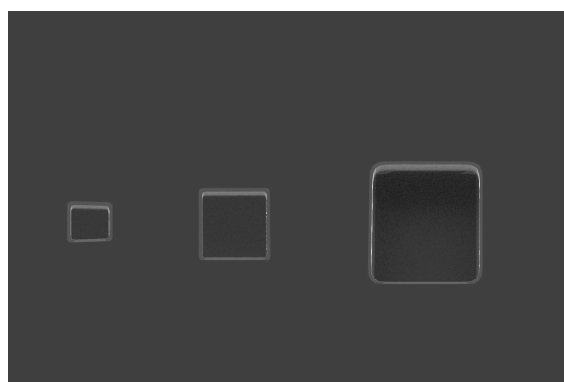
Obrázek 5.19: Náklon 0°



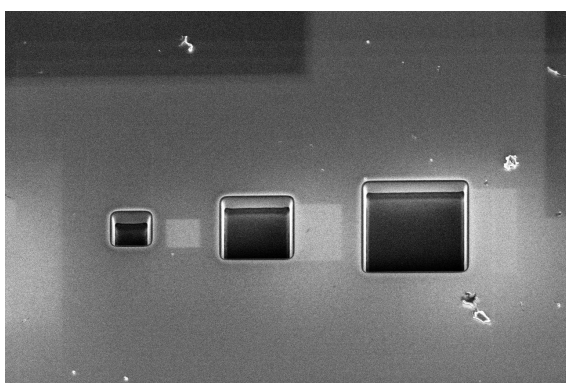
Obrázek 5.20: Výstup ze simulátoru



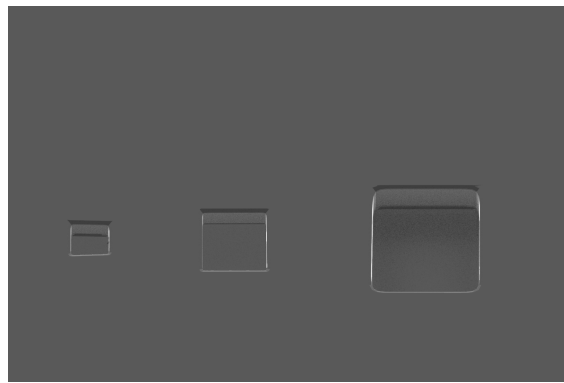
Obrázek 5.21: Náklon 10°



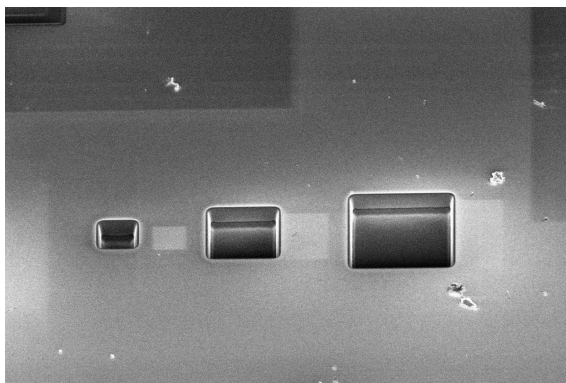
Obrázek 5.22: Výstup ze simulátoru



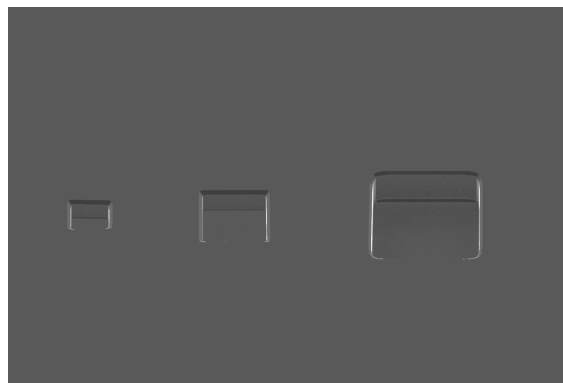
Obrázek 5.23: Náklon 30°



Obrázek 5.24: Výstup ze simulátoru



Obrázek 5.25: Náklon 45°



Obrázek 5.26: Výstup ze simulátoru

První experiment s jednodušším modelem materiálu v mikroskopu. Pro různé úhly jsou získány zhruba stejné hodnoty evaluačních metrik. Zvláštní pozornost si zaslouží vyjímka na obrázku 5.24, kde krásně vyšlo vykreslení jedné ze stěn – je vidět stejně jako na reálném snímku světlejší stěna s tmavší spodní hranou, která přechází do dna čtvercové prohlubně. Také v tomto konkrétním úhlu vyšla dobře spodní hrana. Tím bylo dosaženo vyššího hodnocení v metrice PSNR a MSE.

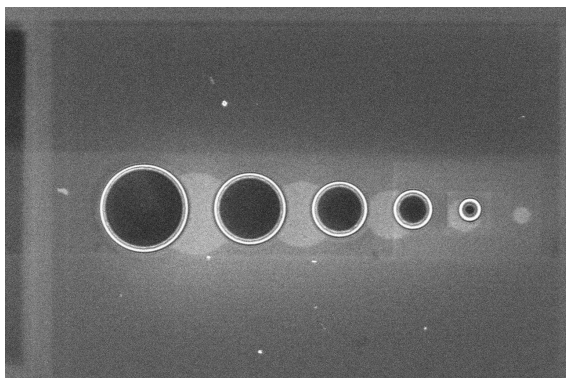
V referenčních snímcích je vidět několik odlesků a světlých skvrn způsobených výstupky na podkladovém materiálu. Na simulovaných obrázcích také není moc věrné znázornění hran. V reálném systému lze nastavit průměr plochy dopadu primárního paprsku. Simulátor tuto vlastnost neimplementuje.

Výsledky měření

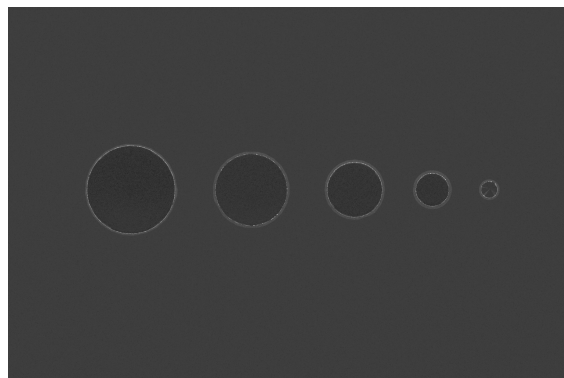
Úhel náklonu(°)	PSNR	SSIM	MSE	Doba generování
0	13.133	0.204	3160.424	~10s
10	13.969	0.214	2607.079	~14s
30	15.539	0.190	1816.147	~62s
45	13.569	0.176	2858.683	~60s

