



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

KVANTITATIVNÍ DIGITÁLNÍ HOLOGRAFICKÁ MIKROSKOPIE POMOCÍ STROJOVÉHO UČENÍ

QUANTITATIVE DIGITAL HOLOGRAPHIC MICROSCOPY USING MACHINE LEARNING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Martin Duša

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Tomáš Vičar

BRNO 2022

Diplomová práce

magisterský navazující studijní program **Bioinženýrství**

Ústav biomedicínského inženýrství

Student: Bc. Martin Duša

ID: 203212

Ročník: 2

Akademický rok: 2021/22

NÁZEV TÉMATU:

Kvantitativní digitální holografická mikroskopie pomocí strojového učení

POKyny PRO VYPRACOVÁNÍ:

1) Proveďte literární rešerši použití strojového učení pro analýzy obrazu pro získání kvantitativních informací z digitální holografické video mikroskopie levitujících mikro a nano částic. 2) Navrhněte optimální metodiku automatizovaného vyhodnocení mikroskopických snímků levitujících mikro a nanočástic pomocí strojového učení. 3) Zaměřte se na přesné určování polohy jedné a více částic ve snímku. Dále se zaměřte na návrh metodiky určení rozměrů a hodnot indexu lomu z digitálního hologramu. 4) Vyjděte z program Deeptrack 2.0 a otestujte metodiku na předem vypočítaných holografických snímcích a navrhněte metodiku hodnocení přesnosti a rychlosti určování hledaných parametrů. 5) Otestujte metodiku na předem vypočítaných holografických snímcích s různými optickými vadami zobrazovací soustavy. 6) Otestujte metodiku na reálných obrazových datech a porovnejte hodnoty určených parametrů s jinými metodami používanými na ÚPT AVČR. 7) Proveďte diskusi získaných výsledků a zhodnoťte účinnost a využitelnost řešení.

DOPORUČENÁ LITERATURA:

- [1] Benjamin Midtvedt, Saga Helgadottir, Aykut Argun, Jesús Pineda, Daniel Midtvedt, Giovanni Volpe, Quantitative Digital Microscopy with Deep Learning, Applied Physics Reviews, 8, 011310 (2021)
- [2] Benjamin Midtvedt, Erik Olsén, Fredrik Eklund, Fredrik Höök, Caroline Beck Adiels, Giovanni Volpe, Daniel Midtvedt, Fast and Accurate Nanoparticle Characterization Using Deep-Learning-Enhanced Off-Axis Holography, ACS Nano 15(2), 2240–2250 (2021)

Termín zadání: 7.2.2022

Termín odevzdání: 20.5.2022

Vedoucí práce: Ing. Tomáš Vičar

doc. Ing. Radim Kolář, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práce představuje metody strojového učení pro určení parametrů mikro a nano částic ze snímků digitální holografické mikroskopie. V teoretické části jsou přiblíženy principy snímání hologramu, holografické mikroskopie a podobnost mezi Mie teorií a hologramem. Druhá část teoretické rešerše je věnována metodám strojového učení využitých při určování kvantitativních informací částic. Praktická část se věnuje návrhu postupu pro určení pozice, indexu lomu a poloměru pomocí architektury U-Net implementované v prostředí PyTorch a DeepTrack 2.1. V závěru práce jsou diskutovány výsledky navržených metodik.

KLÍČOVÁ SLOVA

DHM, Hologram, U-Net, Mie teorie, PyTorch, DeepTrack

ABSTRACT

This thesis presents machine learning methods for determining the parameters of micro and nano particles from digital holographic microscopy images. In the theoretical part the principles of hologram imaging, holographic microscopy and the similarity between Mie theory and hologram are presented. The second part of the theoretical review is devoted to machine learning methods used in determining the quantitative information of particles. The practical part is focused on the design of a procedure for determining the position, refractive index and radius using the U-Net architecture implemented in PyTorch and DeepTrack 2.1. The results of the proposed methodologies are discussed at the end of the paper.

KEYWORDS

DHM, Hologram, U-Net, Mie Theory, PyTorch, DeepTrack

DUŠA, Martin. *Kvantitativní digitální holografická mikroskopie pomocí strojového učení*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2022, 80 s. Diplomová práce. Vedoucí práce: Ing. Tomáš Vičar

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Martin Duša
VUT ID autora:	203212
Typ práce:	Diplomová práce
Akademický rok:	2021/22
Téma závěrečné práce:	Kvantitativní digitální holografická mikroskopie pomocí strojového učení

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

* Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Touto cestou bych rád poděkoval vedoucímu mé diplomové práce panu Ing. Tomáši Vičarovi za odborné vedení, užitečné rady, trpělivost a strávený čas nad mojí prací. Dále bych chtěl poděkovat mému konzultantovi Mgr. Otovi Brzobohatému Ph.D za rady a pozitivní přístup. Závěrem bych chtěl poděkovat své rodině, přítelkyni a kamarádům za psychickou i jinou podporu během zpracovávání mé závěrečné práce a celého studia.

Obsah

Úvod	12
1 Holografická mikroskopie	13
1.1 Optická holografie	13
1.1.1 Základní koncept hologramu	13
1.1.2 Superimpozice objektu a referenčního paprsku	13
1.1.3 Záznam hologramu	14
1.1.4 Proces rekonstrukce hologramu	15
1.2 Digitální holografie	16
1.3 Digitální holografická mikroskopie	18
1.3.1 Soustavy digitální holografická mikroskopie	18
1.3.2 Použitá soustava v experimentu	19
1.4 Paralela mezi Mie teorií a hologramem částice	20
1.4.1 Mie teorie	20
1.4.2 Hologram částice a Mie teorie	21
2 Zpracování obrazu pomocí strojového učení	24
2.1 Umělé neuronové sítě a hologram částic	24
2.1.1 Úvod do neuronových sítí	24
2.1.2 Umělé neuronové sítě v holografické mikroskopii	25
2.2 Konvoluční neuronové sítě	26
2.2.1 Základy pro konvoluční neurální sítě	26
2.2.2 Využití konvolučních sítí při určování parametrů částice	26
2.3 U-Net	29
2.3.1 Použití U-Net pro 3D rekonstrukci polohy částice	30
2.4 Klasifikátory	31
2.4.1 Metoda podpůrných vektorů v digitální holografické mikroskopii	31
3 Návrh metodiky pro určení parametrů částic pomocí strojového učení	33
3.1 Deeptack 2.1	33
3.1.1 Simulace optického systému	33
3.1.2 Obraz rozptylu světla na objektu	34
3.1.3 Generování a úprava dat	34
3.1.4 Aberované obrazy	38
3.1.5 Modely a trénování v rámci Deeptack 2.1	38
3.2 Návrh metodiky pro určení parametrů částic	39

3.2.1	Předzpracování a vytvoření trénovacích dat	40
3.2.2	Návrh a optimalizace topologie U-Net pro určení parametrů	41
3.2.3	Navrhnutí procesu trénování modelu	44
3.3	Výsledky optimalizace modelu U-Net na aberovaných datech	45
3.3.1	Výsledky optimalizace pro určení 2D polohy	46
3.3.2	Výsledky optimalizace pro určení polohy z	47
3.3.3	Výsledky optimalizace pro určení poloměru	49
3.3.4	Výsledky určení indexu lomu	50
3.4	Výsledky optimalizace modelu U-Net na neaberovaných datech	51
3.4.1	Výsledky pro určení 2D polohy	51
3.4.2	Výsledky určení polohy z	53
3.4.3	Výsledky určení poloměru	54
3.4.4	Výsledky určení indexu lomu	55
3.5	Trénování modelu v Deeptack 2.1.	56
3.5.1	Trénování modelu pro určení 2D pozice	56
3.5.2	Trénování modelu pro určení polohy v ose z	57
3.5.3	Trénování modelu pro určení poloměru	58
3.5.4	Trénování modelu pro určení indexu lomu	59
4	Testování modelů a aplikace na experimentální data	61
4.1	Aplikace modelů na data s aberacemi a bez aberace	61
4.1.1	Určení 2D pozice	61
4.1.2	Určení polohy z pozice	63
4.1.3	Určení poloměru	65
4.1.4	Určení indexu lomu	67
4.2	Aplikace modelů na experimentální data	69
	Závěr	75
	Literatura	76

Seznam obrázků

1.1	Schéma znázorňující Superimpozice referenční vlny a objektu.	14
1.2	Proces vytvoření hologramu	15
1.3	Schéma rekonstrukce hologramu	16
1.4	In-line DHM	18
1.5	Off-axis DHM	19
1.6	Sestava použitá pro snímání experimentálních snímků	20
1.7	Vliv velikosti částice na Mie rozptyl.	21
1.8	Porovnání reálné a fitované radiální symetrie hologramu [23]	22
2.1	Určení poloměru částic pomocí ANN. [13]	25
2.2	Architektura konvoluční neuronové sítě	26
2.3	Schéma architektury CATCH	28
2.4	Schéma WAC-NET	29
2.5	Topologie sítě U-Net představena v [28]	30
2.6	Určení parametrů částice pomocí SVM. [14]	32
3.1	Vytvořený obrázek pomocí Deeptrack 2.0	35
3.2	Porovnání obrazů s původními hodnotami.	36
3.3	Porovnání obrazů s korekční konstantou.	37
3.4	Graf SSMI indexu v závislosti na ose z obrazu.	37
3.5	Typy aberací	38
3.6	Příklad trénovacích dat.	40
3.7	Příklad masky a hloubkové mapy	41
3.8	Příklad map hodnot	41
3.9	Obecný diagram topologie Unet	42
3.10	Diagram postupu ladění hyperparametrů	44
3.11	Shrnutí procesu učení a validace diagramem	45
3.12	Průběh učení modelu pro určení 2D polohy učený na aberovaných datech.	47
3.13	Průběh učení modelu pro určení polohy z učený na aberovaných datech.	48
3.14	Průběh učení modelu pro určení poloměru učený na aberovaných datech.	50
3.15	Průběh učení modelu pro určení indexu lomu učený na aberovaných datech.	51
3.16	Průběh učení modelu učených na datech bez aberací pro určení 2D polohy.	52
3.17	Průběh učení modelu učených na neaberovaných datech pro určení polohy z.	54
3.18	Průběh učení modelu pro určení poloměru.	55
3.19	Průběh učení modelu pro určení indexu lomu.	56

3.20	Průběh učení modelu DT pro určení polohy z.	57
3.21	Ukázka predikce modelu DT určující polohu z.	57
3.22	Průběh učení modelu DT pro určení poloměru.	58
3.23	Ukázka predikce modelu DT určující poloměr.	59
3.24	Průběh učení modelu DT pro určení indexu lomu.	60
3.25	Ukázka predikce modelu DT určující index lomu.	60
4.1	Vliv typu aberace na model trénovaný na aberovaných datech pro určení 2D pozice.	61
4.2	Příklad falešně pozitivní detekce pozice.	62
4.3	Vliv typu aberace na model trénovaný na neaberovaných datech pro určení 2D pozice.	63
4.4	Vliv typu aberace na model trénovaný na aberovaných datech pro určení polohy z.	64
4.5	Vliv typu aberace na model trénovaný na neaberovaných datech pro určení polohy z.	64
4.6	Vliv typu aberace na DT modelu určující polohu z.	65
4.7	Vliv typu aberace na model trénovaný na aberovaných datech pro určení poloměru.	66
4.8	Vliv typu aberace na model trénovaný na neaberovaných datech pro určení poloměru.	66
4.9	Vliv typu aberace na DT modelu určující poloměr.	67
4.10	Vliv typu aberace na model trénovaný na aberovaných datech pro určení indexu lomu.	68
4.11	Vliv typu aberace na model trénovaný na neaberovaných datech pro určení indexu lomu.	68
4.12	Vliv typu aberace na DT model určující index lomu.	69
4.13	Subtrakce nahraného pozadí bez částice od snímku s částicí.	70
4.14	Aplikace modelu učený na aberovaných datech pro určení 2D polohy.	70
4.15	Aplikace modelu učený na neaberovaných datech pro určení 2D polohy.	71
4.16	Příklady detekce pozice z v experimentálních datech.	72
4.17	Příklady detekce poloměru v experimentálních datech.	73
4.18	Příklady detekce indexu lomu v experimentálních datech.	74

Seznam tabulek

3.1	Parametry navrženého optického systému.	33
3.2	Parametry navržené částice.	34
3.3	Přehled hyperparametrů modelu a optimalizátoru pro trénování. . . .	43
3.4	Parametry modelů pro určení 2D pozici.	46
3.5	Chyby modelů pro určení 2D pozici.	46
3.6	Parametry modelů pro určení polohy z učených na aberovaných datech.	48
3.7	Chyby modelů pro určení z polohy.	48
3.8	Parametry modelů pro určení poloměru učených na aberovaných datech.	49
3.9	Chyby modelů pro určení poloměru.	49
3.10	Parametry modelů pro určení indexu lomu učených na aberovaných datech.	50
3.11	Chyby modelů pro určení indexu lomu.	50
3.12	Parametry modelů pro určení 2D pozici.	51
3.13	Chyby modelů učených na datech bez aberaci pro určení 2D pozici. .	52
3.14	Parametry modelů pro určení polohy z na neaberovaných datech uče- ných na aberovaných.	53
3.15	Chyby modelů pro určení polohy z učených na neaberovaných.	53
3.16	Parametry modelů pro určení poloměru učených na neaberovaných datech	54
3.17	Chyby modelů pro určení poloměru.	54
3.18	Parametry modelů pro určení indexu lomu učených na neaberovaných.	55
3.19	Chyby modelů pro určení indexu lomu.	55
4.1	Porovnání vytvořených modelů s metodou z UTP AV ČR pro určení z polohy.	71
4.2	Porovnání vytvořených modelů s metodou z UTP AV ČR pro určení poloměru	72
4.3	Porovnání vytvořených modelů s metodou z UTP AV ČR pro určení indexu lomu	73

Úvod

Digitální holografická mikroskopie je nově objeveným a efektivním nástrojem používaný pro práci s daty pocházející z fyzikální sféry až po biomedicínské data. Po uplynutí 20 let od sestavení prvního prototypu přechází digitální mikroskopie z výzkumné části do praxe, kde nachází uplatnění ve vypočetní tomografii (CT) a automatické průtokové cytometrii. Po tuto dobu bylo sepsáno nespočet přístupů pro analýzu holografických obrazů. Například využití algoritmů založených na Fourierovské transformaci či radiální symetrii pozorovaných částic. S rozmachem metod hlubokého učení jako počítačového vidění, segmentace a rozpoznávání obrazů je přístup pro extrakci parametrů částice účinnější a jednodušší na implementaci. [5]

Důvodem pro extrakci parametrů mikroskopických částic je v mnoha experimentech využita jako lokální sonda okolního prostředí. Jmenovitě pro kalibrace optické pinzety, měření molekulárních sil či pozorování růstu koloidních krystalů. [29] Zároveň nanočástice zaujímají důležitou složku několika odvětví, a to přes farmacii, potravinářství a enviromentální sledování. Velikost částice a složení ovlivňuje chování a funkce částice je nutná a rychlá a přesná charakterizace těchto částic. Ve farmaceutickém průmyslu při interakci mezi buňkami a nanočásticemi (např. viry, extracelulární vezikuly atd) je do značné míry ovlivněno velikostí částic a jejich složení. Zároveň můžeme přidat i další parametr jako index lomu, podle něhož je možné odvodit kompozici částice. [25]

Cílem práce je představení metod strojového učení pro kvantitativní digitální holografické mikroskopie a implementace metod hlubokého učení pro určení parametrů částic z holografických snímků.

Ze začátku práce jsou stručně popsány principy hologramu částic a jejich podobnost s Mie-Lorenz teorií. Dále uvedení digitální holografické mikroskopie, její standartní konfigurace a použité sestavy pro pořízení experimentálních dat. Na tuto část navazuje přehled metod strojového učení používaných pro určení parametrů částic, jako pozice v prostoru, velikost a index lomu. Každá podkapitola obsahuje stručný teoretický popis a použité konkrétní problémové řešení. Z té poté vychází návrh metodik pro určení parametrů a popis použitého programového řešení. Pro návrh a implementaci byl použit aplikační rámec Deeprack 2.1. a PyTorch. Výsledky implementací jsou porovnány mezi sebou a diskutovány. V závěru práce jsou metodiky aplikovány na experimentální data poskytnuté ÚTP AV ČR. Po aplikaci jsou diskutovány reálné využití metod.

1 Holografická mikroskopie

Při klasickém snímání scény pomocí fotografie je zaznamenávána pouze velikost intenzity světla dopadajícího na fotografický materiál, čímž se ztrácí informace o rozložení objektu v prostoru. Holografie přidává do obrazu informaci o fázovém zpoždění odraženého světla od objektu. Fázové zpoždění se v obraze projeví jako rozložení objektu v prostoru. Základním konceptem je tedy vytvoření 2D obrazu obsahujícího prostorovou informaci pozorovaného objektu. [18] [19]

1.1 Optická holografie

1.1.1 Základní koncept hologramu

Jak bylo nastíněno v úvodu kapitoly, na záznamu hologramu se podílí intenzita a fáze odraženého záření. Tuto skutečnost lze popsat časově nezávislou vlnovou rovnicí:

$$E(x, y, z) = E_0 \exp \left[j \left(\vec{k} \cdot \vec{r} + \varphi(x, y, z) \right) \right] \quad (1.1)$$

kde $E(x, y, z)$ je výsledná intenzita elektrického pole, E_0 je amplituda pole, \vec{k} je propagační vektor (někdy označován jako vlnový vektor tedy, který popisuje periodický průběh vlny), \vec{r} je vektor vzdálenosti od počátku souřadnicového systému k bodu na vlnoploše označené (x, y, z) a $\varphi(x, y, z)$ je fáze vlnoplochy.

Jak bylo nastíněno v úvodu kapitoly pro holografii je záznam fáze stěžejní, nicméně přímý záznam fáze je neproveditelný, proto je převedena na intenzitu a její rozložení je zachyceno pomocí snímacího prostředku. Aby převedení bylo možné, holografie přidává referenční zdroj záření. Kombinace referenčního záření a zářením odraženého od objektu s určitým fázovým posunem vyústí v interferenční vzor (hologram).

Záznam a rekonstrukci hologramu je možné rozdělit do tří dílčích kroků, zahrnující superimpozici záření odražené od objektu a referenčního paprsku, záznam interferenčního obrazu na expoziční médium a naposledy samotnou rekonstrukci holografického obrazu. [1]

1.1.2 Superimpozice objektu a referenčního paprsku

Objekt i referenční záření lze popsat komplexní funkcí pole jako:

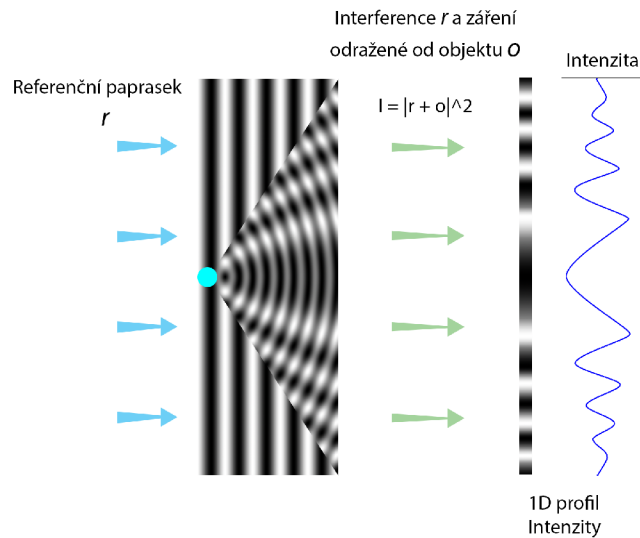
$$\tilde{r} = a_r \exp \left[j \left(\vec{k}_r \cdot \vec{r} + \varphi_r(x, y, z) \right) \right] \quad (1.2)$$

$$\tilde{o} = a_o \exp \left[j \left(\vec{k}_o \cdot \vec{r} + \varphi_o(x, y, z) \right) \right] \quad (1.3)$$

kde \tilde{r} je pole referenčního paprsku, \tilde{o} je pole objektu, a_r a a_o jsou konstantní amplitudy, \vec{r} je poziční vektor, \vec{k}_r a \vec{k}_o jsou propagační vektory a naposled φ_r a φ_o je fáze referenčního paprsku a fáze světla odraženého od objektu. Superimpozici dvou vlnových funkcí je možné matematicky vyjádřit umocněnou koherenční sumou a to:

$$I = |\tilde{r} + \tilde{o}|^2 \quad (1.4)$$

kde intenzita I vzniká interferencí světla odraženého od objektu o spolu s referenčním zdrojem světla r . Superimpozice je graficky znázorněna na obr.1.1. [11]



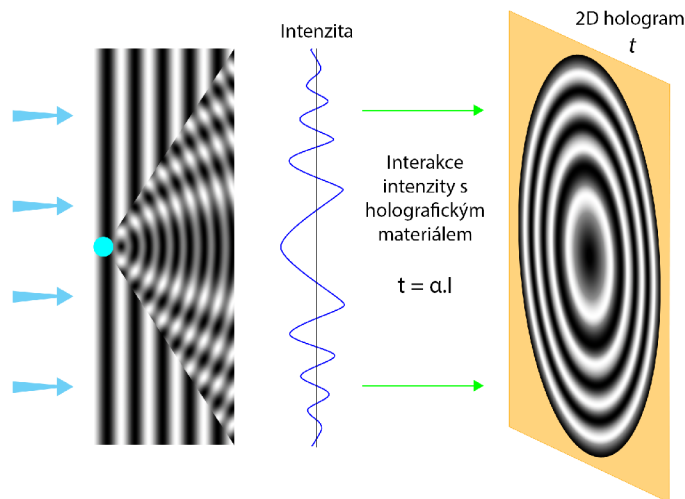
Obr. 1.1: Schéma znázorňující Superimpozice referenční vlny a objektu. Referenčního paprsek r vytváří překryv s vlnou odraženou od objektu O , načež se vytváří interferenční vzor. Z pohledu radiální roviny je vidět závislost intenzit v prostoru.

1.1.3 Záznam hologramu

Pro záznam hologramu optickou metodou se využívá fotocitlivý materiál. Vložním materiálu do míst, kde vzniká interferenční vzor o definované intenzitě, je vytvořena holografická mřížka, jinak také hologram. Hologram lze poté popsat transmittanční funkcí úměrné k intenzitě interferenčního vzoru:

$$t = \alpha I \quad (1.5)$$

kde, t je výsledný hologram a parametr α je dán fotocitlivým materiálem. Proces je poté zachycen na obr. 1.2



Obr. 1.2: Proces vytvoření hologramu na fotocitlivý papír. V závislosti na parametrech papíru α a intenzitě je vytvořen hologram.

1.1.4 Proces rekonstrukce hologramu

Rekonstrukce hologramu probíhá pomocí referenčního paprsku nebo s jeho sdruženou podobou. Jestliže osvítime takovým paprskem hologram, vzniká zrekonstruovaný virtuální obraz snímaného objektu. Tato skutečnost lze matematicky zapsat jako:

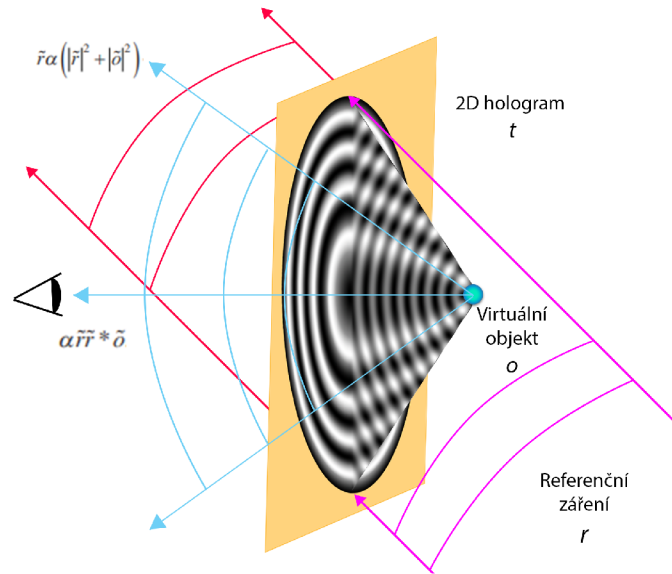
$$\tilde{p}_r = t(x, y, z)\tilde{r} \quad (1.6)$$

Kde, \tilde{p}_r je zrekonstruovaný virtuální obraz, t je fyzický hologram a \tilde{r} je referenční paprsek. Pokud dosadíme komplexní rovnice pro objekt a referenční záření do rovnice 1.4 a získáme podobu hologramu skrz rovnici 1.5 a pro rekonstrukci použijeme rovnici, 1.6 získáme tvar:

$$\tilde{p}_r = \alpha\tilde{r} (|\tilde{r}|^2 + |\tilde{o}|^2) + \alpha\tilde{r}\tilde{r}\tilde{o}^* + \alpha\tilde{r}\tilde{r}^*\tilde{o} \quad (1.7)$$

kde, \tilde{o}^* je virtuální obraz a \tilde{r}^* je konjugovaný referenční paprsek. První část rovnice znázorňuje referenční záření po projití hologramem (hodnota je při porovnání s ostatními zanedbatelná), druhá odpovídá dvojnásobku rozdílu fáze referenčního a objektového záření a představuje podobnost se zářením odraženého od objektu, ale s opačným zakřivením, přičemž konverbuje k vytvoření reálného obrazu před hologramem. Poslední část rovnice je identická k intenzitě záření roptýlené od objektu,

příčemž se podílí na vytváření virtuálního obrazu, a to vyrušením fáze referenčního záření fyzickým hologramem, čímž je ponechána pouze fáze snímaného objektu, která vytváří virtuální obraz za hologramem. Celý proces je graficky znázorněn na obr. 1.3. [1] [3]



Obr. 1.3: Schéma rekonstrukce hologramu. Při osvětlení referenčním paprskem \mathbf{r} nebo jeho obdobou se na 2D hologramu \mathbf{t} láme a vytváří virtuální objekt \mathbf{o} za obrazem z pohledu sledujícího.

1.2 Digitální holografie

U optické holografie byl hologram zaznamenávaný na analogový nosič v podobě fólie z fotocitlivého materiálu. V digitální holografii se využívá k zápisu interferenčního obrazce digitální kamera. Tyto kamery jsou založeny na CCD či CMOS čípech. U digitálního přístupu se využívá procesu vzorkování a kvantizace k převedení analogového záznamu. Tento přístup musí splňovat Nyquistovu podmínku v kombinaci s rozlišením kamery a velikosti objektu. Podmínky do značné míry limitují výběr kamery, jelikož samotné rozlišení ovlivňuje správné zaznamenání hologramu. Aby se předešlo porušení podmínek (nedocházelo k aliasingu apod.) můžeme z rovnice pro difrakční mřížku odvodit největší možnou prostorovou frekvenci f_{max} :

$$f_{max} = \frac{1}{\Lambda_{min}} = \frac{\sin \theta_{max}}{\lambda} \quad (1.8)$$

kde, f_{max} je maximální prostorová frekvence, Λ_{min} je minimální vzdálenost mezi zářezy na mřížce a θ_{max} je maximální úhel interferenčního paprsku a λ je vlnová

délka. S uvážením Nyquistovy podmínky, kdy musíme snímat nejen maximální ale i minimální intenzitu, je výpočet prostorové frekvence uveden jako:

$$f_{max} = \frac{1}{2 \Delta x_H} \quad (1.9)$$

kde, Δx_H je vzdálenost mezi dvěma pixely kamery a definuje podmínky pro velikost objektu snímaného pomocí digitální holografie.

Proces snímání hologramu je obdobný jako v podkap. 1.1, ale k rekonstrukci se poté používají numerické metody jako Fresnelova, či konvoluční metoda.

Fresnelova rekonstrukční metoda vychází z předpokladu, že vzdálenost mezi objektem, kamerou a obrazovou plochou je dostatečná, aby uspokojila podmínku pro aproximaci Fresnelovy difrakce (Světlo po projití stínítkem o určitém tvaru vytváří interferenční vzorec skládající se z světlých a tmavých proužků). Použitím Huygens-Fresnelova principu vycházejícího ze předpokladu, že šířící se vlna postupuje jako obálka získaná z elementárních vln získaných z bodových zdrojů na hraně původní vlny, odvodíme integrální formu pro popis intenzity pole hologramu. Použitím Fresnelovy aproximace na matematický popis hologramu a dalších úpravách této rovnice získáváme tvar obsahující poziční souřadnice ve spojitém tvaru. Nicméně zaznamenáváme hologram diskrétní formou, proto jednoduchou transformací převádíme spojitou formu rovnice na diskrétní zápis, přičemž musíme uvažovat závislost mezi vzdáleností v ose z od obrazové roviny a rozlišení kamery. Tato závislost vychází ze vzorce pro vzorkovací kmitočet 1.9 a lineární závislosti mezi vlnovou délkou vůči hologramu ve vzdálenosti v ose z . Tímto získáme diskrétní tvar rovnice a pomocí Fourierové transformaci je možné rekonstruovat původní pozici objektu.

Limitací Fresnelovy metody je přímá závislost rozlišení kamery a její vzdálenosti v ose z od hologramu. U konvoluční metody vzorkovací kmitočet není omezený vzdáleností, ale pouze rozlišením kamery. Stejně jako Fresnelova aproximace využívá integrální popis vlny vycházející Huygens-Fresnelova principu s tím rozdílem, že na popis rovnice je pohlíženo jako na konvoluci mezi hologramem a jádrovou funkcí obsahující poziční souřadnice. Pro rekonstrukci je poté použita Fourierova transformace, kdy vycházíme ze vztahu mezi transformací a konvolucí:

$$F\{E_i(x_i, y_i)\} = F\{r(x_2, y_2) \cdot t(x_2, y_2)\} \cdot F\{g(x_2, y_2)\} \quad (1.10)$$

Kde F je Fourierova transformace, $E_i(x_i, y_i)$ je obrazová rovina, $r(x_2, y_2)$ je rekonstrukční pole na kameře, $t(x_2, y_2)$ je transmittanční funkce a $g(x_2, y_2)$ je jádrová funkce. Poté můžeme použít inverzní Fourierovu transformaci na druhou část rovnice pro rekonstrukci obrazové roviny. [1] [2]

1.3 Digitální holografická mikroskopie

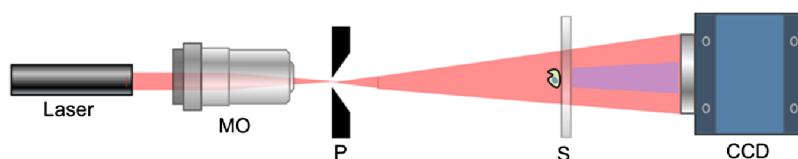
Digitální holografická mikroskopie (DHM) využívá právě digitální holografii a je silným nástrojem používaný nejen v biomedicínských oblastech. Oproti konfokální mikroskopii či mikroskopii využívající světelné pole je DHM schopna zaznamenat objekt i ve třech rozměrech v podobě hologramu. Princip záznamu, tvorby a rekonstrukce hologramu je popsán v podkap. 1.1 a 1.2. Jednou z hlavních výhodou je pozorování buněk v nativním stavu, jelikož DHM nevyžaduje fluorescenční barvení či jiné značení.[15] [17]

Sestava DHM se odvíjí od použitého interferometru, přičemž je můžeme rozdělit na In-line a Off-axis. Dalšími částmi DHM jsou prostorové filtry, děliče paprsků, zrcadla, čočky a detektory. Detektor pro digitalizaci je kamera založena na CCD nebo CMOS čipu.[22] Dalšími možnými modifikacemi snímání je řízení koherence osvětlovacího paprsku.[21]

1.3.1 Soustavy digitální holografická mikroskopie

In-line DHM

Z názvu vypovídá, že In-line DHM je konfigurovaná do jedné linie, jak můžeme vidět na obr. 1.4. Skrz objektiv je vyslán usměrněný paprsek z laseru směrem na clonu se štěrbinou o velikosti vlnové délky. Při průchodu clonou je paprsek rozptýlen. Rozptýlené světlo, které nedopadá na vzorek a není rozptýleno pozorovaným vzorkem, je považováno za referenční. Rozptýlený paprsek od vzorku spolu s referenčním paprskem se poté podílí na vytváření hologramu. Soustavu můžeme ještě zjednodušit o odebrání clony za předpokladu, že zdroj světla bude koherentní. Tato sestava je vhodná pro zkoumání materiálů s vysokým odrazem nebo se silným zpětným rozptylem. [11][22]

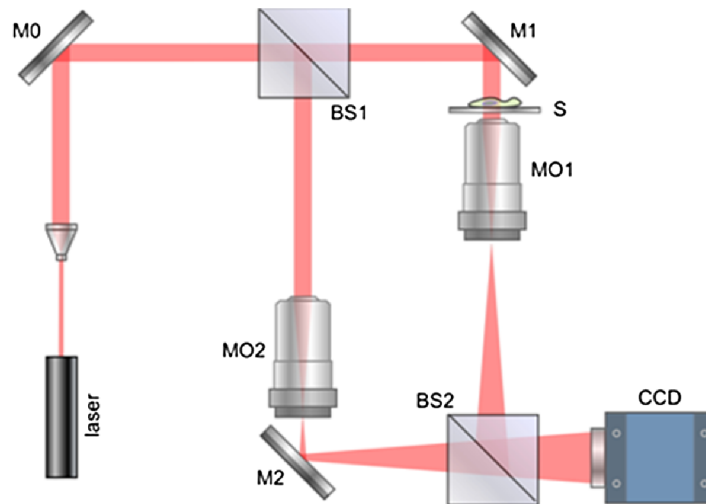


Obr. 1.4: Soustava In-line DHM (MO je označení pro objektiv mikroskopu, P je clona, S je vzorek a CCD je typ použité kamery)[11]

Off-axis DHM

Off-axis soustava využívá Mach-Zehnder interferometr vyobrazen na obr.1.5. V této konfiguraci se paprsek dělí do dvou větví. Referenční větev obsahuje objektiv

(MO2) pro zaostření paprsku. Větev musí být nastavena tak, aby se zakřivení paprsku od vzorku a referenční paprsku shodovalo. Druhá větev obsahuje zkoumaný vzorek. Od vzorku je rozptýleno světlo a zvětšovací objektivem (MO1) je toto světlo usměrněno. Na polopropustném zrcadle (BS2) se paprsky setkávají a vytvářejí hologram. Hologram je zachycen CCD nebo CMOS kamerou. Off-axis metoda je vhodnější především pro transparentní materiály, proto se tato soustava využívá primárně pro biologické vzorky.[11] [22]

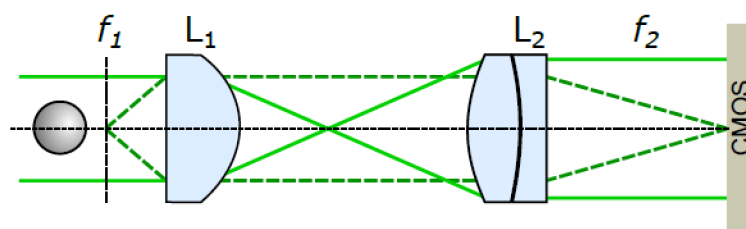


Obr. 1.5: Soustava Off-axis DHM (M 0-2 je označení pro zrcadla, BS 1-2 je označení pro polopropustné zrcadlo, S je vzorek, MO 1-2 je označení pro objektiv mikroskopu, CCD je typ digitální kamery)[11]

1.3.2 Použitá soustava v experimentu

Pro snímání experimentálních snímků pro tuto práci byla použita in-line konfigurace. Jako zdroj světla pro osvětlení vzorku byl použit zelený laser s průměrem 200 μm a s vlnovou délkou 532 nm. Objektiv mikroskopu byl vytvořen prostřednictvím asférické čočky (ohnisková vzdálenost $f_1 = 11.0$ mm a numerická apertura $NA = 0.23$), na Obr.1.6 představuje L_1 a achromát ($f_2 = 200.0$ mm) označený na Obr.1.6 jako L_2 . Výsledné hologramy jsou snímány pomocí CMOS kamery.