Tabulka 5.3: Výsledky měření pro tři čtverce

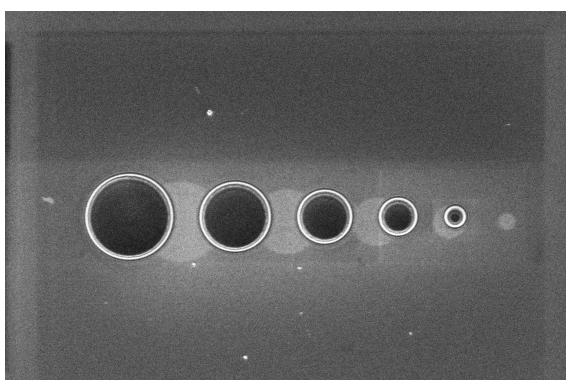
Malé kružnice



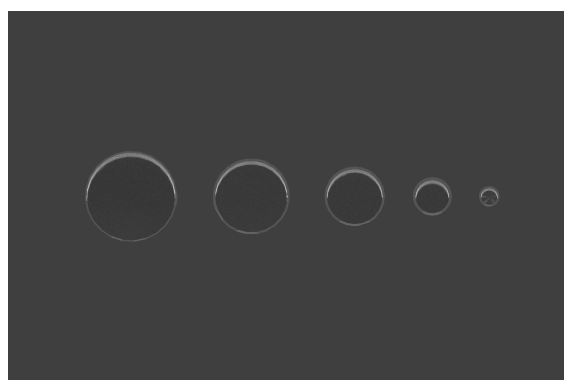
Obrázek 5.27: Náklon 0°



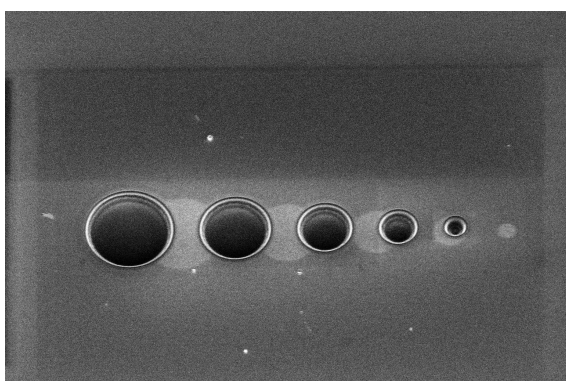
Obrázek 5.28: Výstup ze simulátoru



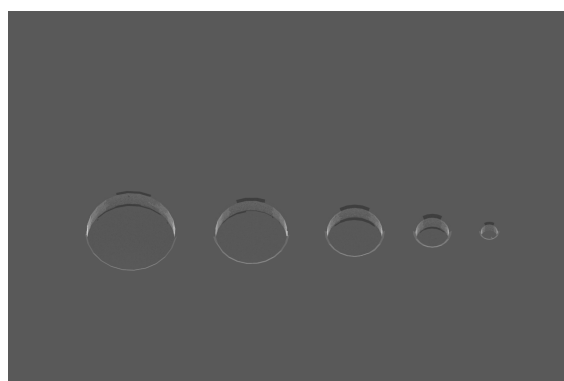
Obrázek 5.29: Náklon 10°



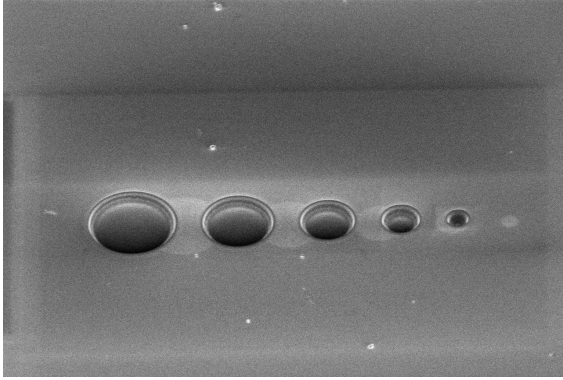
Obrázek 5.30: Výstup ze simulátoru



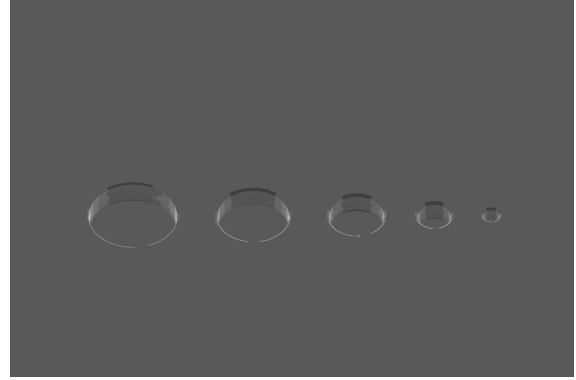
Obrázek 5.31: Náklon 30°



Obrázek 5.32: Výstup ze simulátoru



Obrázek 5.33: Náklon 0°



Obrázek 5.34: Výstup ze simulátoru

Poslední test tohoto druhu byl věnován obdobnému modelu, jako v předchozím případě, jen místo čtverečků byly vyhloubeny do silikonové destičky kruhy. Na všech snímcích z mikroskopu jsou viditelné světlé kruhy mezi jednotlivými prohlubněmi, obdobně jako u čtverců. Tato vlastnost je v simulátoru také zanedbána. Je to částečně způsobeno tím, že vyhloubené díry jsou tak hluboké, že v časovém úseku pro detekci elektronu uvnitř díry, ještě odpovídající sekundární elektron není detekovaný. Dochází k jeho pozdější detekci, právě v okamžiku, kdy je snímán prostor mezi dvěma kružnicemi. Tím dochází k zesvětlení daného prostoru.

Nejllepší hodnotu PSNR a MSE opět naměříme na simulovaném obrázku nakloněném o třicet stupňů. Můžeme pozorovat stejné jevy jako v předchozím případě.

Výsledky měření

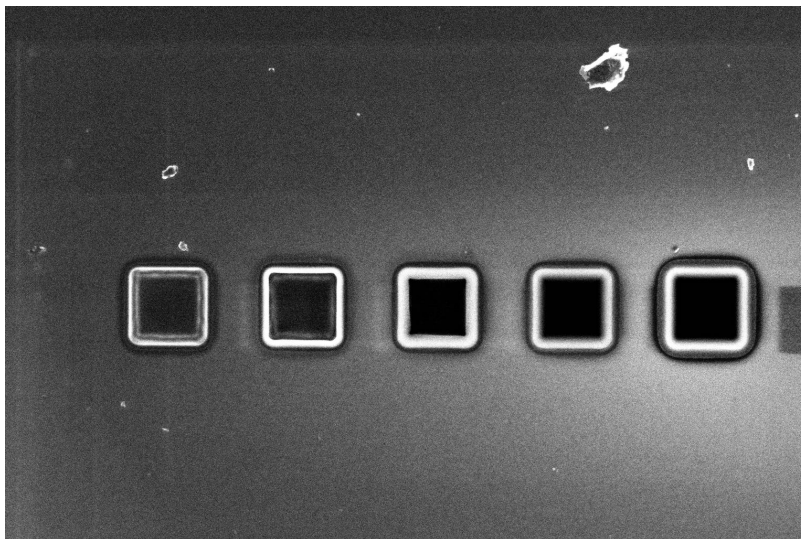
Úhel náklonu($^\circ$)	PSNR	SSIM	MSE	Doba generování
0	13.938	0.123	2626.036	~ 11 s
10	15.310	0.133	1914.513	~ 23 s
30	18.151	0.115	995.461	~ 84 s
45	17.156	0.193	1251.703	~ 92 s