Pomocí této soustavy byly nasnímány vzorky levitujících nanočástic oxidu křemičitého (SiO_2). Nanočástice levitujícího křemíku byly zachyceny pomocí radiofrekvenční pasti. [23]

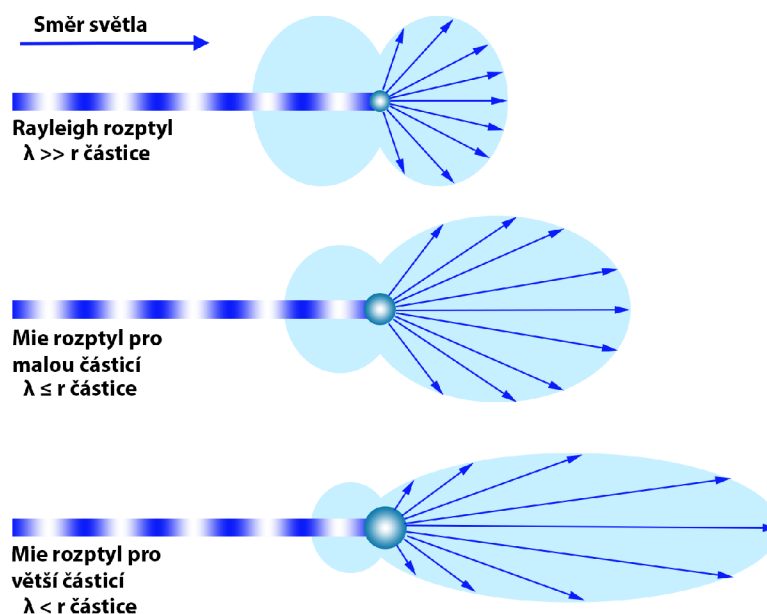


Obr. 1.6: In-line holografický mikroskop použitý pro snímání experimentálních dat. Šedá kulička znázorňuje sledovanou částici, f_1 je ohnisková vzdálenost pro sférickou čočku L_1 , f_2 je ohnisková vzdálenost pro achromatickou čočku L_2 , blok CMOS je použitá kamera. Šíření zeleného laseru je z leva doprava. [23]

1.4 Paralela mezi Mie teorií a hologramem částice

1.4.1 Mie teorie

Mie teorie pojednává o rozptylu elektromagnetického záření na izotropním, homogenní sférickém tělesu. [6] Jinými slovy popisuje rozptyl světla na podobně velkém kulovitém objektu, jako je vlnová délka světla. Mie rozptyl je druh elastického rozptylu, který je dělen do třech skupin podle velikosti poměru vlnové délky záření a poloměru tělesu. Jestli je objekt, přes které se záření, láme menší než vlnová délka, mluvíme o Rayleigh rozptylu. Je-li poměr velikostí částice a vlnové délky větší, než 1, podléhá rozptyl světla na tělesu zákonům geometrické optiky. Pokud je tento poměr kolem jedné, tedy poloměr částice je přibližně roven vlnové délce, považujeme tento rozptyl za Mie rozptyl. Vliv velikosti objektu a záření je možné vidět na Obr. 1.7 [7]

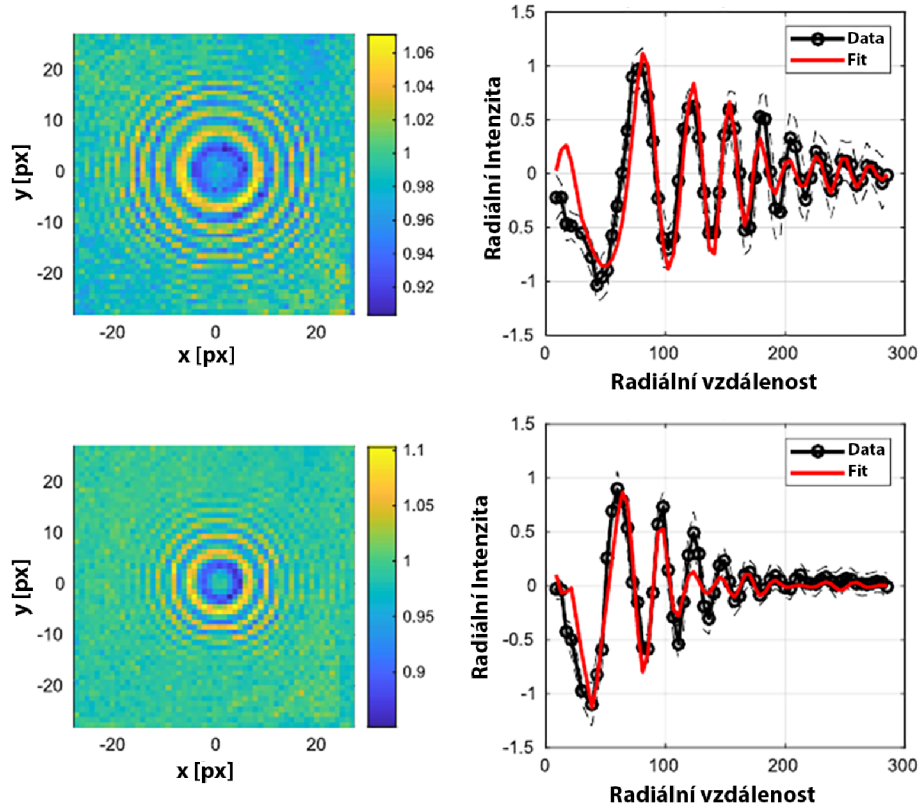


Obr. 1.7: Vliv velikosti částice na Mie rozptyl.

Mie rozptyl můžeme pozorovat u koloidních roztoků, aerosolů či tkání. Nejčastějším příkladem pro znázornění Mie rozptylu jsou koloidní roztoky, jelikož obsahují částice v řádu nano a mikro. U organismů můžeme najít koloidní roztoky v podobě tělních tekutin (např. krev, lymfa). [10] U aerosolů se s Mie rozptylem setkáváme při environmentálních měření (koncentrace malých částí ve vzduchu a oblacích) a útlumu paprsku u optických komunikačních systémů (např. Lidar) [6][9]

1.4.2 Hologram částice a Mie teorie

Při použití DHM právě na částice o velikosti blízké vlnové délce můžeme zaznamenané hologramy interpretovat pomocí výše popsaného Mie rozptylu. Kdy intenzita světla rozptýlená na částicích podléhající Mie teorie je skoro totožná s vytvořenými hologramy se stejnými částicemi. Této skutečnosti je poté možné využít při tvorbě nových umělých dat pro trénování metod založených na strojovém učení jako v pracích [25], [14] a [16]. Dalším použitím Mie teorie je samotný odhad parametrů částic, jak nastínili v práci [8] a [15], kde porovnávají radiální profily, či samotné hologramy pro odhad parametrů částice (poloha, poloměr a index lomu).



Obr. 1.8: Porovnání reálné a fitované radiální symetrie hologramu . Nalevo je experimentální hologram zaznamenaný experimentální soustavou 1.3.2 a vlevo je porovnání radiálních symetrií skrz fitování. Hodnoty pro horní hologram jsou průměr 659 nm, index lomu 1.455 a pozice z $-69.6 \mu\text{m}$. Pro spodní jsou hodnoty stejné až na pozici z $(-43.6 \mu\text{m})$ [23].

Jak bylo popsáno v podkap. 1.1.2, výslednou interferenci referenční a objektové vlny je možné popsat skrze rovnici 1.4. Objektovou vlnu lze aproximovat funkcí $f(k(x, y, z))$ za předpokladu, že objekt je menší v porovnání s vlnovou délkou osvětlení:

$$\tilde{o} = \tilde{r}f(k(x, y, z)) \quad (1.11)$$

funkce $f(k(x, y, z))$ zde představuje vlnu vzniklou z rozptylu referenční vlny na objektu. Funkci je možné přepsat do expanzivní řady vyjadřující obecnou Lorentz-Mie teorii. Řada se skládá z harmonických sférických vektorů a expanzivních koeficientů u_{mn} a v_{mn} . Rozložením expanzivních koeficientů získáme koeficienty β_n a α_n , které jsou podobné Lorentz-Mie koeficientům a budou pro vytvoření umělého hologramu zásadní, jelikož obsahují informaci o průměru a_p , indexu lomu částice n_p a média n_m , ve kterém se částice vyskytuje a nabývají tvaru:

$$u_{mn} = i^{n+3m-|m|} \frac{2n+1}{n(n+1)} \frac{(n-m)!}{(n-|m|)!} g_{n,TE}^m \beta_n(ka_p, \frac{n_p}{n_m}) \quad (1.12)$$

$$v_{mn} = i^{n+3m-|m|} \frac{2n+1}{n(n+1)} \frac{(n-m)!}{(n-|m|)!} g_{n,TM}^m \alpha_n(ka_p, \frac{n_p}{n_m}) \quad (1.13)$$

kde, n a m jsou koeficienty tvaru z expanzivní řady, $g_{n,TM}^m$ a $g_{n,TE}^m$ jsou koeficienty pro magnetické TM a elektrické TE příčné vlnění. Dalšími úpravami lze získat normalizovaný tvar rovnice pro aproximaci radiálního profilu reálného hologramu. Pro aproximaci můžeme využít fitování koeficientů v rovnicích 1.12 a 1.13 a měření odchylky. Tímto způsobem lze získat informace o pozici, indexu lomu či poloměru částice. Podobným způsobem je využíváno Lorentz-Mie teorie k tvorbě simulací hologramů. [4]

2 Zpracování obrazu pomocí strojového učení

Strojové učení v oblasti zpracování obrazu je nedílnou součástí mnoha odvětví počínaje zábavním průmyslem přes automobilovou dopravu po vědecké a medicínské aplikace. V následujících podkapitolách jsou popsány metody strojového učení a jejich implementace na hologramy částic.

Metody hlubokého učení využívané pro získání kvantitativních informací z digitální holografické mikroskopie podle [5] se stávají z *Dense neural network* (zkr. DNN), *Convolution neural network* (zkr. CNN) a *U-Net*. Metody hlubokého učení nejsou jediné přístupy pro extrakci příznaků, přičemž je možné využít i klasických klasifikátorů jako *Support vector machine* (zkr. SVM) apod.

2.1 Umělé neuronové sítě a hologram částic

2.1.1 Úvod do neuronových sítí

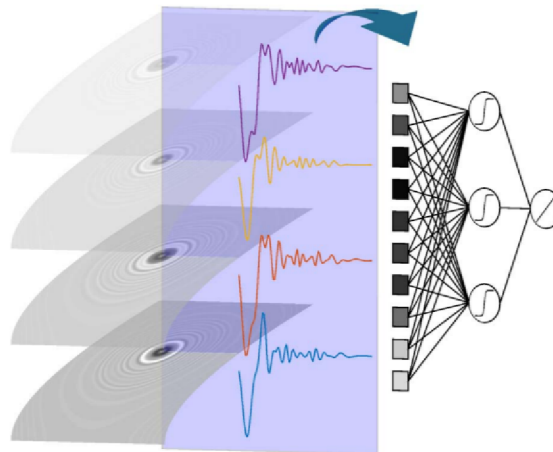
Předností umělých neuronových sítí (dále jen *ANN*) v kontrastu s klasickými algoritmy je schopnost adaptace a učení. Architektura je odvozena z neuronových propojení v mozku. Základní jednotkou je umělý neuron a odehrává se zde vyhodnocení vstupních hodnot. Ty jsou modifikovány skrze trénovatelné váhy a transformovány pomocí nelineární aktivační funkce. Trénování vah probíhá nejčastěji prostřednictvím algoritmu zpětného šíření chyby, přičemž můžeme algoritmus modifikovat o adaptaci rychlosti učení, moment učení nebo jejich kombinace. Vypočet prvotní predikce je provedena s aktuálními hodnotami vah. Se zvolenou kritériální funkcí je vypočítána chyba mezi predikcí a pravdivou hodnotou. Podle vypočítané chyby se poté iterativně upravují váhy. Tento proces se opakuje v iteracích a je ukončen po dosažení nulové chyby nebo chybového limitu.

Aby byly použitelné pro složitější klasifikaci, je nutné použití většího množství jednotlivých jednotek, které jsou uspořádané do vrstev. Jednou z metod hlubokého učení jsou hustě propojené umělé neuronové sítě (dále jen *DNN*). Pojem hustě propojené odkazuje na přidání umělých neuronů do jedné vrstvy a zařazení většího množství vzájemně propojených vrstev za sebe. Díky navýšení hustoty je *DNN* schopná aproximovat složitější funkce, což zajišťuje řešení i sofistikovanějších problémů jako zpracování obrazových dat. Nevýhodou *DNN* je neměnný počet vstupů a výstupu, což má za následek rigiditu modelu, kdy musíme na vstupy posílat pouze předpracovaná data. Výstupem sítě jsou hledané parametry či klasifikace.[12]

2.1.2 Umělé neuronové sítě v holografické mikroskopii

Autoři v práci [13] navrhli dopřednou neuronovou síť pro určení vnitřního a vnějšího poloměru částice (dutá částice) z radiální symetrie (1D reprezentace intenzity hologramu) vyextrahované z hologramů částice (viz. Obr 2.1). Radiální intenzity zdiskretizovali na vektor a ty poté použili jako vstup. S touto úvahou zároveň omezili snímání hologramů na pouhou 1D intenzitu, což by mělo za následek zrychlení snímacího procesu v reálném čase.

Pro každý parametr vytvořili samostatnou DNN. Pro trénování navržené topologie využili syntetická data vygenerované prostřednictvím Mie teorie. Aby předcházeli přeučení, využili L1 regularizace. Zároveň uvádějí motivaci výběru L1 oproti L2, a to z důvodu tendence L1 regularizace směřovat váhy sítě k nule, čímž se snaží vektorová reprezentace o zvýšení efektivity v hardwarové aplikaci. Trénování s 10 různými inicializačními váhami se zabraňuje zachycení v lokálním minimu. Narozdíl od jiných prací zanedbávají vycentrování částice (změna lokalizace částice může ovlivnit výslednou radiální symetrii) a lokalizují první moment difrakce, který přesouvají do středu. Zároveň metoda nebere v úvahu překrývání spekter jednotlivých hologramů.



Obr. 2.1: Určení poloměru částic pomocí ANN. [13]. Z hologramu je vypočítána radiální symetrie. Jako vektor je umístěn na vstupní vrstvu ANN, kde je poté vyhodnocen poloměr částice

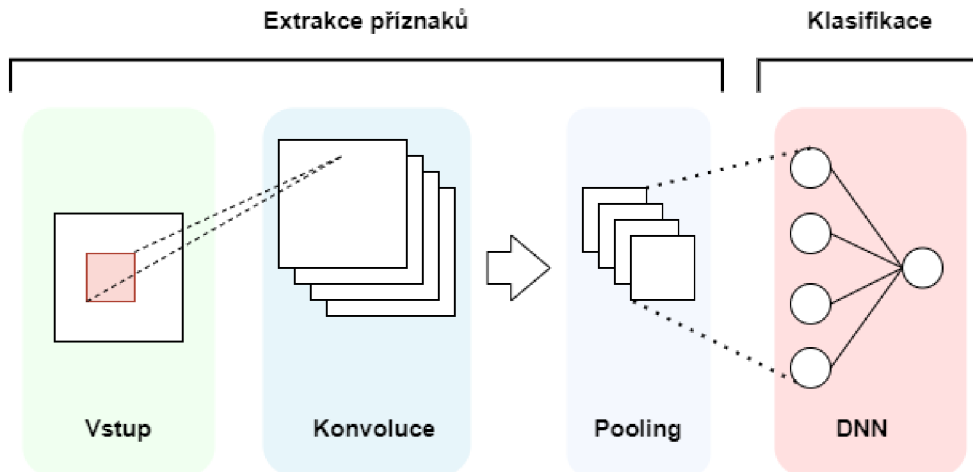
2.2 Konvoluční neuronové sítě

2.2.1 Základy pro konvoluční neurální sítě

Jak bylo nastíněno v nevýhodách DNN, je nutné předzpracovat obrazová data, jelikož každý pixel představuje jeden vstup. V porovnání s konvolučními neuronovými sítěmi (dále jen *CNN*) je tato nevýhoda odstraněna zařazením konvoluční vrstvy před klasifikací. Vrstva obsahuje 2D filtry o rozměrech $k \times k$ a skrze 2D konvoluci vytváří skalární součin filtru a obrazu. Součin se poté interpretuje do nové matice nazývané jako příznaková mapa. Počet vytvořených map je odvozen od počtu použitých filtrů. Vlastnosti filtrů jsou voleny tak, aby extrahovaly potřebné informace (hrany či specifické tvary).

Za konvoluční vrstvou je zařazena poolingová vrstva. Pooling vychází z předpokladu, že hodnoty v obraze pod maskou o rozměrech $l \times l$ můžeme označit za redundantní a pomocí matematické operace (průměr, max) tuto oblast podvzorkuje. Podvzorkování má za následek zmenšení velikosti obrazu a příznakových map, čímž je snížena výpočetní náročnost modelu.

Poslední částí architektury *CNN* je klasifikační vrstva. Vrstva je realizována jednoduchou *DNN*, kdy na vstup jsou přivedeny zvektorizované příznakové mapy a na výstupu získáváme hledanou klasifikaci. [32]



Obr. 2.2: Architektura konvoluční neuronové sítě

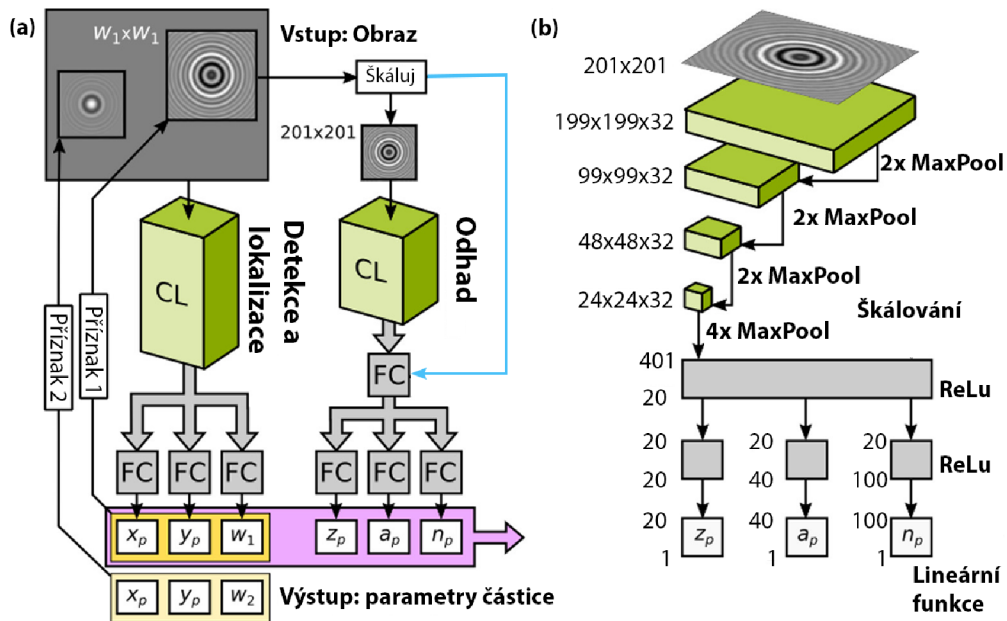
2.2.2 Využití konvolučních sítí při určování parametrů částice

Mezi nejčastější přístupy k odhadu parametrů částic patří metoda CNN. V práci [27] navrhli autoři síť pro zpracování snímků částic znečišťujících ovzduší v reálném čase. Neuronová síť má za úkol detekovat počet částic polystyrenu a oxidu křemíku

v závislosti na změně difrakčního vzoru hologramu. Navrhnutá topologie obsahovala 2 konvoluční vrstvy s max-poolingem a DNN jako klasifikační vrstvou. Pro zefektivnění trénování byl použit Dropout a optimalizační algoritmus ADAM. K výpočtu chyby byla použita kritériální funkce *softmax cross entropy*. Počet výstupní neuronů je 13, kdy prvních 9 koresponduje s počtem přítomných polystyrenových částic (0-8) a zbylých 4 pro oxid křemičitý (0-3). Pro vytvoření trénovacího souboru dat použili autoři experimentální sestavu (in-line DHM), s jejíž pomocí nasníмали částice s předpokládanými parametry, což se výrazně odlišuje od postupu u většiny prací, kdy je využita Mie teorie pro generování syntetických dat.

Autoři práce [16] si dali za úkol porovnat heuristické algoritmy, kaskádový klasifikátor a *CNN* pro určení 3D polohy, průměru a indexu lomu. Mezi heuristické algoritmy můžeme zařadit například kruhovou Hough transformaci s následnou detekcí vrcholů. Kaskádové klasifikátory se řadí mezi tzv. slabé klasifikátory a jsou dále rozepsané v kap.2.4. Pro určení 2D polohy byly heuristické algoritmy nejlepší, nicméně pro další analýzu částic nepoužitelné. Podobné výsledky zajišťoval model *CNN*, přičemž měla nejlepší výsledky pro určení ostatních parametrů. V porovnání s ostatními metodami v odhadu parametrů nejhůře dopadla metoda kaskádových klasifikátorů, ale co se týče výpočetní náročnosti a vybavenosti, překonává obě metody.

Složitější architekturu využívající *CNN* představili autoři v článku [20] a nazvali ji CATCH (*Characterizing and Tracking Colloids Holographically*). Síť je možné rozdělit do jednotlivých modulů s daným úkolem. První modul obstarává vyhledávání, lokalizace a určení velikosti vzoru hologramu jedné částice v záznamu a je založen na modelu sítě YOLOv3. YOLOv3 je aplikační rámec používaný pro objektovou detekci v reálném čase. Do toho modulu vstupuje šedotónový obraz částice a výstupem je 2D lokalizace středu částice a její rozsah interferenčního vzoru (v podobě oblasti zájmu). Vyříznutý obraz je předán do druhého modulu. Zde probíhá odhad parametrů jako pozice směrem k ohniskové rovině, indexu lomu a poloměru částice. Obraz částice je škálován a předán navrhnutému modelu *CNN*. Třetím blokem v kaskádě je předání příznakových pixelů a parametrů pro upřesnění parametrů skrz adaptaci na rovnici intenzity Mie teorie skrz metodu nejmenších čtverců.

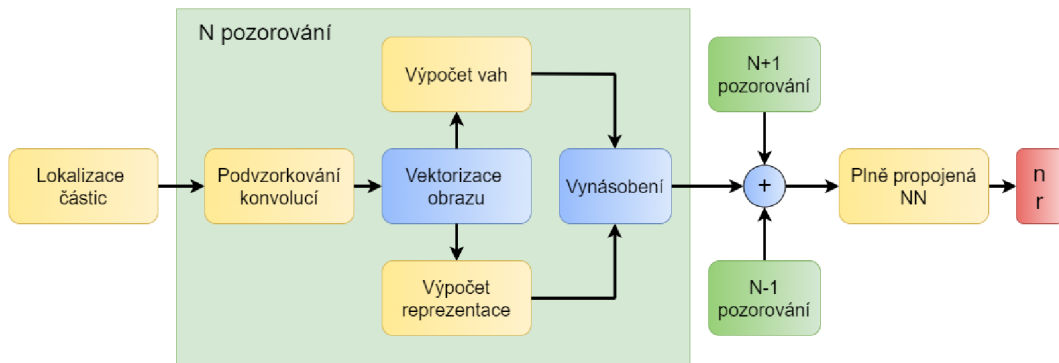


Obr. 2.3: Schéma architektury CATCH [20]

V práci [25] byly za použití off-axis DHM nasnímány hologramy částic. Pozorovaným vzorkem je průtok kapaliny v mikrofluidním čipu. Médium protékající kanálkem v čipu obsahuje variaci částic s odlišnými hodnotami parametrů. Sledovanými parametry jsou poloměr, index lomu a 3D pozice nanočástic.

Autoři uvádí, že je možné využít i jiných analytických metod pro extrakci parametrů z hologramu částic a byly by i použitelné, kdyby obrazy nedisponovaly nízkým SNR. Problém s nízkým SNR vychází z předpokladu, že informace o pozici částice jsou uloženy ve vyšších frekvencích spektra obrazu. Nízké SNR lze odstranit prostřednictvím průměrování vícero obrazů, ale pro přímé průměrování je nutné znát přesnou polohu částice ve 3D prostoru. Dokonalá lokalizace je v tomto případě nemožná, proto přistupují s návrhem architektury *CNN*.

Dále autoři řeší problém s nesourodými obrazy z důvodu netolerovatelného šumu či interference holografických vzorů jednotlivých částic a zavádí ohodnocování obrazů v závislosti na jeho poškození. Velikost váhy přiřazené k obrazu poté ovlivňuje příspěvek v průměrování obrazů. Autoři tento typ modelu nazvali *weighted average convolution neural network* (WAC-NET).



Obr. 2.4: Schéma WAC-NET.

Pro trénování WAC-NETu využili syntetické hologramy vytvořené na základě Mie teorie. Vzory částic jsou promítány do obrázků 64x64. Poloměr a index lomu jsou rovnoměrně rozloženy do intervalu. Zároveň jsou data argumentována o decentralizaci a korelována s šumem, aby odpovídala experimentálním vzorkům. Umělá data byla navíc generována průběžně, tudíž nedocházelo k přeučení z důvodu malého souboru dat.

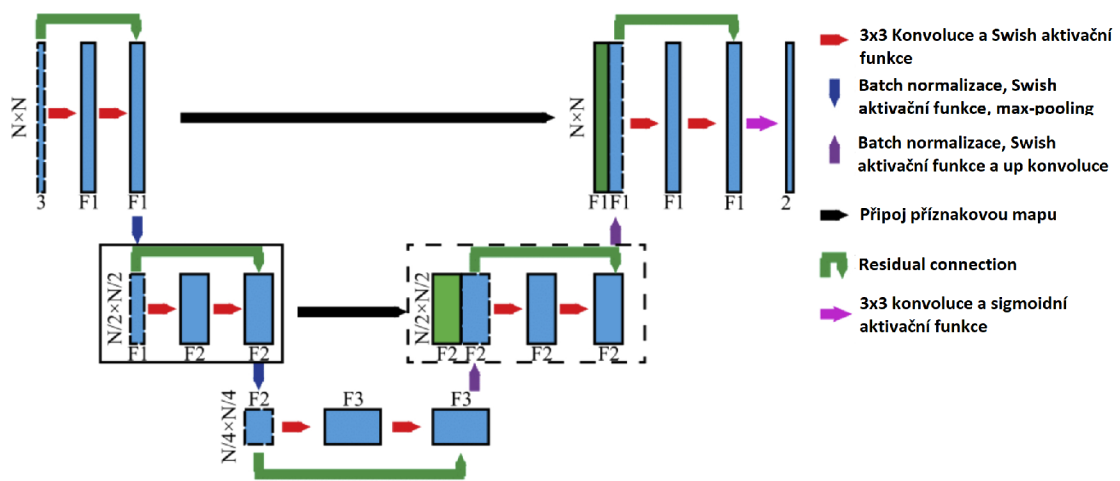
2.3 U-Net

Prvotním využitím architektury U-Net byla segmentace biomedicínských dat, nicméně tato metoda se ukázala jako efektivní nejen pro segmentaci, ale i pro lokalizaci objektů na obraze či rekonstrukce 3D polohy. Topologie U-Net přímo vyhází z architektury CNN s tím rozdílem, že výstupem sítě nejsou hledané parametry, ale zrekonstruovaný obraz. Topologii je možné rozdělit do dvou části na kontraktilní cestu (angl. *contracting path*) a expanzivní cestu.

Kontraktilní cesta topologicky vychází z CNN, avšak v tomto případě je klasifikační vrstva vynechána. Expanzivní cestu je možné přirovnat k reverzní CNN, neboť místo podvzorkování (pooling) jsou aktivační mapy nadvzorkovány, čímž se počet aktivačních map snižuje. K nadvzorkovaným mapám jsou skrz *skip connection* připojeny oříznuté příznakové mapy z kontraktilní dráhy, jež byly vytvořeny na stejné úrovni (oříznutí je nutné kvůli ztrátě hraničních pixelů po každé konvoluci v kontraktilní dráze) a transformovány dekonvolucí. Tím se připojené příznakové mapy podílí na rekonstrukci výsledného obrazu. Výsledkem sítě je obraz se segmentovanými objekty v závislosti na trénovacích datech. [24]

2.3.1 Použití U-Net pro 3D rekonstrukci polohy částice

Jak už bylo výše nastíněno U-net není primárně zaměřený na segmentaci. V práci [28] je modifikace U-Net využita pro rekonstrukci 3D pozice koloidních částic z hologramu. Od původní architektury se liší přidáním *residual connection* pro obě dráhy (viz. 2.5). Propojení zefektivní využití trénovacího času a snížení pravděpodobnosti zachycení trénování na lokálním minimu. Dalším odchýlení od původního návrhu je i změna aktivační funkce z ReLU na Swish (Sigmoid-weighted Linear Unit). Změna je podložena převahou nulových hodnot v příznakové mapě díky malým velikostem centroidů částic ve snímaném obraz. ReLU je v oblasti 0 monotónní a její derivace odpovídá také 0, oproti Swish, která je blízko 0 nemonotónní. Tímto přístupem se zvýší počet efektivních parametrů pro trénování.



Obr. 2.5: Topologie sítě U-Net představena v [28]

Aby zvýšili přesnost určování pozice, připojili k samotnému hologramu pro trénování dva typy obrazů vycházející z hologramu. Pomocí metody úhlového spektra je hologram částice transformován do trojrozměrného komplexního optického pole. Z tohoto pole je generována hloubková mapa. Mapa je tvořena pozicí pixelu s nejvyšší intenzitou. Mapa maximální fáze je generována jako maximální úhel z pole. Výstupem sítě je poté šedotónový obraz, kde hodnota stupně odpovídá relativní hloubce obrazu. Druhým výstupním obrazem je binární maska popisující střed částice v osách x a y . Rozdílné obrazy na výstupu způsobují nesprávné určování chyby, proto pro každý kanál zavádějí rozdílné kritériální funkce. Pro kanál odhadující hloubkovou mapu vybrali Huber kritériální funkce. Huber funkce je modifikována L1 a L2 kritériální funkce, kdy v závislosti na vypočítané chybě a volitelném parametru δ je volena L1 nebo L2. Parametr δ ovlivňuje přesnost určení částice v ose z . Naopak

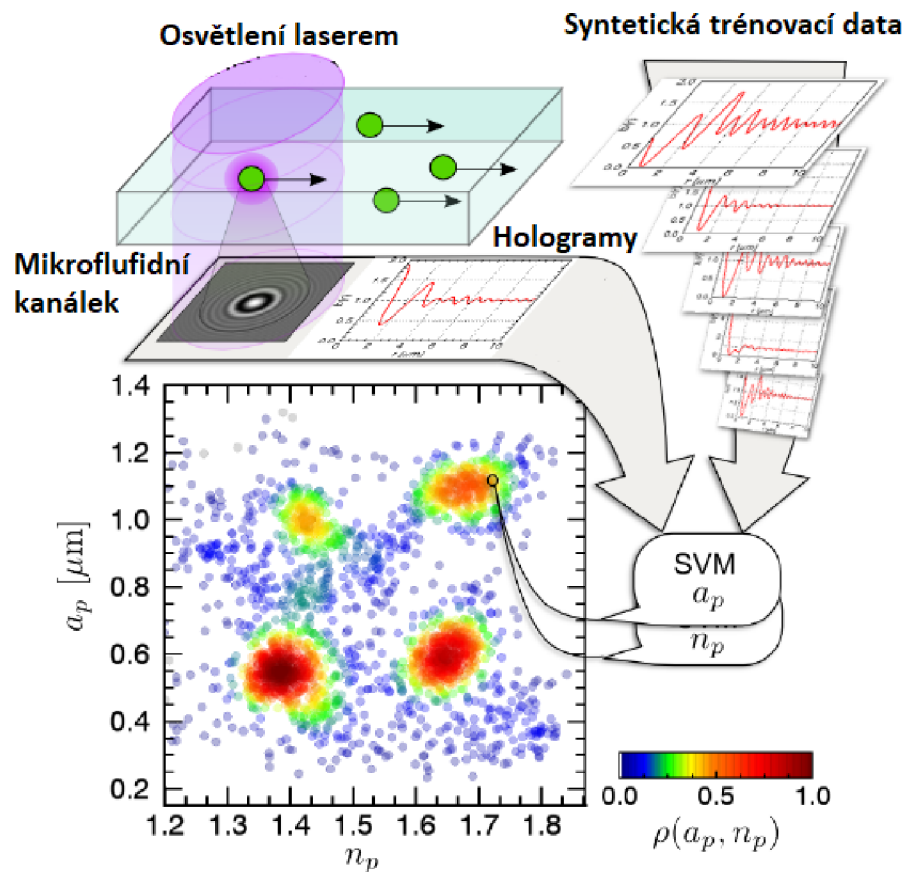
příliš malý parametr delta může vest až k nestabilnímu učicímu procesu, takže kritériální funkce je skoro nahrazena L1 funkcí. Pro Druhý kanál určující střed částice byla vybrána L1 kritériální funkce s regulací v podobě celkové variaci. Autoři uvádí, že tento druh funkce byl již minulosti použit pro rekonstrukci hologramů, protože poskytuje odolnost na odlehlé hodnoty. Funkce se chová jako filtr typu dolní propusti a vrací vyhlazený obraz a je tedy zamezeno i falešně pozitivním detekcím.

2.4 Klasifikátory

2.4.1 Metoda podpůrných vektorů v digitální holografické mikroskopii

Myšlenkou metody podpůrných vektorů (dále jen *SVM*) a zároveň jejím principem je rozdělení dat do tříd v N -rozměrného prostoru, kde N udává počet příznaků. Separátor je v podobě nadroviny, přičemž metoda hledá nejlepší umístění nadroviny. K určení polohy nadroviny v prostoru jsou potřebné podpůrné vektory. Vektory je možné vypočítat z hraničních hodnot jednotlivých tříd. Ze třídy jsou vybrány okrajové hodnoty, které představují kotevní body pro vektory a jsou na sebe rovnoběžné. Tím je vytvořena oblast, která se řídí podmínkou, že neobsahuje žádné prvky a zároveň odděluje dvě třídy. Do oblasti poté vkládáme separační nadrovinu. Optimalizace při hledání největší vzdálenosti mezi dvěma třídami, probíhá pomocí kvadratického programování. Metoda naráží výpočetně na vyšší počet příznaků, jelikož je nutné data namapovat do vyšších dimenzí, čímž se zvyšuje výpočetní náročnost. *SVM*, jak je výše popsána je proveditelná pouze za předpokladu, že jsou třídy lineárně separabilní. Využití na nelineární separaci je možné po úpravě pomocí *Kernel trick*. [30]

Autoři práce [14] navrhli metodiku se základem pro určování parametrů za předpokladu podobnosti mezi vypočítaným vektorem podle Mie teorie a radiálním profilem hologramu částice. Jelikož je *SVM* metoda založena na trénování s učitelem je potřeba zvolit trénovací data. Pro trénování jsou zvoleny předvypočítané vektory podle Mie teorie. Známé parametry pro výpočet byly průměr, index lomu a pozice směrem k ohniskové rovině. Ze snímku pořízeného in-line DHM je poté vyextrahován 1D profil a pomocí naučeného modelu jsou nalazeny hledané parametry. Podobnost mezi trénovací a získanou radiálním profilem určili pomocí báze funkce založené na jejich diferencí a výpočtu plochy pod křivkou. Celý proces je poté zachycen na Obr.2.6



Obr. 2.6: Určení parametrů částice pomocí SVM. [14]. Graf znázorňuje výsledky aplikované metodiky 2500 částic 4 druhů částí. pro SVM znázorňuje závislost a_p (představuje průměr částice) na n_p (představuje index lomu) a $\rho(n_p, a_p)$ je hustotu částic.

3 Návrh metodiky pro určení parametrů částic pomocí strojového učení

3.1 Deeptrack 2.1

DeepTrack 2.1 (dále jen DT) je aplikační rámec (angl. *Framework*) založený na programovacím jazyku Python 3.6. [5] DT poskytuje uživateli nástroj pro vytváření simulací optických systémů, generování obrazů skrz tyto systémy a uživatelský přívětivý přístup pro návrh a učení modelů hlubokého učení vycházející z knihovny TensorFlow.

3.1.1 Simulace optického systému

Jedním ze základních funkcí generování dat pro trénování modelů je možnost simulovat několik typů mikroskopu, a to mikroskop využívající světelné pole, fluorescenční mikroskop a mikroskop s koherentním osvětlením. Po výběru typu mikroskopu je nutné definovat jeho numerickou aperturu (NA), typ osvětlení ve vlnové délce, zvětšení, rozlišení a velikost výstupního obrazu, aby byly syntetická data precizní. Zároveň obsahuje možnost pro přesné definování optické soustavy pro každý mikroskop.

K vytvoření trénovacího a validačního souboru dat pro naši aplikaci je důležitý výběr správného typu mikroskopu a nastavení jeho parametrů. Z dokumentace pro *DT* doporučují pro simulaci holografického mikroskopu použití světelného mikroskopu skrz metodu *BrightField()* a její parametry byly nastaveny následovně:

Tab. 3.1: Parametry navrženého optického systému.

Parametr	Hodnota
Numerická apertura [NA]	0.2
Rozlišení	20^{-6}
Zvětšení optické soustavy	20
Vlnová délka zdroje světla	532^{-9}
Index lomu prostředí [n]	1
Velikost výstupního obrazu	64x64

Hodnoty parametrů vycházejí z experimentální soustavy popsané v podkap. 1.3.2. Velikost obrazu je vybrána na základě doporučení z dokumentace pro *DT*. Zároveň je přidán iluminační gradient. Tato funkce přidá do obrazu nerovnoměrné rozložení jasu, čímž se obraz přibližuje v podobnosti s experimentálním obrazem.

3.1.2 Obraz rozptylu světla na objektu

Po definování optické soustavy je nutné definovat pozorovaný objekt. *DT* simuluje rozptyl světla pro jednoduché geometrické tvary, jako elipsy, elipsoidy, bodové a kulové částice. Současně simuluje specifické rozptyly světla jako Mie rozptyl na vyplněné nebo stratifikované sférické částici. Pro správné zobrazení se definují parametry objektů jako jejich poloměr, index lomu, 3D pozici v obraze, rotaci kolem dané osy, polarizační a sběrný úhel.

Pokud budeme vycházet z podkap. 1.4.2, tedy podobnosti mezi hologramem a Mie rozptylem. Tuto skutečnost lze využít pro vytvoření hologramů částice pomocí metody *MieSphere()*. Parametry metody byly nastaveny následovně:

Tab. 3.2: Parametry navržené částice.