Tabulka 5.4: Výsledky měření pro menší kruhy

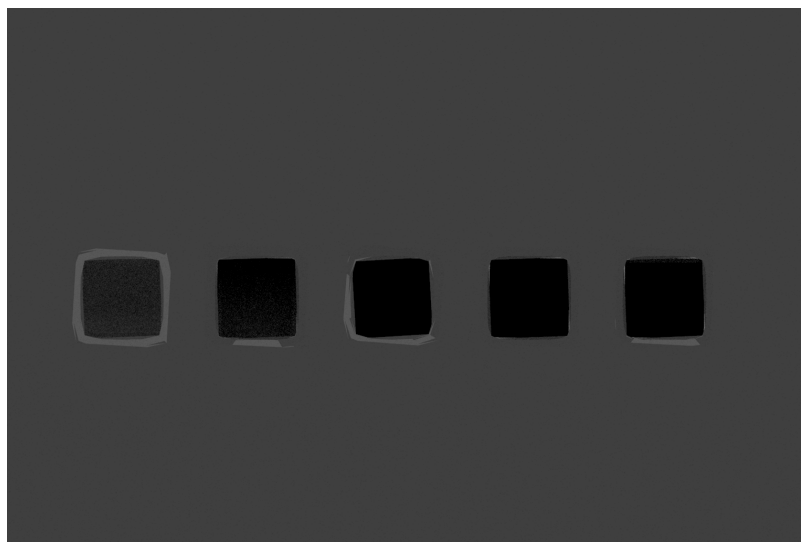
5.3.3 Vliv počtu vzorků na pixel

Cílem tohoto testu je zjistit, jaký vliv má na simulovaný výstup počet vzorků, které se použijí pro jeho vykreslení. Daný parametr lze nastavit v systému PBRTv3 ve vstupním souboru, při vytváření sampleru. Vliv se posoudí nejlépe podle metrik užitých v předcházejících experimentech.

Pro testování je využit obrázek 5.35, pro který jsou vygenerovány simulované obrázky postupně s jedním, osmi, šestnácti a dvaatřiceti vzorky na pixel. Naměřené hodnoty jsou k vidění v tabulce 5.5.



Obrázek 5.35: Výstup z mikroskopu



Obrázek 5.36: Výstup simulace, osm vzorků na pixel

Z tabulky měření můžeme vyčíst, že hustota vzorkování nehraje v konečném důsledku příliš velkou roli, co do výsledků srovnávacích metod. Největší rozdíl, který navýšení počtu

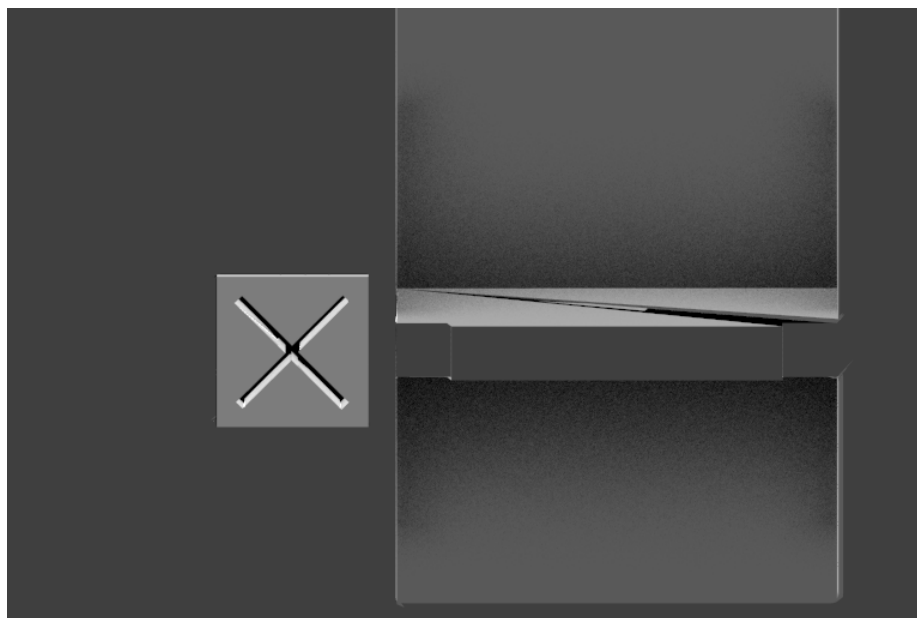
vzorků přináší, je zvýšení doby výpočtu. Tato doba naroste ještě o něco více, pokud zkoumáme vzorek, jehož hlavní rovina není kolmo pod zdrojem elektronového paprsku, tedy je generován větší počet elektronů. Při natočení vzorku na obrázku 5.36 o 45° a použití třiceti dvou vzorků na pixel zabere výpočet přibližně dvě minuty. To už se pohybujeme za přijatelnou hranicí.

Počet vzorků na pixel	PSNR	SSIM	MSE	Doba generování
1	13.350	0.170	3006.873	~1s
8	13.370	0.171	2992.616	~10-11s
16	13.373	0.172	2991.021	~20s
32	13.375	0.172	2989.711	~50-51s

Tabulka 5.5: Změna výsledků měření podle nastavení vzorkování

5.4 Výkon aplikace

V této sekci je popsán zátěžový test simulátoru. K testovacím účelům byl vytvořen model skládající se ze dvou částí s různým materiálem, které tvoří 676929 trojúhelníků. Model byl vytvořen na základě jednoduchého modelu, který byl zjemněn a vyhlazen pomocí funkcí v 3D editoru Blender. Vykreslení celého takhle velkého modelu zabere přibližně do 20 sekund, při rozlišení výstupu 768 na 512 pixelů. Při stejném nastavení rendereru a zvýšení rozlišení na 1920x1080 zabere vykreslení zhruba jednu minutu, což považuji za dobrý výsledek.



Obrázek 5.37: Výstup zátěžového testu, šestnáct vzorků na pixel

Testování probíhalo na pracovním notebooku Dell Latitude 5590, který byl osazen procesorem Intel Core i7 8650U na taktovací frekvenci 1.90GHz. Operační paměť činí 32GB a platforma Windows 10.

Využití zdrojů

PBRTr3 si vede podrobné statistiky o využití zdrojů. Poskytuje informace o vnitřní optimalizační struktuře, počtu vysledovaných paprsků a další údaje. Detailní výstup při vykreslení plnohodnotného HD obrázku je pro ukázkou níže.

Statistics:

BVH		
Interior nodes		385301
Leaf nodes		385302
Primitives per leaf node		676929 / 385302 (1.76x)
Integrator		
Camera rays traced		33177600
Intersections		
Regular ray intersection tests		555621325
Ray-triangle intersection tests		42784967 / 1870438045 (2.29%)
Memory		
BVH tree		33.85 MiB
Film pixels		63.28 MiB
Primitives		46.48 MiB
Texture MIP maps		0.02 kB
TransformCache		260.00 kB
Triangle meshes		59.65 MiB
Scene		
Lights		2
Materials created		2
Shapes created		676929
Probes per TransformCache lookup		1.000 avg [range 1 - 1]
TransformCache hits		3 / 6 (50.00%)
Triangles per triangle mesh		676929 / 2 (338464.50x)
Texture		
Trilinear lookups		4

Kapitola 6

Závěr

V práci se podařilo naimplementovat základní jednoduchou simulaci elektronového mikroskopu. Simulace bere v potaz úhel dopadu a počet emitovaných elektronů směrem od vzorku a jejich zachycení detektorem. To vše vytvořeno jako rozšíření vykreslovacího systému PBRT. Vstupní soubor simulátoru je validní soubor s koncovkou .pbrt, ve kterém je definovaný model a ostatní prvky scény. Simulátor je postaven jako rozšíření stávající funkcionality PBRTv3 a lze mezi způsobem vykreslování jednoduše přepínat přepsáním parametrů vstupního souboru. Jediné omezení je, že pro sledování elektronových paprsků pomocí „electron“ integrátoru je potřeba, aby byla kamera typu „ElectronGun“ a scéna obsahovala právě dvě světla, jedno globální a druhé s názvem „electronDetector“.