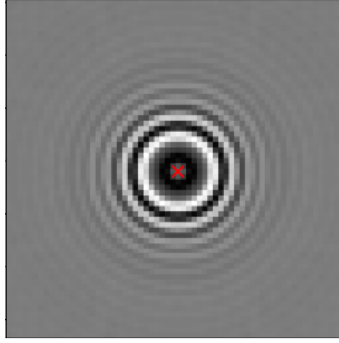
Parametr	Hodnota
Počet částic v obraze	[1, 6]
Pozice částice ve osách x a y [px]	[0, 63]
Pozice v ose z	[0,250]
Index lomu částice [n]	[1.2,1.8]
Poloměr částice [m]	[264·10 ⁻⁹ , 336·10 ⁻⁹]

Hodnoty vycházejí z částic oxidu křemičitého použitého v experimentálním souboru dat. Výrobce nanočástic oxidu křemičitého uvádí 12 % odchylku v poloměru částice, od původní hodnoty poloměru 300⁻⁹, což odpovídá změně poloměru o $\pm 36^{-9}$. Interval indexu lomu byl vybrán na základě článku [14]], věnující se určení indexu lomu, přičemž index lomu oxidu křemičitého je uprostřed tohoto intervalu o hodnotě 1.47.

3.1.3 Generování a úprava dat

Vygenerování dat probíhá skrz zavolání definované optické soustavy se zvolenou částicí. Tím dostáváme obraz jako bychom se dívali na částici skrze mikroskop (viz. Obr. 3.1). Ke každému vygenerovanému obrazu je přidán popis jeho parametrů (pozice, index lomu, velikost, poloměr atd.).

Vytvořený rozptylový obraz je ideální z pohledu jeho dalšího zpracování (viz. Obr. 3.1). Není tedy zatížený šumem, částice je centrovaná a rovnoměrně osvětlená, tedy neodpovídá reálnému vzoru nasnímané částice. Aby uměle vytvořené obrazy byly použitelné pro trénování modelu, umožňuje *DT* přidání do obrazu Gaussovský šum a iluminačního gradientu. Poté byly obrazy normovány za využití maxima a minima z obrazu do škály od 0 do 1.



Obr. 3.1: Příklad zobrazení obrazku Mie rozptylu pomocí *Brightfield microscopy*. Červený křížek označuje střed. Parametry: $n = 1.443$ a $r = 540$ nm

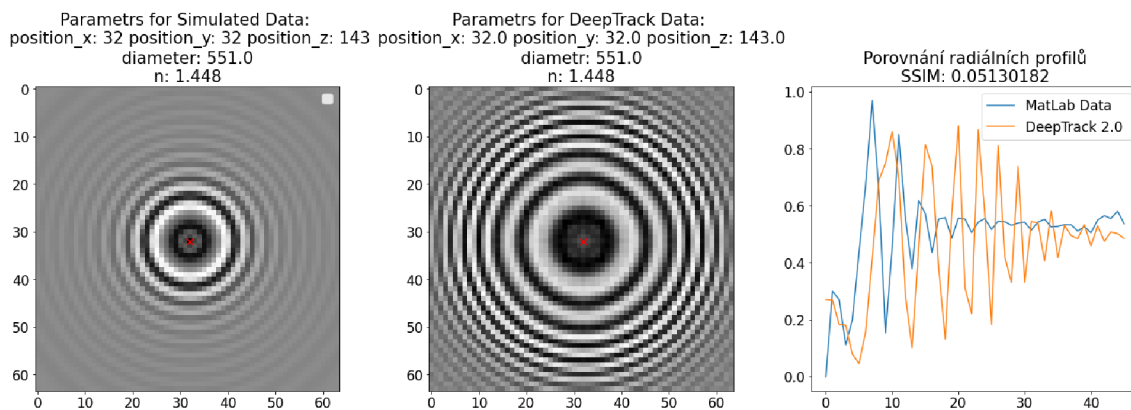
Než budou vygenerované data použity pro trénování sítě je nutné simulované hologramy ověřit v podobnosti s experimentální sadou. Pro tento účel jsou využity hologramy vygenerované prostřednictvím Matlab a Python skriptu poskytnuté z ÚPT AV ČR. Tyto hologramy byly ověřeny v článku [23] s experimentálními snímky částic. Zároveň hologramy vytvořené skrze metodu použité v práci [23] nejsou vhodné pro učení sítě, jelikož jsou málo variabilní, tedy obsahují pouze jednu částici, jsou centrovány do středu a jejich počet je omezený. Současně je výpočet hologramu strojově náročnější, v porovnání s DT. Problém s variabilitou je možné předcházet složitější úpravou vytvořené funkce pro tvorbu hologramů z MatLabu a omezený počet dat lze vyřešit metody augmentací.

Další možným řešením je pomocí DT vytvořit trénovací data a jejich porovnání s již ověřenými hologramy. Pro ověření byl navrhnut program, do kterého byl zakomponován poskytnutá funkce v Python, generátor dat z DT a jako podobnostní kritéria byly zvoleny koeficient SSIM (angl. název *structural similarity index measure*). SSIM porovnává dva obrazy, přičemž uvažuje strukturu, kontrast a jas. V článku [31] představují výpočet indexu jako:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (3.1)$$

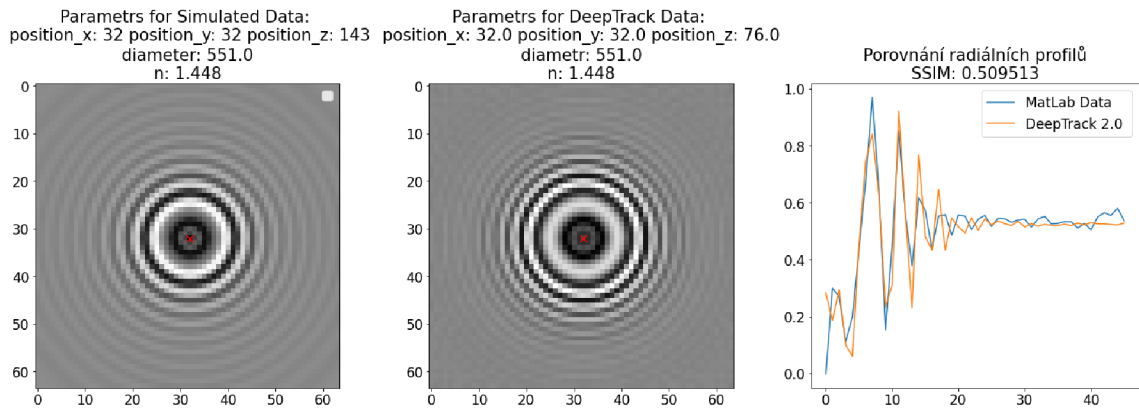
kde μ je vážený průměr, σ je variace, C_1 a C_2 je konstanta předcházející nestabilitě, pokud jsou části rovnice $(\sigma_x^2 + \sigma_y^2)$ a $(\mu_x^2 + \mu_y^2)$ blízké nule. Pokud index SSIM vychází 1 jsou obrazy stejné a -1 pokud jsou plně rozdílné.

Při zachování stejných hodnot pro generování obrázků pomocí poskytnutého skriptu a DT můžeme vidět porovnání na Obr. 3.2



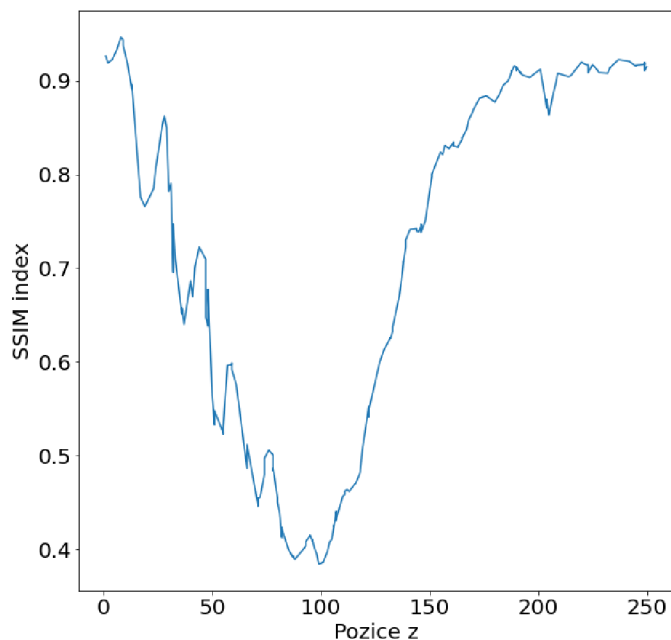
Obr. 3.2: Porovnání obrazů s původními hodnotami. První obraz zleva je vytvořený pomocí MatLab skriptu, uprostřed je obraz vytvořený pomocí Deeptrack 2.1 a poslední obraz je porovnání radiálních profilů zmíněných obrazů, přičemž SSIM index vyšel zaokrouhleně 0.05130. V popisku jsou poté uvedeny parametry pro danou částici.

Z prvních dvou obrazů z 3.2 můžeme vyvodit, že nejsou podobné, což potvrzuje porovnání radiálních profilů i SSIM index s hodnotou 0.05130. Obrazy vytvořené pomocí DT se jeví blíže směrem ke kameře v ose z. Tato chyba nejspíše vychází z MatLab skriptu, pomocí něhož byly vypočítány radiální profily pro vytváření umělých dat. Proto byla zavedena korekční konstanta o hodnotě 0.532. Tato konstanta vychází z vlnové délky použitého laseru a nastavení parametrů ve skriptu pro vytváření radiálních profilů poskytnutého z ÚTP AV ČR. Na Obr. 3.3 je již použita korekční konstanta.



Obr. 3.3: Porovnání obrazů s korekční konstantou. První obraz zleva je vytvořený pomocí MatLab skriptu, uprostřed je obraz vytvořený pomocí Deeptrack 2.1 s korekční konstantou a poslední obraz je porovnání radiálních profilů zmíněných obrazů, přičemž SSIM index vyšel zaokrouhleně 0.5095. V popisku jsou poté uvedeny parametry pro danou částici.

Pro upravenou osu z o korekční konstantu vychází SSIM index s hodnotou 0.5095 podstatně lépe než v předchozím pokusu. Hodnota SSIM platí pouze pro tento konkrétní případ. Pro ověření na celém intervalu na ose z bylo vygenerováno 1000 obrazů a porovnány a výsledky byly zaneseny do Obr. 3.4

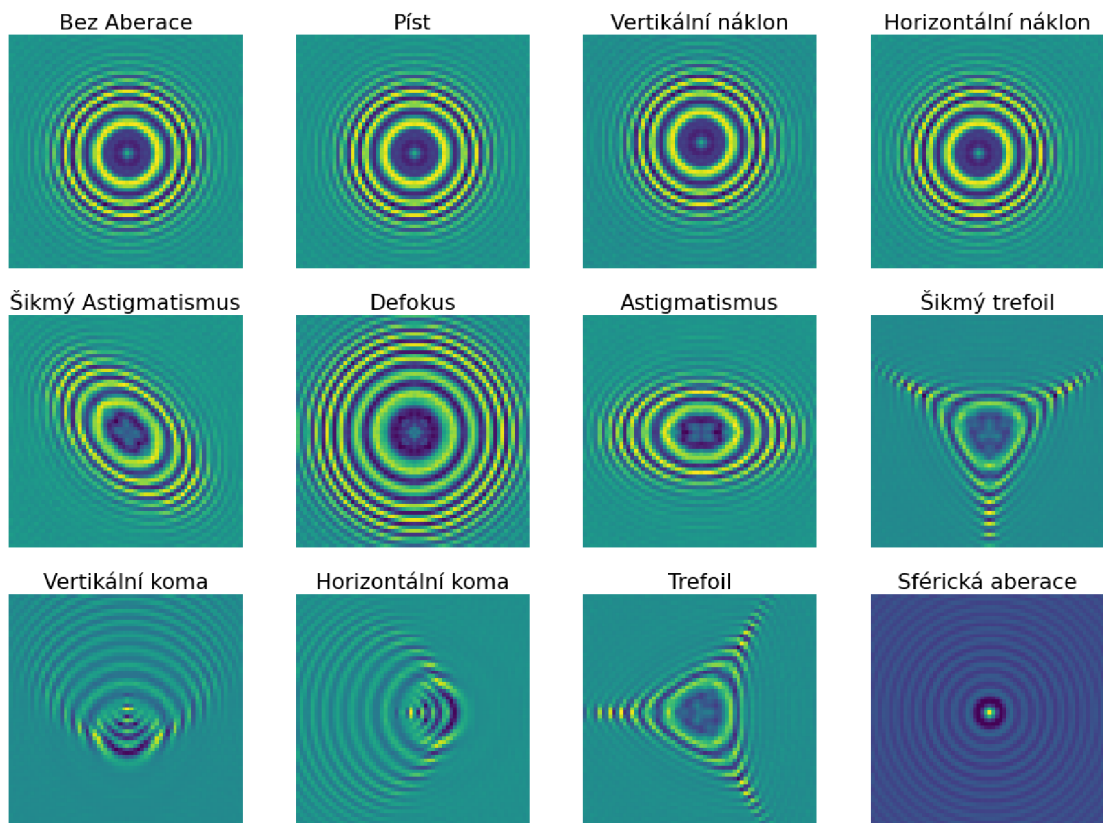


Obr. 3.4: Graf SSIM indexu v závislosti na ose z obrazu.

Podle grafu výše je možné říct, že porovnávané obrazy jsou nejpodobnější nej-dále od kamery a zároveň nejbliže. Malý index SSIM v oblasti 50 až 150 může být způsoben posunem radiálního profilu, nebo vymizením vln vyskytujících dále od středu. Pokud budeme počítat s odchylkou v rámci výsledků můžeme hodnotu indexu považovat za přijatelný a využít data pro trénování a validaci.

3.1.4 Aberované obrazy

Aberované obrazy jsou označovány jako obrazy s určitou odchylkou či poškozením. V případě pozorování částic se jedná o optické aberace způsobené nedokonalostmi optických čoček. Aby natrénovaný model byl robustnější je možné použít aberované obrázky pro další trénování. DT poskytuje vytvoření aberace na vygenerovaných obrazech. Dostupné aberace lze vidět na Obr. 3.5.



Obr. 3.5: Typy aberací vytvořených prostřednictvím Deeptack 2.1.

3.1.5 Modely a trénování v rámci Deeptack 2.1

DT umožňuje jednoduchý a uživatelsky přívětivý návrh modelů hlubokého učení. Modely jsou postaveny na programovém balíčku Keras spojený s balíčkem Ten-

sortFlow poskytující nepřehledné množství návrhu strojového učení. DT nabízí již implementované modely CNN, U-NET, RNN a GAN (angl. *generative adversarial networks*). V rámci aktualizace DT přidali autoři specifické modely LodeSTAR (model pro detekci částic bez značení) a MAGIK (model na grafové bázi pro propojení a stopování částic). Díky naprogramované nástavbě složitost návrhu modelu je minimální a stačí jenom zadání hyperparametrů jako počet učebních epoch, velikost dávky, počet vrstev a výstupů, aktivační funkce napříč vrstvy, kritériální funkci a velikost vstupního obrazu. Výhodou tohoto přístupu je jednoduchost, přehlednost a díky flexibilitě návrhu knihovny je možné přidávat i programové řešení z balíčku Keras, které nejsou součástí DT. Výhody zde mají své omezení a to v jednoduchosti nástavby, která může vyústit neobratnost návrhu, kdy není možná editace a návrh jednotlivých vrstev. Tento fakt poté může ovlivňovat úspěšnost modelu.

Po definování modelu je důležité vytvořit generátor dat. Úlohou generátoru je dodávání dávek trénovacích dat při trénování navržené sítě. DT poskytuje tři typy generátorů, a to standartní, průběžný a omezený průběžný. Hlavním rozdílem mezi těmito variantami je způsob, jakým generují data. Standartní generátor vytvoří zadaný maximální počet dat a podle velikosti vstupní dávky dodává data modelu. Souběžný generátor má za cíl urychlit trénování vyvážením tvorby nových obrázků a znovu použití již vygenerovaných obrázků. Nakonec omezený souběžný generátor má stejné schopnosti jako souběžný, ale s limitací pro konkrétní obraz. Limitace spočívá v maximálním použití v trénování pro daný obraz.

3.2 Návrh metodiky pro určení parametrů částic

Pro určení parametrů částice z holografických snímku podle kap.2 je možné využít heuristického přístupu, či metod hlubokého učení založených na konvolučních operátorech jako CNN nebo U-Net a jejich modifikace.

První variantou pro určení pozice jedné částice je možné využití CNN. Nicméně při aplikaci na větší počet částic je CNN nepoužitelná a musí být rozšířena o další metody pro lokalizaci difrakčních vzorů (např. YOLOv3). Přístupy založené na této metodice byly využívány v pracích zmíněných v kap.2.

Další možnou cestou je využití architektury U-Net pro segmentaci středu částice. Jelikož výstupem U-Net je segmentovaný obraz, odpadá nutnost rozšiřování o lokalizaci difrakčních vzorů. Ve segmentovaném obrazu stačí lokalizovat střed částice.

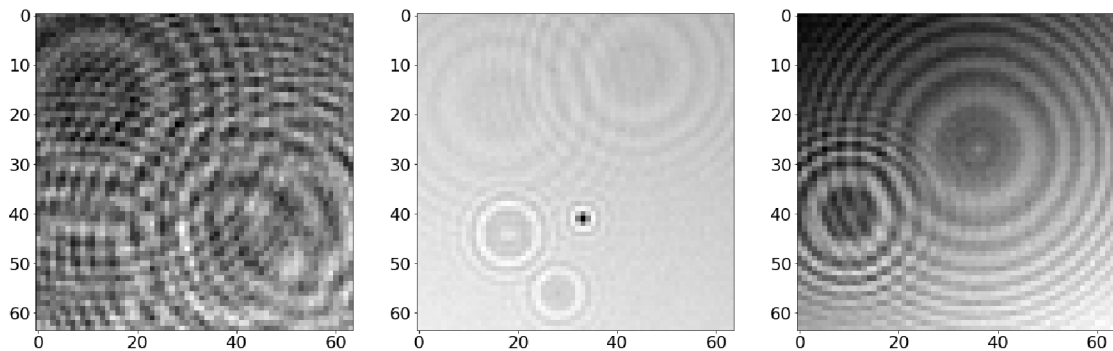
K určení zmíněných parametrů existují i další přístupy např. fitování pomocí Mie koeficientů zmíněných v podkap. 1.4.2.

Pro tuto práci byla zvolena varianta využívající topologii U-Net pro její jednoduchost a robustnost při zpracování většího počtu částic. Od vybrané varianty se poté odvíjí předzpracování trénovacích dat, volba parametrů sítě a další algoritmy.

Pro srovnání modelu vytvořeného pomocí DT byl implementován model U-Net v aplikačním rámci PyTorch.[34] Návrhu a optimalizaci se poté věnuje tato podkapitola.