Práce má příslib i do budoucna. V simulaci je stále několik vlastností, které lze ještě zpřesnit a tím docílit více realistického zobrazení. Reálný mikroskop totiž zobrazuje ve výsledku hodně zašumělých bodů, které způsobují elektrony odražené se od zkoumaného vzorku vícekrát. Tento defekt se částečně eliminuje zkracováním doby svícení a snímání jednoho konkrétního vzorku. Čím kratší dobu generujeme primární elektrony, tím méně sekundárních elektronů se může projevit na výsledném pixelu. Stejně jako by simulátor měl pro realističnost zobrazovat tyto defekty snímku, dalo by se využít výstupu simulátoru pro odstranění těchto defektů ze snímků reálného mikroskopu.

6.1 Vlastnosti implementovaného rozšíření systému PBRTv3

Implementovaný simulátor dobře vypočítává dva zásadní jevy.

- Detector shadowing – místa vzorku, která jsou odvrácená vůči detektoru jsou v obraze tmavší, protože větší část SE se z daného místa nedostane až k detektoru
- Edge ringing – Čím ostřejší úhel dopadu, tím větší množství SE

Tím lze zvolenou metodu považovat za vhodnou a je možné uvažovat o simulování dalších pokročilejších jevů.

Kvalita 3D modelu je stěžejní

Pro získání opravdu pěkných obrazových materiálů ze simulátoru, je potřeba poskytnout na vstup dostatečně kvalitní a přesné modely. Aby takový model odpovídal skutečnosti, neměl by obsahovat téměř žádnou větší rovnou plochu. Je také potřeba dobře nadefinovat materiál pro každý odlišný prvek vzorku.

Proto se zde nabízí nějaká forma zjednodušení přípravy 3D modelů. Systém by bylo možné rozšířit o nástroj pro poloautomatizovanou přípravu 3D modelů. Program by očekával na vstupu jednoduchý model, který by se pomocí teselace a deformací vyhladil, aby se odstranily ostré hrany a přechody.

Samotná implementace by mohla být poměrně jednoduchá, všechny operace jsou optimalizované na GPU. V případě užití grafické karty by byla i výsledná rychlost dostačující. Alternativně by se místo klasických trojúhelníkových modelů dalo použít jiné reprezentace vstupních dat, například pomocí voxelů.

System PBRTv3 je dobře optimalizován

Výsledná zátěžová zkouška simulace ukázala, že PBRTv3 nemá potíže s elektronovou simulací ani přes vyšší počet trojúhelníků. Je nutné přiznat, že výsledná simulace je hodně zjednodušená a s dalšími simulovanými vlastnostmi by časová náročnost opět o něco vzrostla.

6.2 Hodnocení práce ve firmě

Tato práce byla vypracována pro firmu Thermo Fisher Scientific Brno (TFS Brno). Výsledky jsou firmou hodnoceny jako přínosné, kdy ukazují, že pomocí sledování paprsků lze generovat obrázky s typickými znaky pro skenovací elektronové mikroskopy.

Firemní hodnocení také poukazuje na některé nedostatky. Největším problémem jsou nežádoucí artefakty na výstupu simulátoru, které jsou způsobeny nekvalitním modelem. Řešení tohoto problému by mělo být ve vytvoření kvalitnějších modelů.

Literatura

- [1] Chlebek, M.: Měření kvality obrazu a video signálů [online]. 2011 [cit. 2019-05-16].
URL <https://theses.cz/id/8r5kh0/>
- [2] Corporation, I.: *Intel Embree*. Intel Corporation, Leden 2019, [Online; navštíveno 15.01.2019].
URL <https://www.embree.org/>
- [3] Jakob, W.: Mitsuba renderer. 2010, [Online; navštíveno 15.04.2019].
URL <http://www.mitsuba-renderer.org>
- [4] Matt Pharr, W. J.; Humphreys, G.: *Physically based rendering: from theory to implementation*. pbr-book.org, Duben 2019, [Online; navštíveno 15.04.2019].
URL http://www.pbr-book.org/3ed-2018/Light_Sources/Light_Interface.html
- [5] Matt Pharr, W. J.; Humphreys, G.: *System implementation and internals*. PBRT, Duben 2019, [Online; navštíveno 15.04.2019].
URL https://www.pbrt.org/faqs.html#extending_plugin
- [6] Novák, L.: *Metody kvantifikace detekční trasy sekundárních elektronů v rastrovacím elektronovém mikroskopu [online]*. Doctoral theses, dissertations, Masaryk University, Faculty of Science, Brno, 2011 [cit. 2019-01-15].
URL <https://theses.cz/id/nbp4rm/>
- [7] nVidia: *NVIDIA® OptiX™ 6.0*. nVidia, Duben 2019, [Online; navštíveno 15.04.2019].
URL https://raytracing-docs.nvidia.com/optix_6_0/index.html
- [8] nVidia: *NVIDIA® OptiX™ Ray Tracing Engine*. NVIDIA, Leden 2019, [Online; navštíveno 15.01.2019].
URL <https://developer.nvidia.com/optix>
- [9] nVidia: *nVidia® RTX™ platform*. nVidia, Duben 2019, [Online; navštíveno 15.04.2019].
URL <https://developer.nvidia.com/rtx>
- [10] Pharr, M.: *Physically based rendering : from theory to implementation*. Cambridge, MA: Morgan Kaufmann, third edition vydání, 2017, ISBN 978-0-12-800645-0.
- [11] Rosebrock, A.: *How-To: Python Compare Two Images*. pyimagesearch.com, Zář 2014, [Online; navštíveno 15.05.2019].
URL <https://www.pyimagesearch.com/2014/09/15/python-compare-two-images/>

- [12] Seiler, H.: Secondary electron emission in the scanning electron microscope. *Journal of Applied Physics*, ročník 54, 1983: s. R1–R18, ISSN 0021-8979.
URL <http://search.proquest.com/docview/23352875/>
- [13] Sundstedt, V.; Gutierrez, D.; Anson, O.; aj.: Perceptual rendering of participating media. *ACM Transactions on Applied Perception (TAP)*, ročník 4, č. 3, 2007: s. 15–es, ISSN 1544-3965.
- [14] Vítek, J.: *Intel Xe budou obsahovat hardware pro ray tracing*. SvětHardware, Květen 2019, [Online; navštíveno 7.05.2019].
URL <https://www.svethardware.cz/intel-xe-budou-obsahovat-hardware-pro-ray-tracing/49233>

Příloha A

Obsah přiloženého paměťového CD

```
CD:\
Code\ -- Zdrojové soubory aplikace
Doc\src\ -- Zdrojové soubory dokumentace
TestData\Images\ -- Vzorové výstupy pro jednotlivé typy mikroskopu
TestData\obj\ -- 3D modely použité pro generování PBRT souborů
                createPBRTModels.bat -- script pro vygenerování modelů ve formátu P
                                Výstup uložen do TestData\pbrt\model
TestData\original\ -- 3D modely z programu Blender.
TestData\pathTracingComparation\ -- pbrt model použitý pro porovnání Pathtracing
TestData\pbrt\ -- zdrojové scény pro testování
                pbrt.exe -- sestavení PBRTv3 rozšířené o ElectronTracing
                GenerateOutput.bat -- vygeneruje výstup do složky .\output
TestData\performance\chunkBeveled -- velký model pro test výkonu
TestData\pythonComparation -- script pro porovnání výstupů.
```

Příloha B

Manuál

B.1 Instalace a sestavení

Výsledné zdrojové soubory jsou umístěny v adresáři:

```
CD:\Code
```

Jedná se o standardní cmake projekt, sestavení bylo testováno pouze na platformě windows. Veškeré potřebné závislosti pro PBRT jsou integrovány přímo do projektu.