3.2.1 Předzpracování a vytvoření trénovacích dat

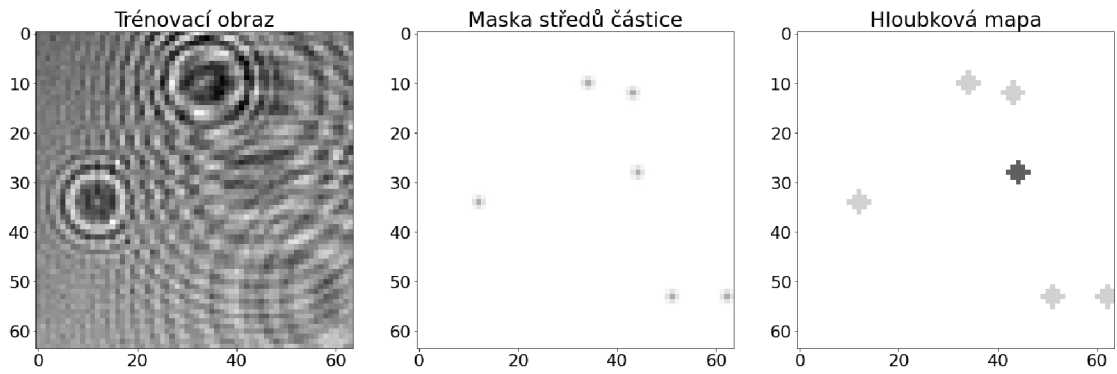
Předzpracování dat pro trénování sítí je nedílnou součástí, protože zásadně ovlivňuje přesnost modelu. Pro trénování byly vytvořeny soubor dat z uměle vytvořených obrazů skrze DT. Celkové počty jsou uvedeny v jednotlivých optimalizacích. Příklad dat můžeme vidět na Obr.3.6



Obr. 3.6: Příklad trénovacích dat vytvořeného pomocí skriptu *datamaker.py*.

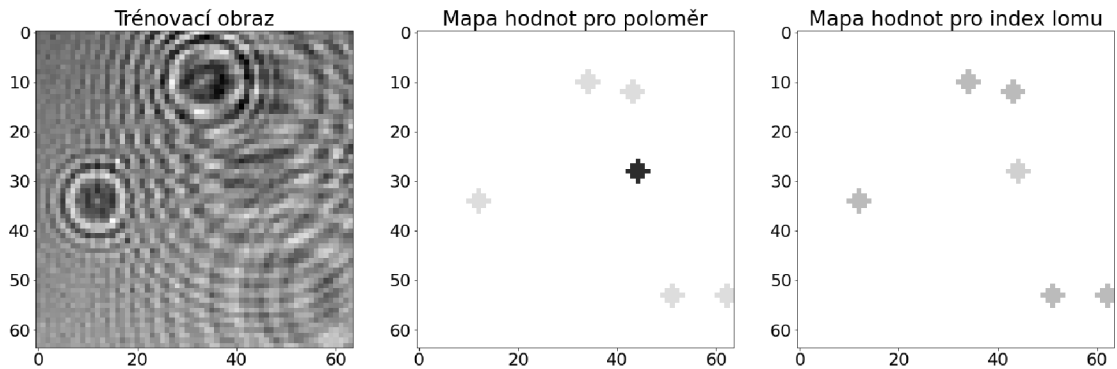
Aby bylo zajištěno správné trénování konvolučního modelu potřebujeme z trénovacích obrazů vytvořit masky, které budou nositeli informace a ideální výstupem z implementovaného modelu. Pro detekci 2D pozice byla implementována funkce *gaus_img()* ze skriptu *func_holo.py*. Funkce *gaus_img()* vytváří v novém nulovém obraze 2D Gaussovo rozložení s nejvyšším vrcholem na středu částice. Pro vytvoření 2D Gaussova rozložení v obraze byla použita funkce *gaussian_filter()* z programového balíčku Scipy. [35]

Pro odhad hodnoty osy z byly vytvořeny hloubkové mapy pomocí funkce *depth_map()*. V novém nulovém obraze do souřadnic středu částic jsou zaneseny hodnoty osy z . Pomocí morfologické operace dilatace jsou zvětšeny do tvaru disku o definované šířce, přičemž se využívá metody povodí, aby nedocházelo k nechtěnému překryvu hodnot. Metody jsou převzaty z programového balíčku scikit-image. [36] Mapy jsou normovány podle definovaného intervalu, aby nedošlo k ztrátě informace. Interval je dán maximální a minimální hodnotou z tab.3.2. Vytvořené hloubkové mapy a masky můžeme vidět na Obr. 3.7.



Obr. 3.7: Příklad masky pro určení polohy ve 2D a hloubkové mapy pro osu z.

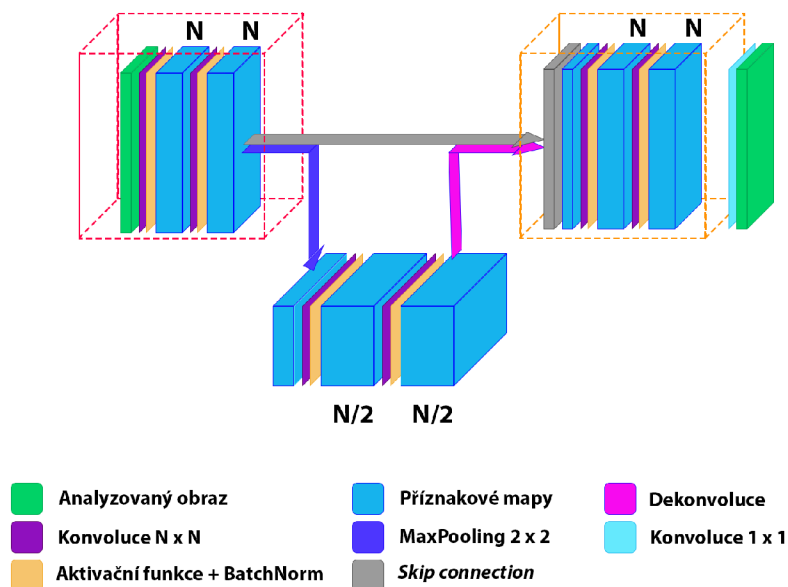
Obdobným způsobem je funkce využita pro vytvoření hodnotové mapy pro index lomu částice a jejího poloměru s tím rozdílem, že byly použity jiné normovací intervaly, ty taktéž můžeme nalézt v Tab.3.2 a příklady vytvořených map jsou vizualizovány na Obr.3.8.



Obr. 3.8: Příklad map hodnot pro index lomu a poloměr částice.

3.2.2 Návrh a optimalizace topologie U-Net pro určení parametrů

Návrh topologie vychází z popisu U-Net v podkap. 2.3 a její obecný diagram je vizualizován na Obr.3.9.



Obr. 3.9: Obecný diagram topologie U-net použitý v rámci závěrečné práce. N aktuální počet konvolučních jader.

Před samotným použitím je důležité určit parametry sítě jako počet konvolučních jader, počet trénovacích epoch, kritériální funkci, optimalizační algoritmus a jeho hyperparametry.

Jako optimalizační algoritmus byl zvolen ADAM. ADAM je metoda pro stochastickou optimalizaci založená na individuálním adaptivním výpočtu učebního kroku pro různé parametry z prvního a druhého momentu gradientu. Nastavitelné parametry u ADAM jsou učební krok α a sestupný krok pro odhad momentu β_1 , β_2 a koeficient ϵ pro zvýšení numerické stability. Přičemž výchozí nastavení autoři doporučují $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ a $\epsilon = 10^{-8}$. [38] V případě našeho návrhu bude jediným nastavitelným parametrem krok učení α . Dalším krokem je správné vybrání kritériální funkce, protože výpočet chyby má velký vliv na nastavení vah a tedy celkovou účinnost sítě. Klasickým přístupem pro segmentaci by byl výběr funkce z binárně založených funkcí jako Dice loss, Focal loss nebo klasifikačně založených funkcí např. Vzájemná informace a její modifikace. [39] V případě segmentování Gaussovských map jsou binárně založené kritériální funkce nevhodnou volbou a problém segmentace leží spíše v oblastech pro kritériální funkce pro regresní modely jako L1, L2 či Hubber kritériální funkce. V rámci některých modifikací kritériálních funkcí je nutné nastavit koeficient, či váhu jakou budou modifikovat výpočet chyby.

Tyto parametry budou zahrnuty do optimalizačního procesu.

Proces nastavení a volby hyperparametrů jako počet konvolučních jader, krok učení apod. by byl časově náročný pouhou empirickou metodou. Abychom tomu předešli využijeme optimalizační algoritmus RayTune [33]. Raytune je knihovna založenou na jazyce python používána pro ladění hyperparametrů. Po implementaci trénovacího kódu do prostředí je nutné zvolit intervaly hyperparametrů, jejich kombinace bude Raytune aplikovat při trénování. Intervaly hodnot můžeme najít v tabulce 3.3.

Tab. 3.3: Přehled hyperparametrů modelu a optimalizátoru pro trénování.

Parametr	Hodnota	Pozn.
Krok učení α	$[1 \cdot 10^{-5}, 1 \cdot 10^{-1}]$	
Počet konv. jader	[8, 16, 32, 64, 128, 256]	
Aktivační funkce	ReLU, Leaky ReLU, GELU	
Váha δ	$[1 \cdot 10^{-4}, 1]$	Učení s Hubber Loss
Velikost objektu	[2, 3, 4, 5, 6]	Učení s Hloubkovou mapou
Rozptyl σ	[0.1, 2]	Učení 2D pozice

Abychom otestovali všechny kombinace z tabulky výše, bylo by potřeba nespočtu jednotlivých pokusů, což je časově a výpočetně náročné. Proto je v optimalizačním prostředí Raytune definován algoritmus předčasného zastavení a prohledávání.

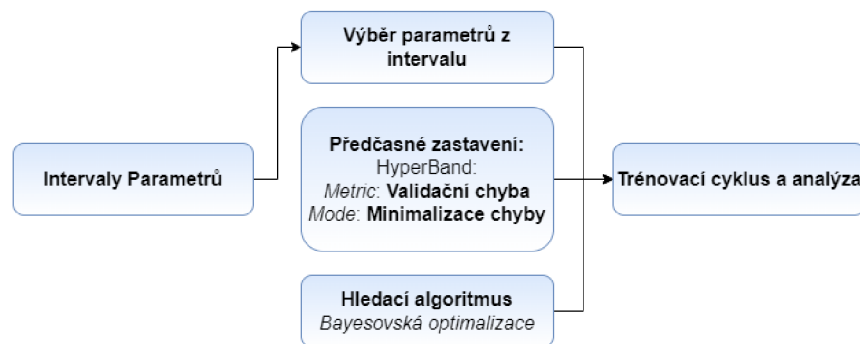
V rámci předčasného zastavení autoři Raytune doporučují metodu ASHA. ASHA je asynchronní podobou HyperBand. Metoda je založena na postupném půlení, kdy z prostoru pro hyperparametry (v našem případě prostor reprezentuje tab. 3.3) náhodně vybere kombinace parametrů. Ty podstoupí proces trénování o 1 epoše. Po dokončení procesu metoda rozdělí modely podle metriky na dvě poloviny v závislosti na výsledku. Horší polovina je terminována a druhá úspěšnější polovina přechází do další epochy, kdy je pouštěn další trénovací cyklus. Po cyklu jsou znovu vyhodnoceny a podstupují stejný proces jako při první epoše. Tento proces se opakuje do doby, než zůstane poslední nejlepší kombinace. [40]

Dále je nastaven prohledávací algoritmus. Výchozí nastavení v Raytune má podobu náhodného a mřížkového prohledávání. Nicméně tento přístup není vhodný pro velké počty hyperparametrů, protože v příkládá pro všechny kombinace stejnou váhu, což prodlužuje učební čas a tím i výpočetní výkon. Z tohoto důvodu byla hledána alternativa. Autoři z článku [41] doporučují využití Bayesovské optimalizace.

S využitím Bayesovská optimalizace vzniká problém s tím, že samotný algoritmus z principu podporuje pouze spojité intervaly hodnot nikoliv diskrétní, či kategorické hodnoty. Problému se dá předejít programovým překódováním vyloučeného typu hodnot. Tato cesta nicméně sníží přesnost optimalizace, a proto bude od ní odpuštěno.

Nicméně autoři v předchozím článku uvádějí, že kombinací výše zmíněných přístupů je kompatibilní a je známá pod zkratkou BOHB, přičemž Raytune disponuje tímto algoritmem, proto bude využit pro optimalizaci.

Celý proces nastavení Raytune můžeme vidět na diagramu 3.10. Po ukončení procesu optimalizace je vypsána tabulka s hodnotami vyzkoušených kombinací hyperparametrů a nejlepší dosaženou konfigurací modelu. Nejlepší konfigurace modelů pro jednotlivé parametry jsou představeny v podkapitolách níže.



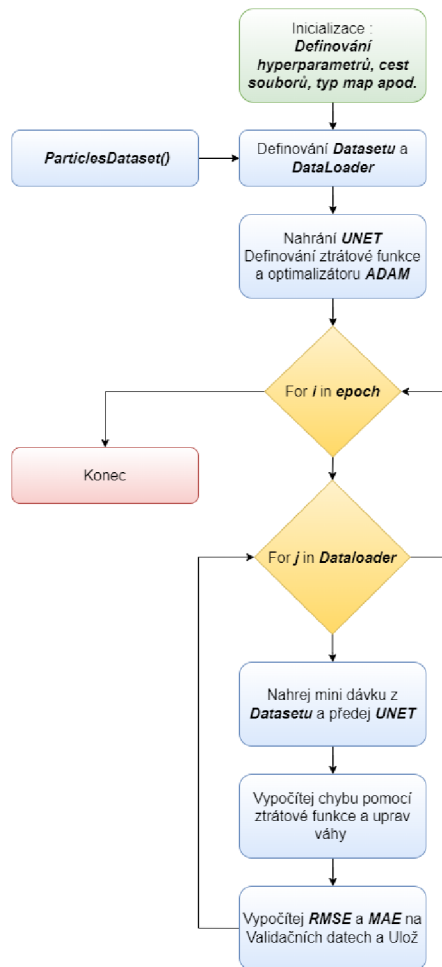
Obr. 3.10: Diagram postupu ladění hyperparametrů

Protože výstupem Raytune je tabulka a při větším počtu kombinací se stává nepřehledná jsou výstupy graficky znázorněny pomocí vizualizačního nástroje TensorBoard z knihovny TensorFlow. [37] O výsledném modelu bude rozhodnuto v závislosti na validační chybě, RMSE a MAE vypočítávané v průběhu optimalizace.

3.2.3 Navrhnutí procesu trénování modelu

Jak bylo zmíněno v kapitole 3.1.5 pro trénování modelů jsou nutné generátory dat, které dodávají modelům trénovací data. Tuto úlohu u trénování modelu navrženém v PyTorch zprostředkovává funkce z téže knihovny *DataLoader()*. Vstupem funkce je trénovací, nebo validační soubor dat (v této práci realizován skriptem *dataset.py*), velikost dávky, promíchání pořadí dat a počet pod procesů použitých při načítání dat ze souboru.

Skript *dataset.py* obsahuje třídu *ParticlesDataset()*. Funkcí třídy je načíst obrázky částic a vytvořit masky, hloubkové a hodnotové mapy v závislosti na vstupních parametrech. Celý proces je shrnut na diagramu 3.11.



Obr. 3.11: Shrnutí procesu učení a validace diagramem

3.3 Výsledky optimalizace modelu U-Net na aberovaných datech

Celý proces optimalizace proběhne pouze pro model (3.9) navrhnutý v PyTorch. Hodnoty hyperparametrů pro nejlépe hodnocený model budou přeneseny do aplikačního rámce *DT* a natrénovány odděleně. Postup trénování je uveden v podkap. 3.5. Výsledky na testovacích datech mezi modely budou porovnány v kap.4.

Velikost trénovacího souboru pro neaberované data byl 131072 a validační soubor dat obsahoval 8192. Data obsahují aberované i neaberované data. Náhodným způsobem je vybrán koeficient vlivu aberace a typ zmíněný v podkap. 3.1.4. Tyto obrazy jsou poté přidány do souboru dat. Trénování proběhlo skrze virtuální organizaci MetaCentrum na grafické kartě NVidia A40.

V první optimalizaci bylo zvoleno 40 epoch učení a 50 konfigurací. Po zhodnocení

výsledků tyto parametry byly sníženy na 20 epoch, z důvod neměnnosti validační chyby, které byla použita jako metrika pro minimalizaci a zrychlení procesu optimalizace. Počet procesů byl závislý na vybraných kriteriálních funkcích. Hodnoty nejlepších modelů pro každou funkci byly zaneseny do tabulek níže. Může nastat situace, kdy Raytune metriky vyhodnotí jako dostačující a dále s trénováním nepokračuje. V tomto případě budou modely dotrénovány a znovu otestovány.

3.3.1 Výsledky optimalizace pro určení 2D polohy

Pro optimalizaci modelů pro určení 2D polohy byly vybrány kriteriální funkce L2 a Hubber funkce. Další možností je využití L1 kriteriální funkce, nicméně podle popisu Hubber funkce v podkap. 2.3.1 je kombinací L1 a L2, kdy L1 funkce popisuje Hubber funkci z větší části, proto budou využity pouze zmiňované typy.

Tab. 3.4: Parametry nejlepších modelů pro určení 2D pozice pro různé kriteriální funkce.

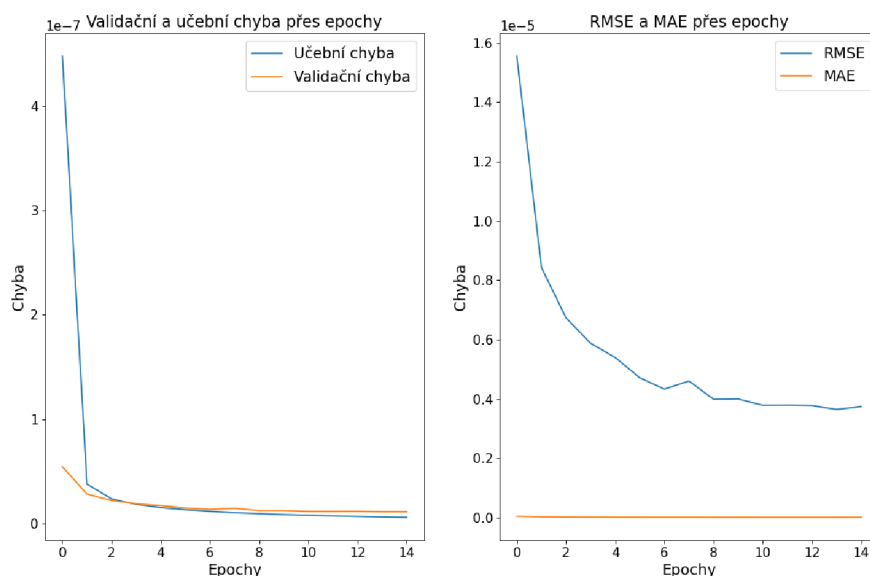
K. funkce	α	Jádra	Aktivace	Dávka	Sigma	Delta
L2	0.0000939	[64, 128]	GELU	15	1.92	-
Hubber	0.005185	[32, 64, 128, 256]	L. ReLU	29	1.208	0.000267

Přičemž výsledky chyb zaznamenávaných během procesu na validační množině jsou uvedeny v tabulce 3.5.