B.2 Použití

B.2.1 Tvorba modelu

Nejdůležitější část, model musí splňovat požadavky na soubor s koncovkou pbrt. V něm je potřeba definovat vertexy, normály a indexy vrcholů tvořící jeden trojúhelník. Nejsnazší způsob, jak tyto hodnoty získat, je využitím pbrt nástroje obj2pbrt.exe, který převede model na požadovaný výstup. K němu je pak třeba doplnit definici scény.

B.2.2 spuštění

```
$pbrt.exe [--outputFile <outputFile>] <pbrtFile>
```

Za předpokladu, že nezadáme žádný parametr, očekává pbrt.exe vstupní soubor na standardním vstupu. Výstupní soubor lze pak definovat přímo ve vstupním souboru.

Příloha C

Ukázka vstupního souboru

Ve vstupním souboru musí být definovány následující objekty

```
LookAt 0 10 0 # eye
        0 0 0 # look at point
        0 0 1 # up vector

Camera "electronGun" "float fov" 45

Sampler "maxmindist" "integer pixelsamples" 8

Integrator "electron" "string interpolationMethod" "nStepInterpolation"

Film "image" "string filename" "simple.png"
        "integer xresolution" [1536] "integer yresolution" [1024]

WorldBegin

LightSource "infinite" "rgb L" [.0 .0 .0]
LightSource "electronDetector" "point from" [0, 0, 10]
```

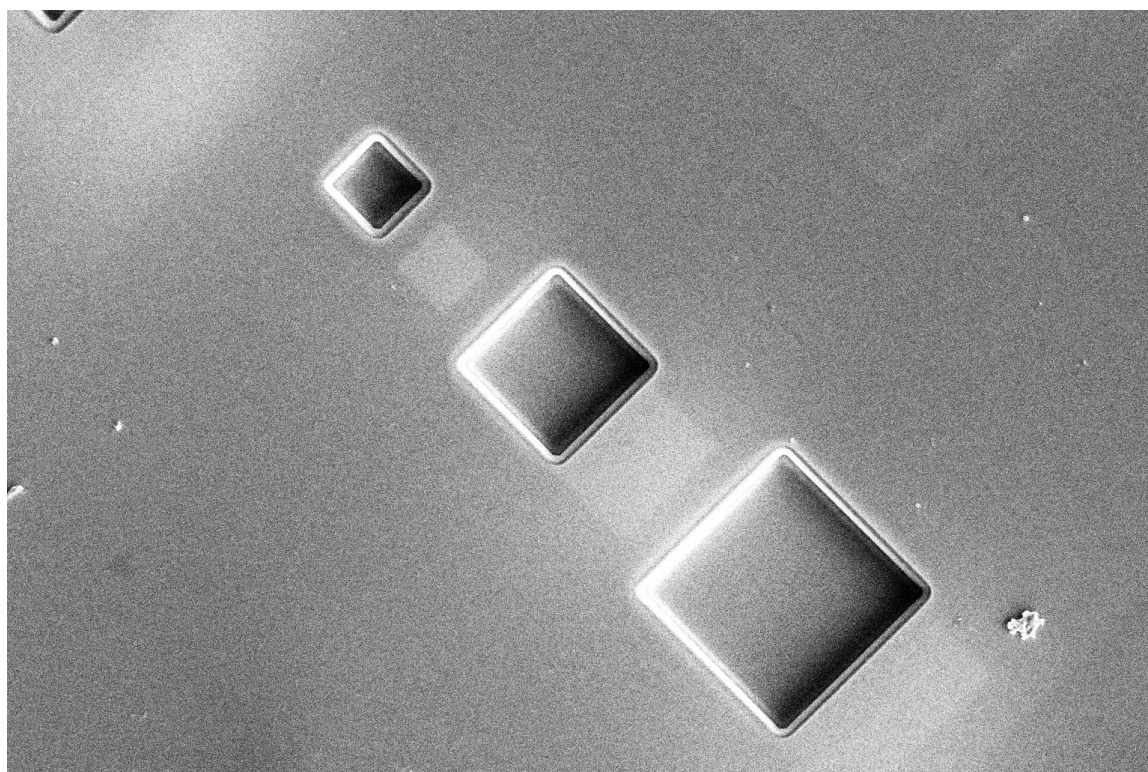
Mezi WorldBegin a WorldEnd je pak samotná definice scény.

```
AttributeBegin
Material "silicon"
Shape "trianglemesh"
  "point3 P" [
-0.2231969982 0.01089600008 -0.2345999926
0.3451279998 -0.2097599953 -0.2345999926
...
0.01501499955 -0.0101859998 0.1709109992
0.01444299985 -0.01012399979 0.1637720019
]
  "normal N" [
0 0 -1
0 0 -1
...
0.05099999905 0.9987000227 0.004499999806
0.05099999905 0.9987000227 0.004499999806
]
  "integer indices" [
0 1 2
4 5 6
...
6546 6547 6548
6546 6548 6549
]
AttributeEnd
WorldEnd
```

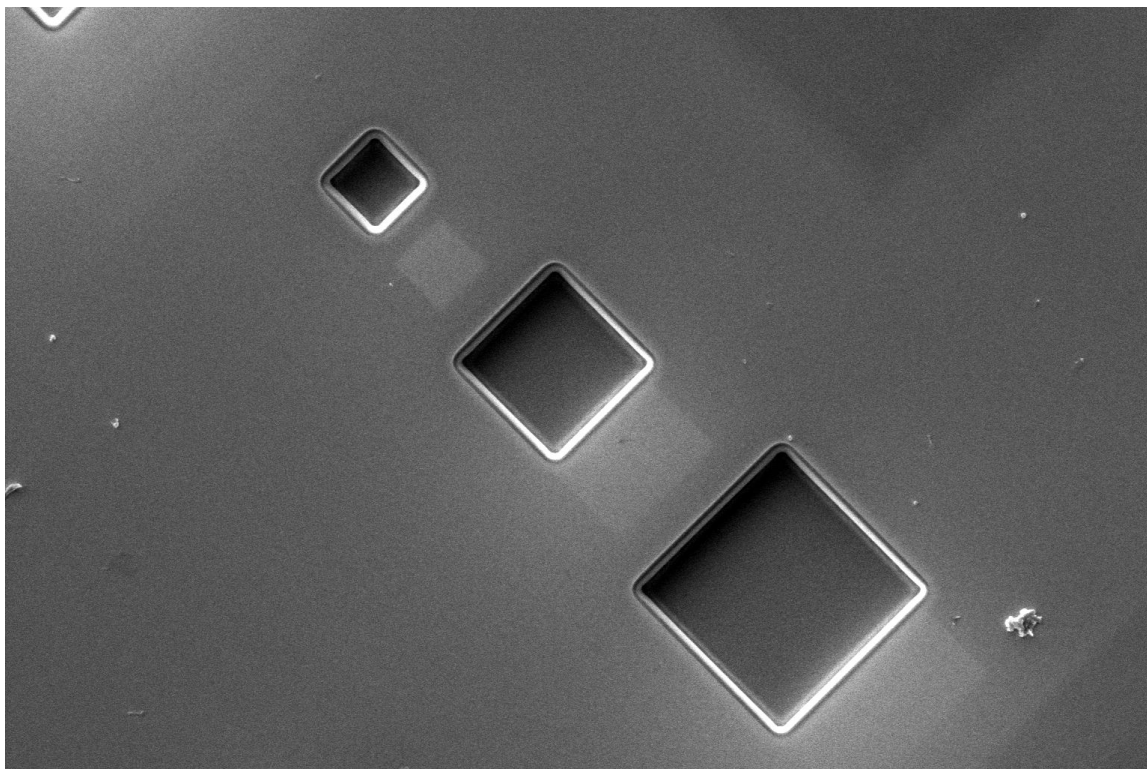
Příloha D

Obrazové přílohy

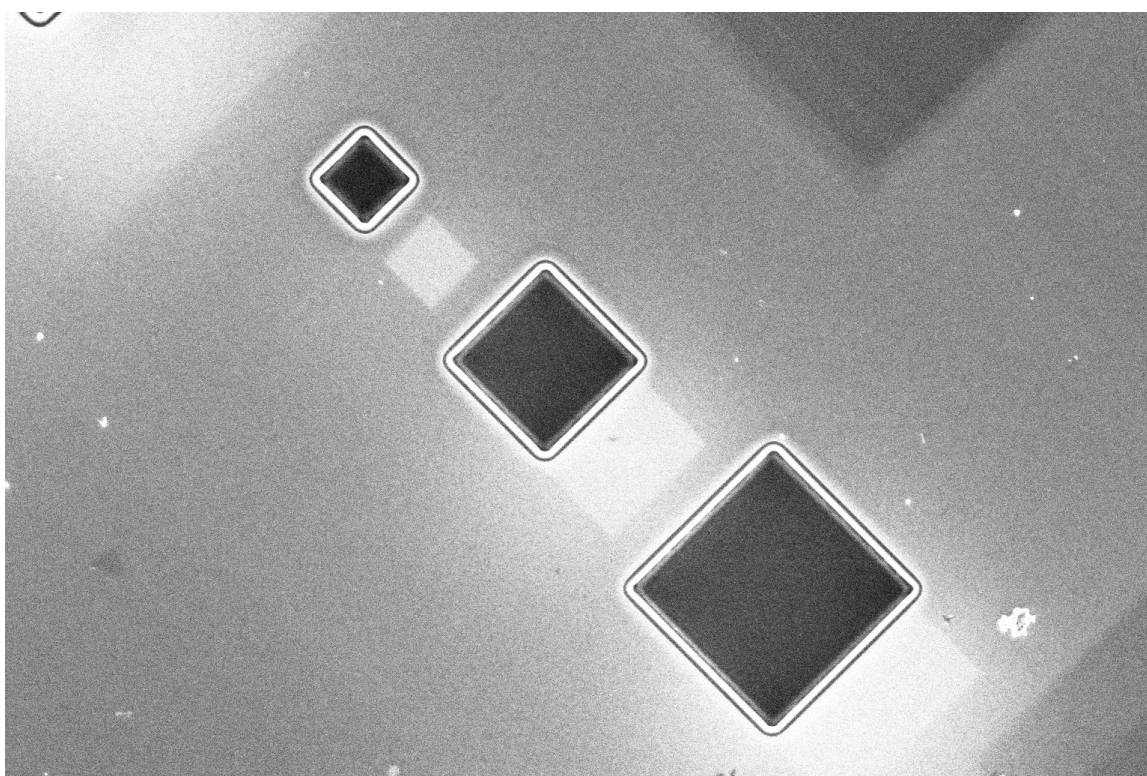
D.1 Systém TLD



Obrázek D.1: Princip elektronových mikroskopů

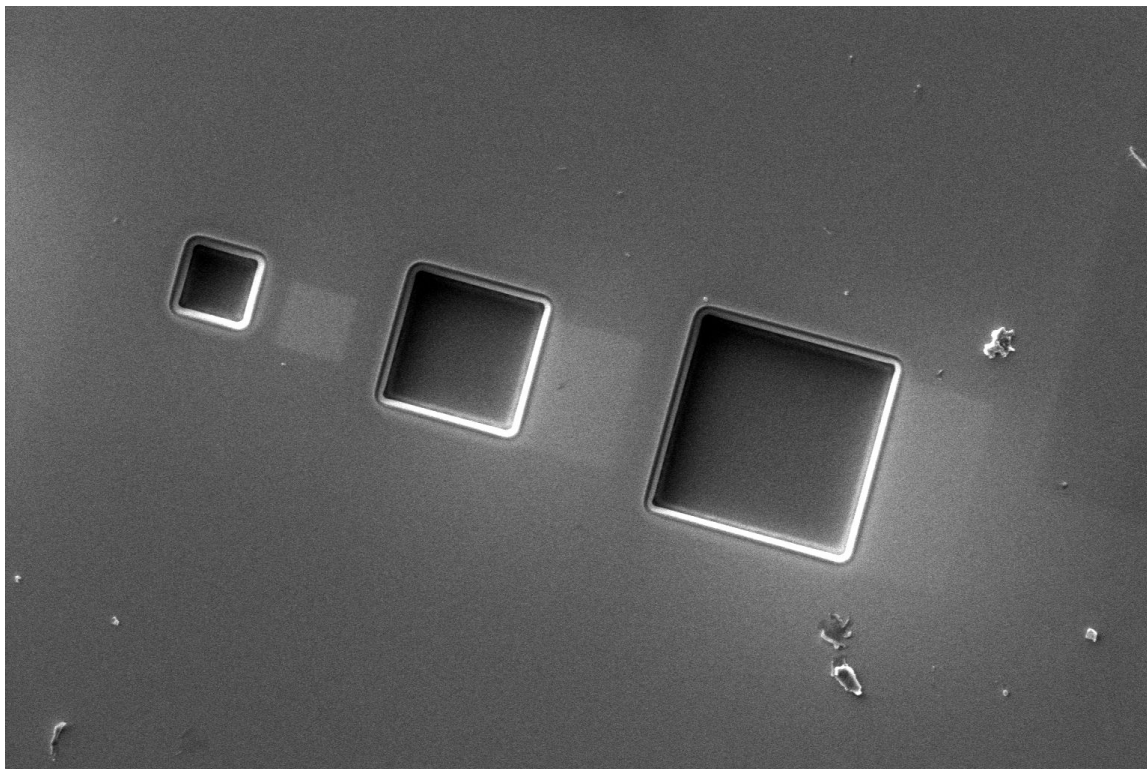


Obrázek D.2: Princip elektronových mikroskopů