Tab. 3.5: Chybovost nejlepších modelů pro určení 2D pozice pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2	$4.2 \cdot 10^{-7}$	$1.4 \cdot 10^{-8}$	$3.59 \cdot 10^{-6}$	15
Hubber	$1.18 \cdot 10^{-8}$	$1.156 \cdot 10^{-8}$	$2.96 \cdot 10^{-6}$	20

Průběh učení vybraného modelu je zachyceny na obr. 3.12. Strmost grafu učení poukazuje na velký krok učení.



Obr. 3.12: Průběh učení modelu optimalizaci nejlepší modelů pro určení 2D polohy učený na aberovaných datech.

Při porovnání mezi modely z tab. 3.4 je model s kriteriální funkcí Hubber má více vrstev, tudíž dokáže vyextrahovat více informace z obrázku. Zároveň podle optimalizace bude vybrán model s nejnižší chybovostí tedy, model s kriteriální funkcí Hubber.

3.3.2 Výsledky optimalizace pro určení polohy z

Jako kriteriální funkce pro optimalizaci predikce polohy v ose z byly vybrány L2 a Hubber. Dalším krokem je omezení aplikace ztrátové funkce pouze na vybrané části v obrazu. Tato aplikace byla vyzkoušena na a předpokladu zvýšení přesnosti určení parametrů. Zaměření na tyto části bude provedena vybráním částí obrazu podle trénovací masky.

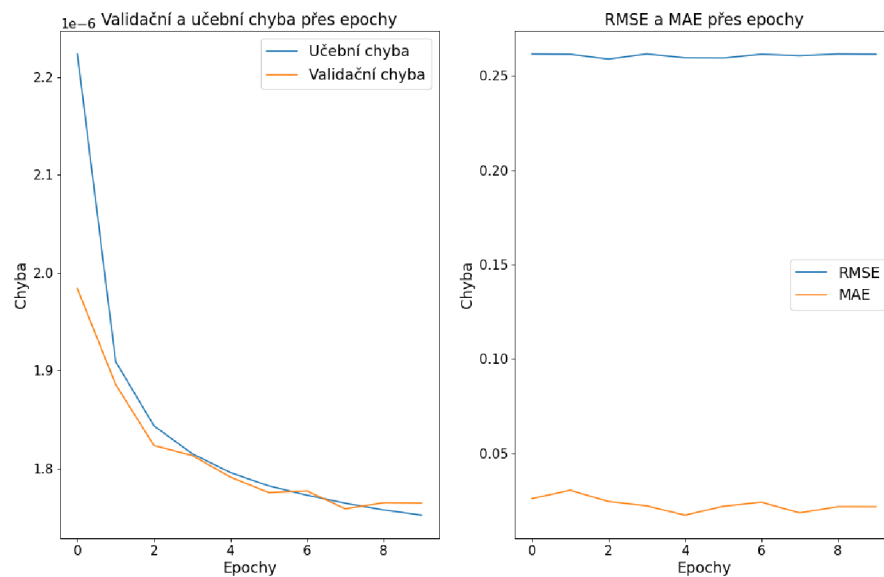
Z výsledných obrazu je možné říci, pokud použijeme segmentaci s výpočtem chyby jenom z oblasti zájmu je nutné použít apriorní informaci o 2D pozici částici, aby bylo možné extrahovat správnou hodnotu z.

Tab. 3.6: Parametry nejlepších modelů pro určení polohy z pro různé kriteriální funkce.

K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.0429	[8, ..., 256]	LReLU	32	5	-
Hub (maska)	$7.08 \cdot 10^{-5}$	[16, 32]	GELU	90	5	$1.05 \cdot 10^{-5}$
Gauss L2	0.0011	[8, 16]	GELU	98	5	-

Tab. 3.7: Chybovost nejlepších modelů pro určení z polohy pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.0540	0.0431	0.229	15
Hub (maska)	$1.7 \cdot 10^{-5}$	0.0407	0.254	14
Gauss L2	$7.9 \cdot 10^{-5}$	0.048	0.23	10



Obr. 3.13: Průběh učení modelu optimalizací nejlepší modelů pro určení polohy z učený na aberovaných datech.

Jako testovací model bude vybrán model s kriteriální funkcí Hubber, z důvodu nejnižších validačních chyb.

3.3.3 Výsledky optimalizace pro určení poloměru

Motivace ke zvolení právě těchto kritériálních funkcí vychází z motivace podobné jako v podkap. 3.3.2

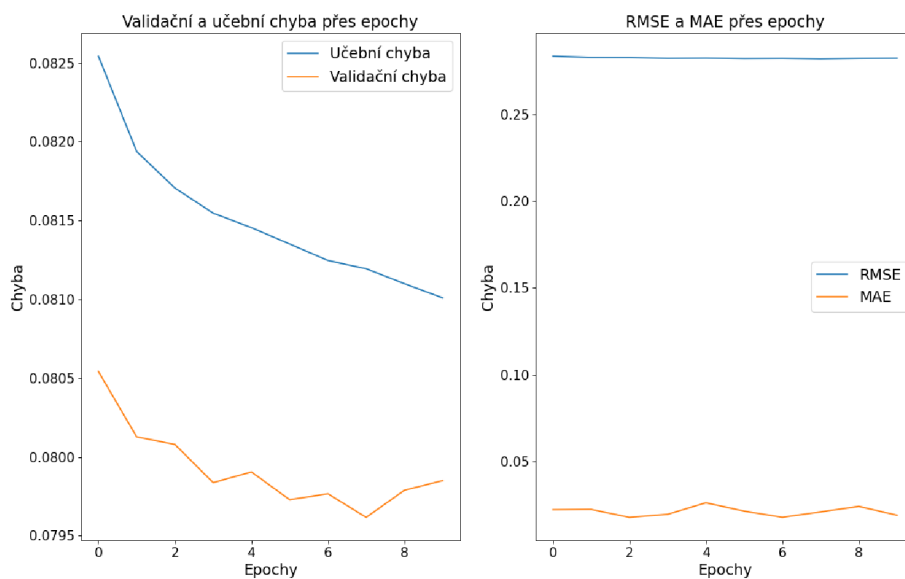
Tab. 3.8: Parametry nejlepších modelů pro určení poloměru pro různé kritériální funkce.

K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.000968	[8, 16, 32, 64]	ReLU	98	3	-
Hubber (maska)	$1.54 \cdot 10^{-5}$	[16, 32]	GELU	115	3	0.00011
Vážené L2	0.0021	[32, ..., 256]	GELU	89	5	-

Tab. 3.9: Chybovost nejlepších modelů pro určení poloměru pro různé kritériální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.0819	0.0543	0.2847	10
Hubber (maska)	$2.837 \cdot 10^{-8}$	0.0485	0.286	15
Vážené L2	0.0011	0.0545	0.285	10

Na základě tab. 3.9 a výsledných hodnot chyb byl vybrán model s kritériální funkcí L2.



Obr. 3.14: Průběh učení modelu optimalizací nejlepší modelů pro určení poloměru učený na aberovaných datech.

3.3.4 Výsledky určení indexu lomu

Motivace ke zvolení právě těchto kriteriálních funkcí vychází z motivace podobné jako v podkap. 3.3.2

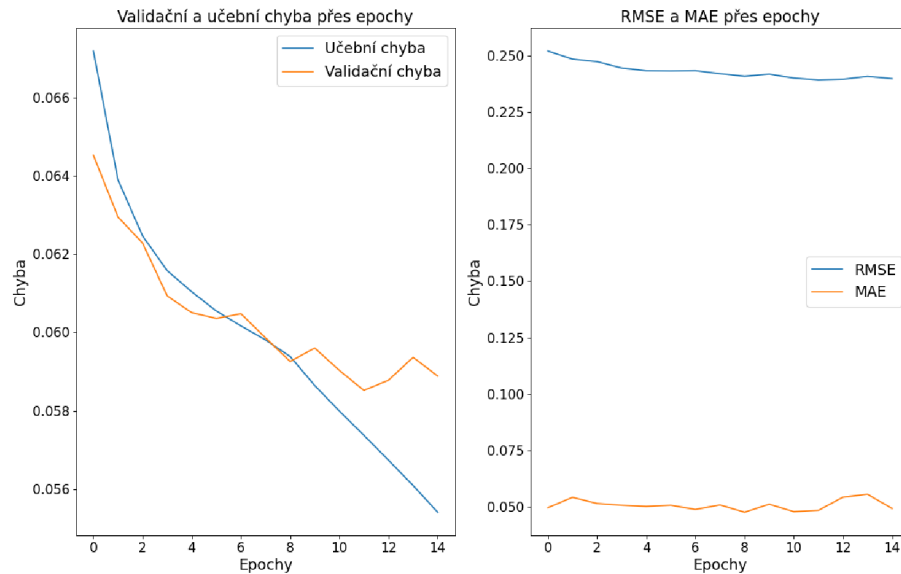
Tab. 3.10: Parametry nejlepších modelů pro určení indexu lomu pro různé kriteriální funkce.

K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.0015	[16, 32, 64]	GELU	14	3	-
Hubber (maska)	0.0198	[128, 256]	GELU	124	5	0.000103
Vážené L2	0.0009	[16, ..., 256]	ReLU	50	5	-

Tab. 3.11: Chybovost nejlepších modelů pro určení indexu lomu pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.0599	0.0467	0.242	15
Hub (maska)	$2 \cdot 10^{-5}$	0.056	0.2601	10
Gauss L2	$8.4 \cdot 10^{-4}$	0.0507	0.243	10

Na základě nejmenší chyby MAE a RMSE z tab. 3.16 je zvolen testovací model s kriteriální funkcí L2.



Obr. 3.15: Průběh učení modelu optimalizaci nejlepší modelů pro určení indexu lomu učený na aberovaných datech.

3.4 Výsledky optimalizace modelu U-Net na neaberovaných datech

Velikost trénovacího souboru pro neaberované data byl 65536 a validační soubor dat obsahoval 8192. Všechny modely kromě modelu pro určení 2D polohy konvergovali extrémně rychle, přičemž validační chyba poukazovala na přeučení. Z tohoto důvodu byly modely naučeny pouze do bodu s nejnižší validační chybou.

3.4.1 Výsledky pro určení 2D polohy

Tab. 3.12: Parametry nejlepších modelů pro určení 2D pozice pro různé kriteriální funkce.

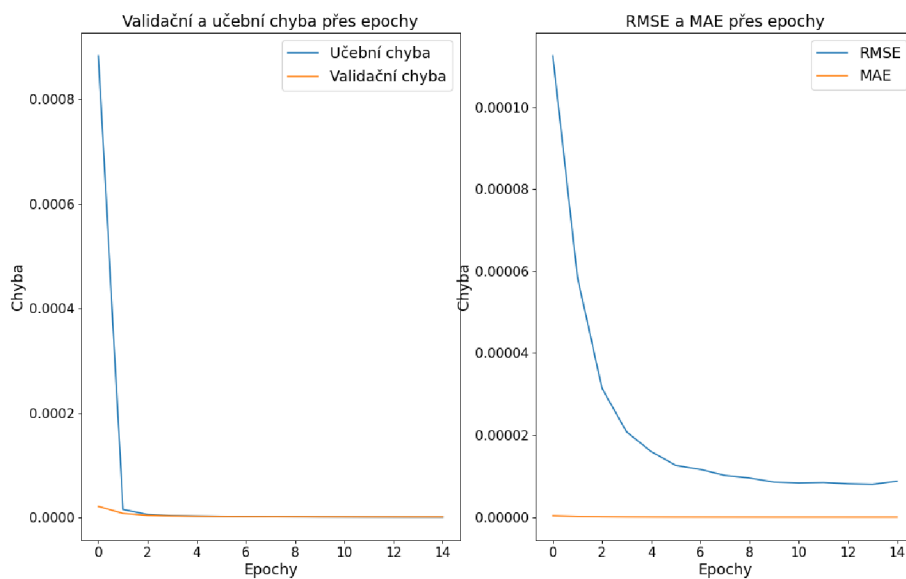
K. funkce	α	Jádra	Aktivace	Dávka	Sigma	Delta
L2	0.0104	[16, 32, 64, 128, 256]	ReLU	28	1.71	-
Hubber	0.00290	[128, 256]	L. ReLU	29	0.754	0.00013

v porovnání s modelem učených na kombinaci aberovaných a čistých datech s Hubber kriteriální funkcí jsou si parametry obou modelů velmi podobné. Optimalizační algoritmus zvolil stejnou aktivační funkci, koeficient delta, počet jader v konvolučních filtrech apod.

Tab. 3.13: Chybovost nejlepších modelů pro určení 2D učených na datech bez aberaci pozice pro různé kriteriální funkce a průběžnou chybu.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2	$1.4 \cdot 10^{-6}$	$2.65 \cdot 10^{-8}$	$6.797 \cdot 10^{-6}$	12
Hubber	$2.29 \cdot 10^{-8}$	$4.03 \cdot 10^{-8}$	$1.03 \cdot 10^{-5}$	12

Pro určení 2D polohy vybereme model využívající kriteriální funkci L2. Tento model nedisponuje nejmenší validační chybou, ale chyba MAE a RMSE jsou značně menší.



Obr. 3.16: Průběh učení modelu učených na datech bez aberaci vybraného po optimalizaci pro určení 2D polohy.

3.4.2 Výsledky určení polohy z

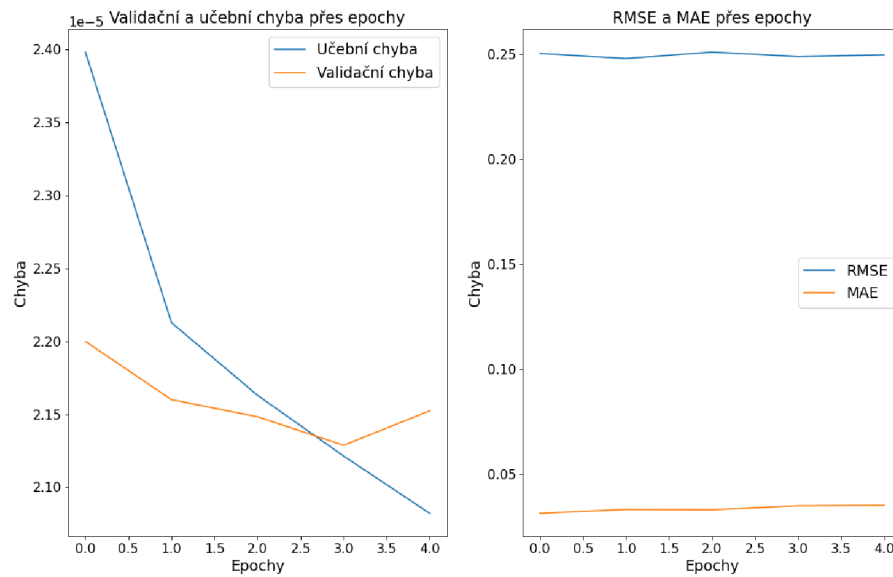
Tab. 3.14: Parametry nejlepších modelů pro určení polohy z na neaberovaných datech pro různé podoby kriteriální funkce.

K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.0027	[64, 128]	LReLU	99	5	-
Hubber (maska)	0.00013	[16, ..., 256]	ReLU	28	3	0.00013
Vážené L2	$2.8 \cdot 10^{-5}$	[128, 256]	ReLU	47	5	-

Tab. 3.15: Chybovost nejlepších modelů učených na neaberovaných pro určení polohy z pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.054	0.048	0.23	15
Hubber (maska)	$2.44 \cdot 10^{-5}$	0.049	0.264	14
Vážené L2	0.0011	0.043	0.258	14

Pro vybrání modelu pro určení polohy z je sporné. Pokud by se rozhodovalo podle validační chyby byl by vybrán model s kriteriální funkcí Hubber. Model s nejnižší hodnotou MAE využívající L2 a nejmenším RMSE disponuje model s kriteriální funkcí L2 váženou Gaussovým obrazem. Pokud uvážíme celkovou nejmenší chybu mezi modely, tak model s Váženou L2 funkcí je nejlepší, proto bude zvolen pro další analýzu



Obr. 3.17: Průběh učení modelu vybraného po optimalizaci učený na neaberovaných datech pro určení polohy z.

3.4.3 Výsledky určení poloměru

Tab. 3.16: Parametry nejlepších modelů pro určení poloměru na neaberovaných datech pro různé podoby kriteriální funkce.

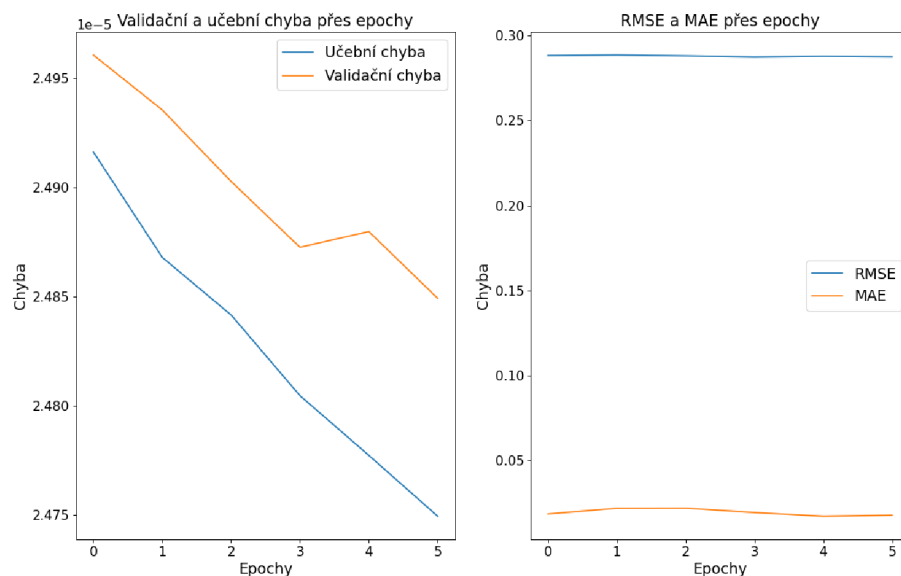
K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.0023	[16, 32, 64, 128]	GELU	102	2	-
Hubber (maska)	0.0012	[16, 32, 64, 128]	GELU	121	2	0.0001
Vážené L2	0.00054	[16, ..., 256]	ReLU	51	5	-

Tab. 3.17: Chybovost nejlepších modelů pro určení poloměru pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.082	0.05	0.285	10
Hubber (maska)	$2.5 \cdot 10^{-5}$	0.049	0.288	10
Vážené L2	0.0013	0.051	0.286	10

Pokud budeme vycházet z tabulky 3.17 pro odhad poloměru r , tak hodnoty RMSE jsou na desetiny stejné. Podobně jsou na tom hodnoty MAE, proto jako rozhodovací

kritérium poslouží validační chyba. Nejmenší validační chyba je pro model s funkcí Hubber.



Obr. 3.18: Průběh učení modelu vybraného po optimalizaci pro určení poloměru.