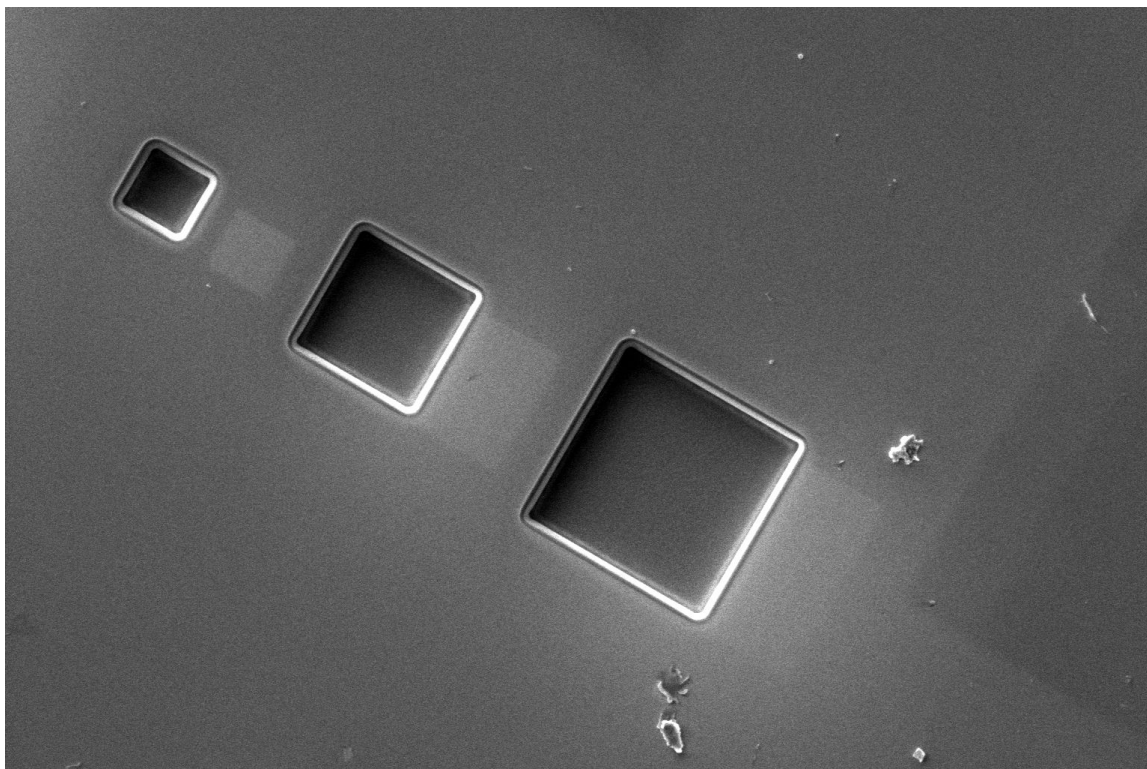


Obrázek D.3: Princip elektronových mikroskopů

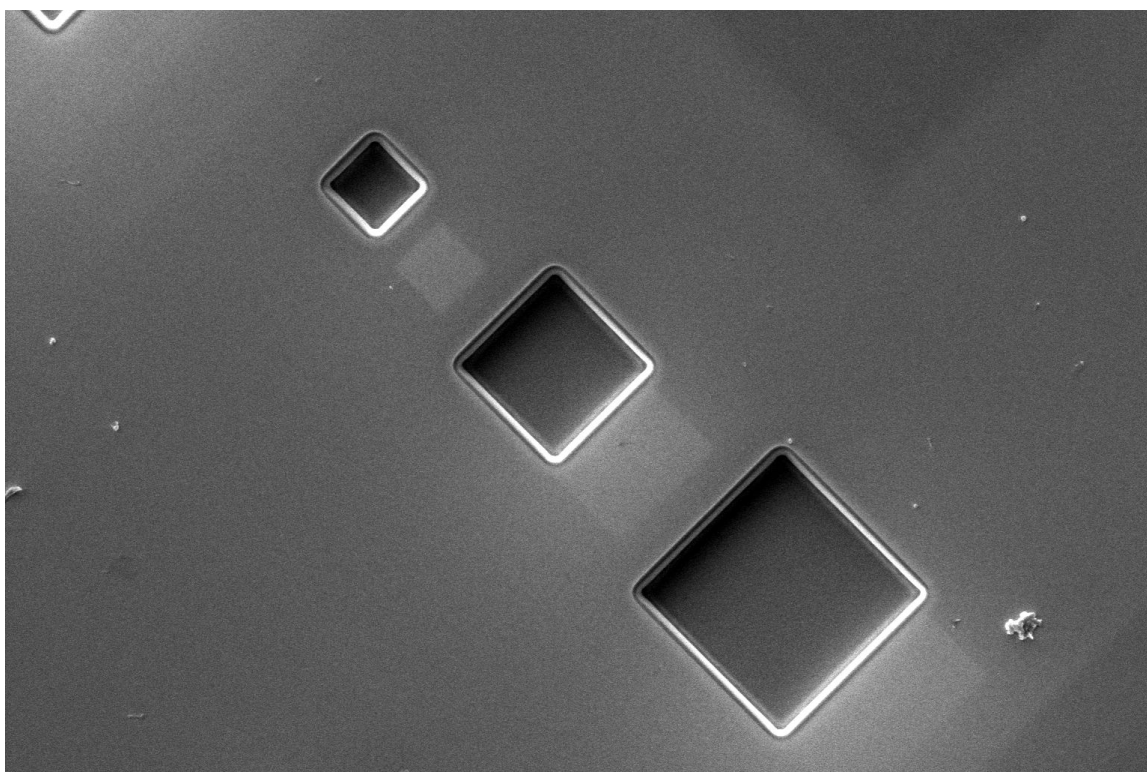
D.2 Systém ICE



Obrázek D.4: Princip elektronových mikroskopů

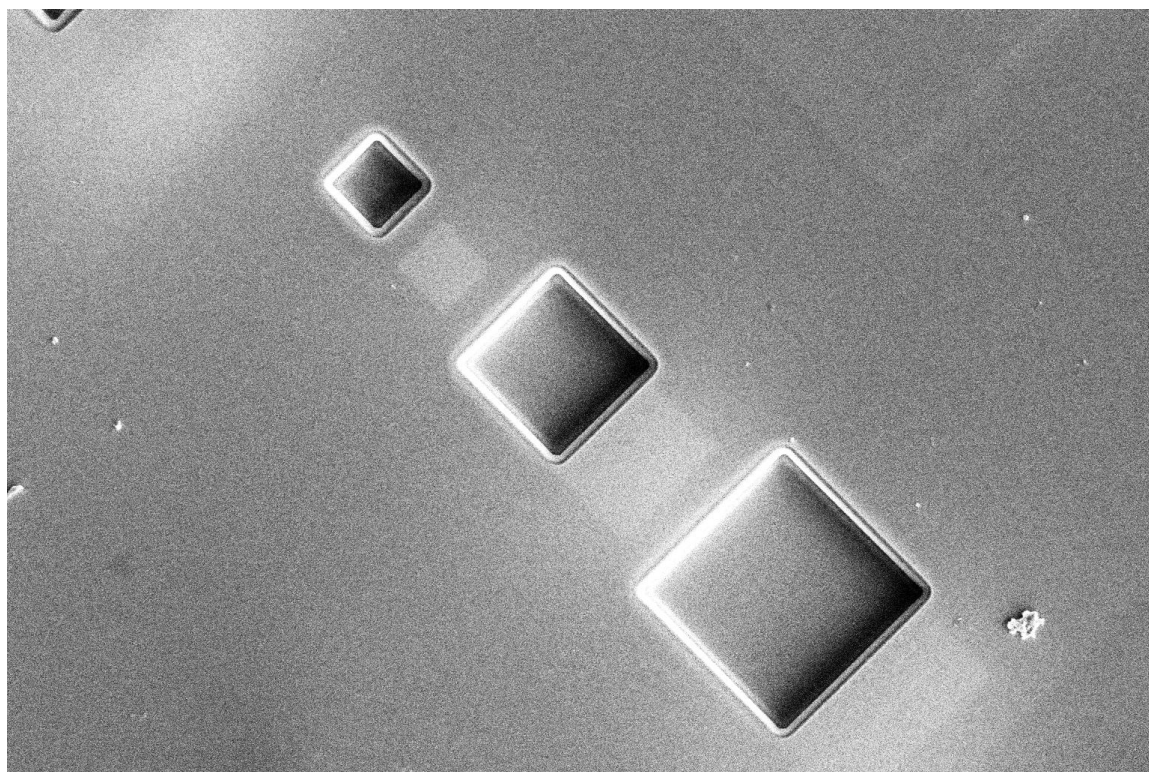


Obrázek D.5: Princip elektronových mikroskopů

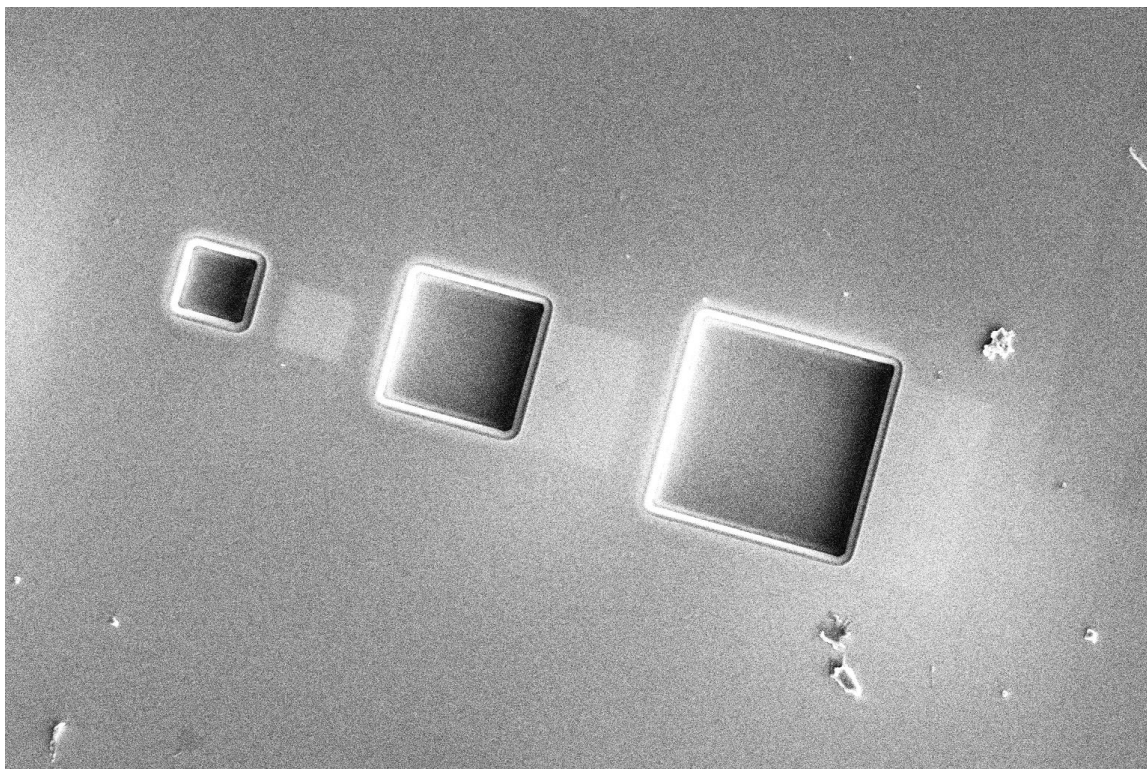


Obrázek D.6: Princip elektronových mikroskopů

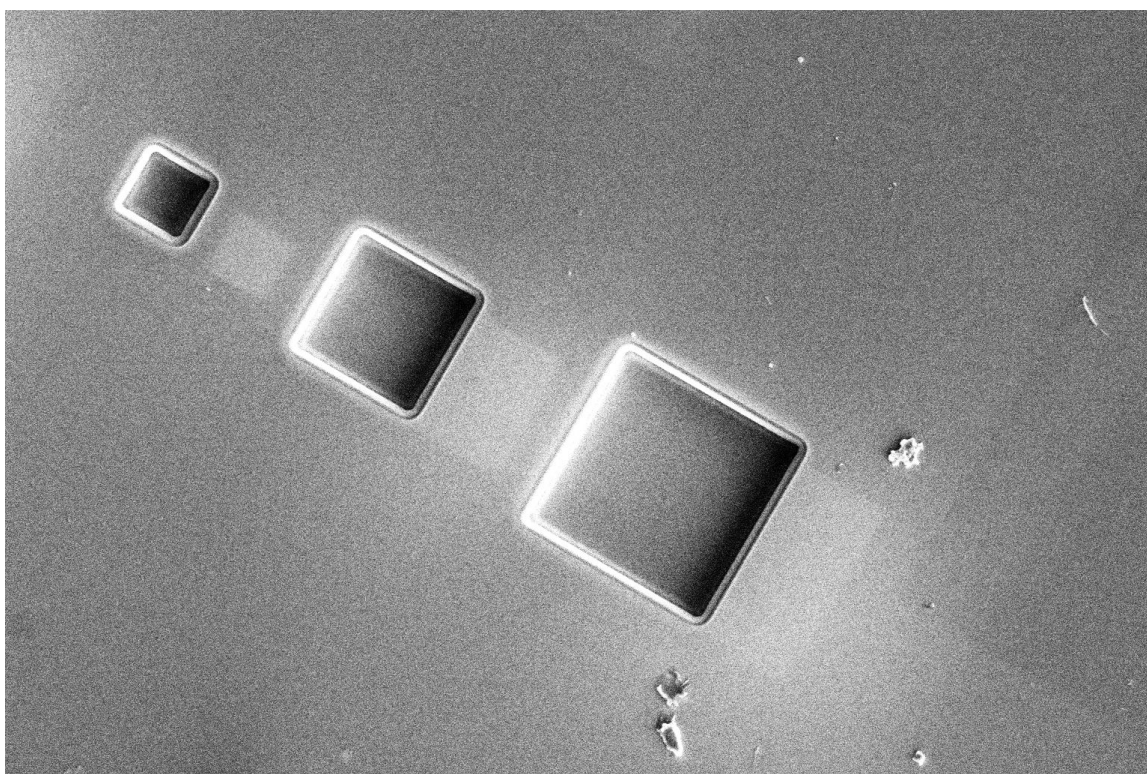
D.3 Systém ETD



Obrázek D.7: Princip elektronových mikroskopů



Obrázek D.8: Princip elektronových mikroskopů



Obrázek D.9: Princip elektronových mikroskopů