3.4.4 Výsledky určení indexu lomu

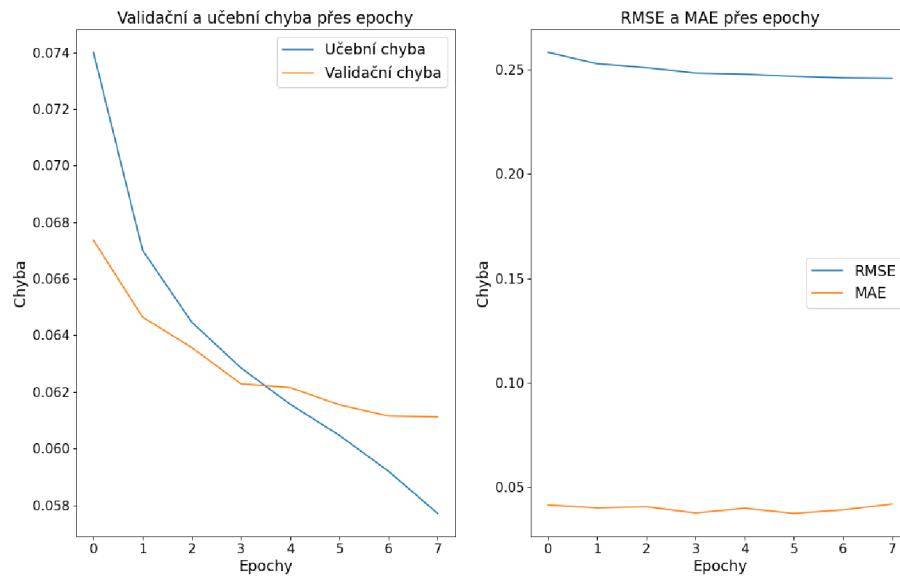
Tab. 3.18: Parametry nejlepších modelů pro určení indexu lomu pro různé kriteriální funkce.

K. funkce	α	Jádra	Aktivace	Dávka	Vel. obj.	Delta
L2 (maska)	0.001	[64, 128, 256]	ReLU	25	5	-
Hubber (maska)	0.00086	[32, 64, 128, 256]	GELU	30	3	0.0001
Vážené L2	0.0435	[16, 32, 64, 128]	LReLU	98	5	-

Tab. 3.19: Chybovost nejlepších modelů pro určení indexu lomu pro různé kriteriální funkce.

K. funkce	Validační ch.	MAE	RMSE	Učících iterací
L2 (maska)	0.061	0.049	0.245	10
Hubber (maska)	$2.03 \cdot 10^{-5}$	0.046	0.259	10
Vážené L2	0.001	0.056	0.252	10

K odhadu indexu lomu podle tab. 3.19 použita L2 funkce, jelikož disponuje nejmenší chybou.



Obr. 3.19: Průběh učení modelu vybraného po optimalizaci pro určení indexu lomu.

3.5 Trénování modelu v Deeptrack 2.1.

Pro trénování modelu v rámci DT byla použita vnitřní funkce *ContinuousGenerator()* popsána v 3.1.5, kdy minimální velikost souboru dat byla nastavena na 1024 a horní hranice byla nastavena na 2048.

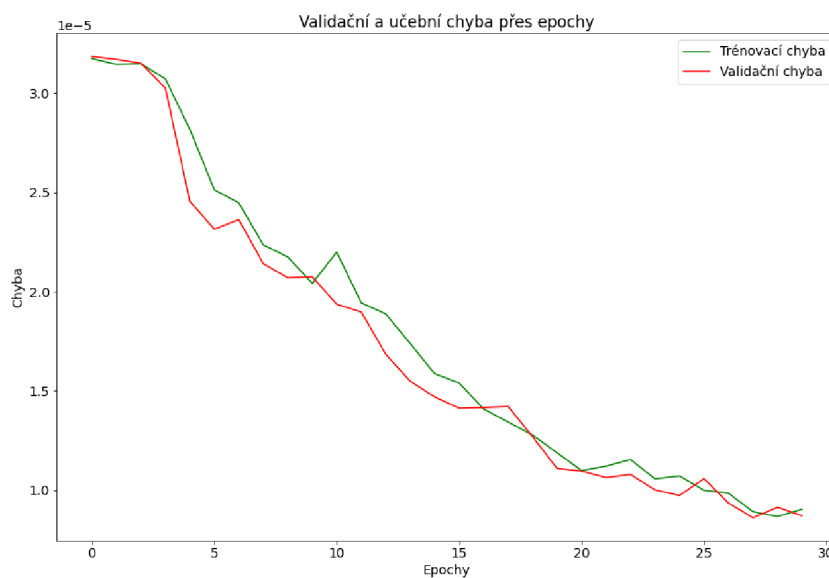
Modely založené na architektuře U-Net byly trénovány pouze na datech bez aberací. Parametry vybraných modelů v předchozí podkapitole budou přeneseny do aplikačního rámce a porovnány v kap. 4.

3.5.1 Trénování modelu pro určení 2D pozice

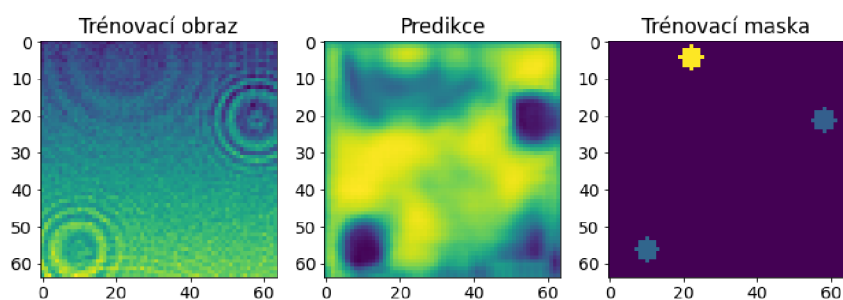
Pro určení pozice 2D pomocí DT byly přeneseny hodnoty pro model, který vyšel jako nejlepší v rámci optimalizace. Nicméně výsledky segmentace 2D Gaussových rozložení bylo neúčinné, proto byl pro tento parametr učení modelu vypuštěno. Křivka extrémně rychle konvergovala a výsledkem byl nulový obraz.

3.5.2 Trénování modelu pro určení polohy v ose z

Konfigurace modelu pro určení polohy v ose z byly přeneseny z tab. 3.14 pro kritériální funkci Hubber. Průběh učení je zaznamenán obr. 3.20. Křivka učení i validace má nemá ideální průběh, což může poukazovat na nízký krok učení. Křivka začíná k 25 epoše pomalu konvergovat. Na obr. 3.21 je možné vidět příklad detekce.



Obr. 3.20: Průběh učení modelu naprogramovaného v aplikačním rámci DT pro určení polohy z.



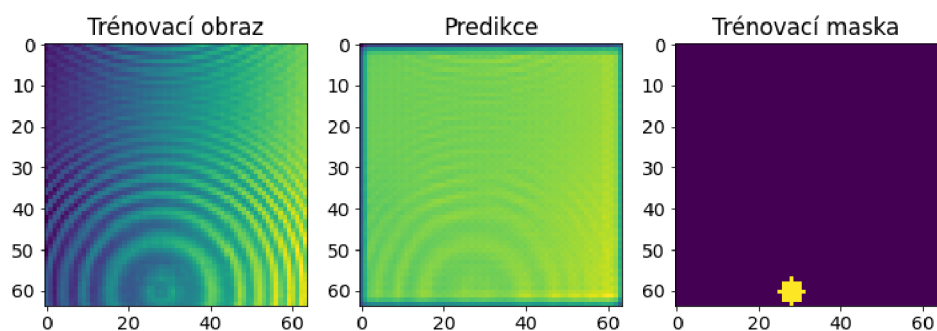
Obr. 3.21: Ukázka predikce modelu DT na validačních datech určující polohu z.

3.5.3 Trénování modelu pro určení poloměru

Konfigurace modelu pro určení poloměru byly přeneseny z tab. 3.16 pro kritériální funkci Hubber. Průběh učení je zaznamenán obr. 3.22. Křivka učení i validace má správný průběh a ke 15 epoše začíná pomalu konvergovat. Na obr. 3.23 je možné vidět příklad detekce. Predikční obraz nevypadá ideálně, nicméně podle validační chyby je detekce přijatelná, proto bude podstoupen další analýze.



Obr. 3.22: Průběh učení modelu naprogramovaného v aplikačním rámci DT pro určení poloměru.



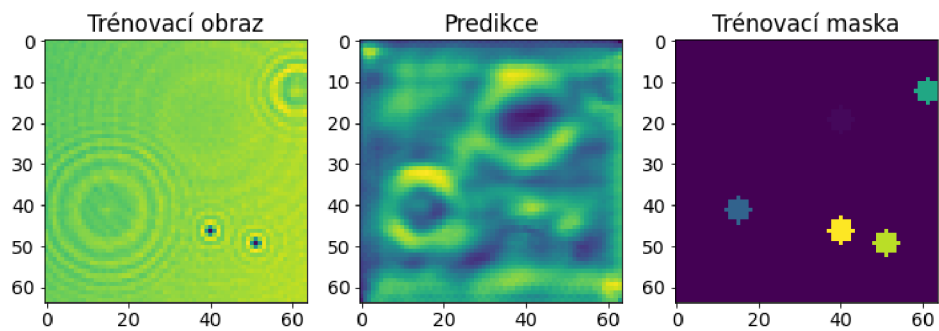
Obr. 3.23: Ukázka predikce modelu DT na validačních datech určující poloměr.

3.5.4 Trénování modelu pro určení indexu lomu

Konfigurace modelu pro určení indexu lomu byly přeneseny z tab. 3.18 pro kriteriální funkci Hubber. Průběh učení je zaznamenán obr. 3.24. Křivka klesá lineárně což může nasvědčovat pomalému kroku učení. K 30 epoše začíná pomalu konvergovat, nicméně to není jednoznačné a je doporučeno další doučení. Na obr. 3.25 je možné vidět příklad detekce.



Obr. 3.24: Průběh učení modelu naprogramovaného v aplikačním rámci DT pro určení indexu lomu.



Obr. 3.25: Ukázka predikce modelu DT na validačních datech určující index lomu.

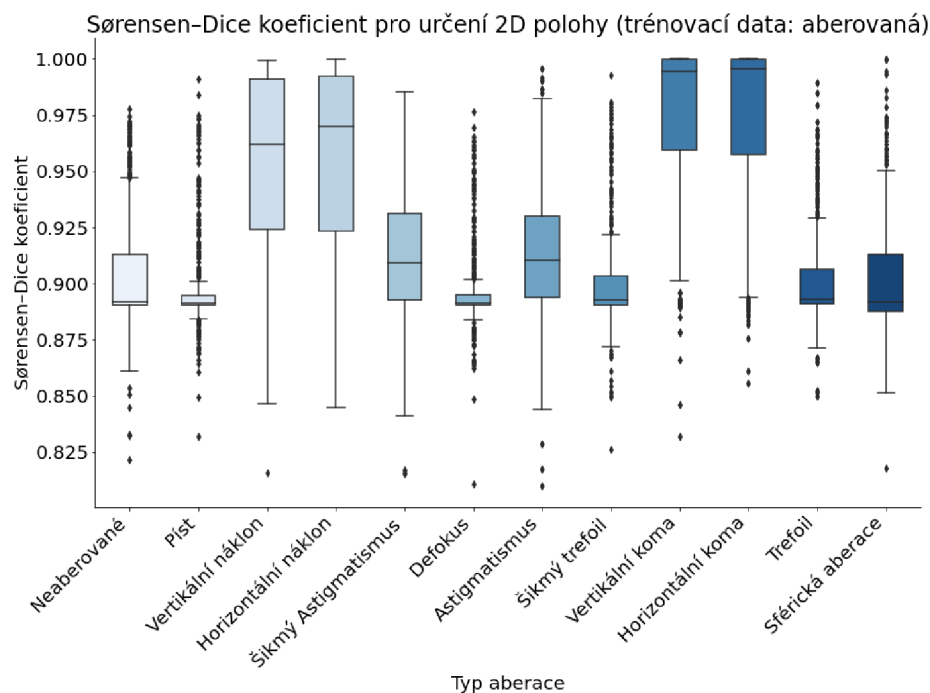
4 Testování modelů a aplikace na experimentální data

4.1 Aplikace modelů na data s aberacemi a bez aberace

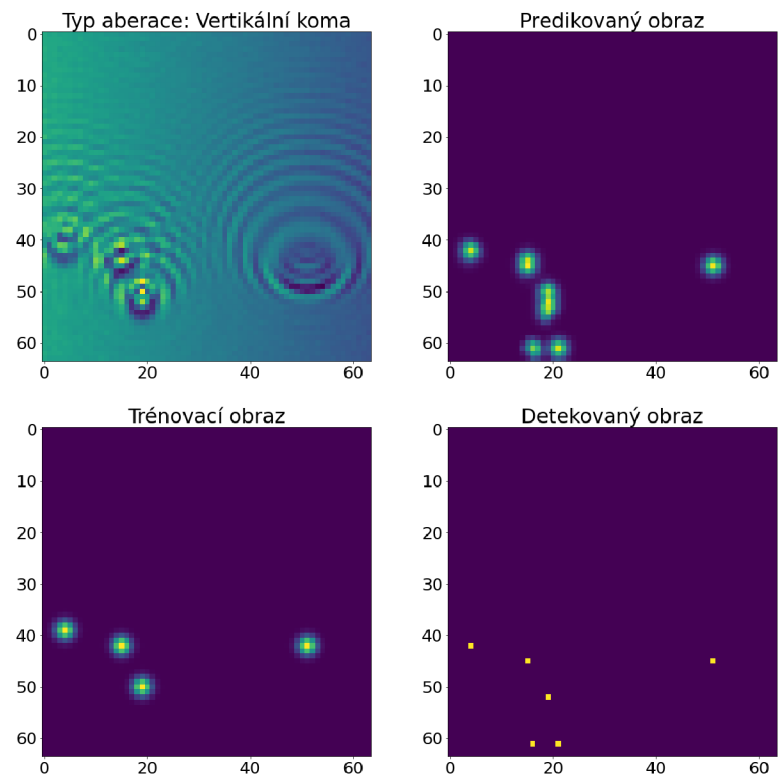
K otestování modelů byly vytvořeny soubory dat o 512 obrazech s definovaným typem aberace viz v podkap. 3.1.4 a souborem neaberovaných dat.

4.1.1 Určení 2D pozice

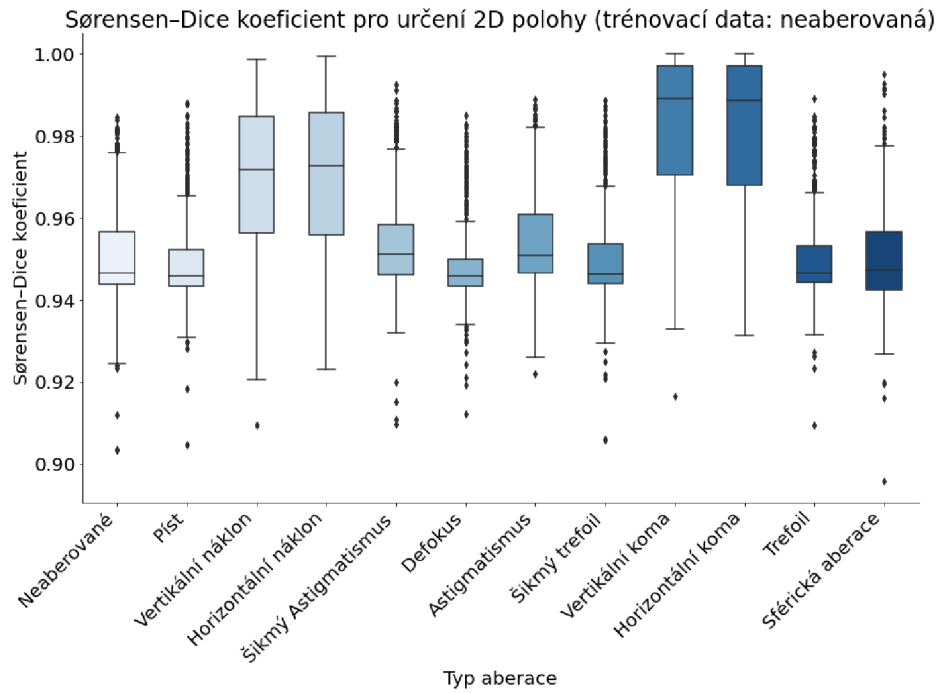
Pro určení pozice byla použita metoda hledání extrémů. Extrémy byly převedeny jako jedničkové body v nulovém prostoru. Tento prostor byl poté zvektorizován a porovnám skrze Sørensen–Dice koeficient s obdobným obrazem obsahující pravdivé hodnoty. Zároveň výsledky mohou být ovlivněny nastavení nepřiměřenou hladinou detekce vrcholů v predikovaném obraze. Na obr. 4.2 je možné vidět falešně pozitivní detekci pozice u obrazu postiženém aberací.



Obr. 4.1: Vliv typu aberace na model trénovaný na aberovaných datech pro určení 2D pozice.



Obr. 4.2: Příklad falešně pozitivní detekce pozice na postiženém obraze vertikální komou pro model trénovaný na aberacích.

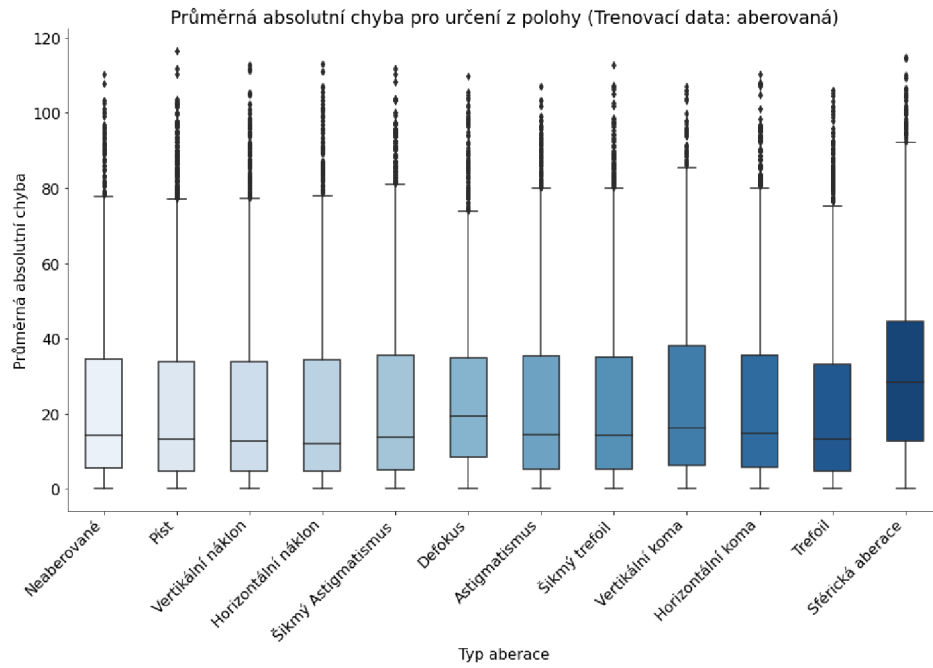


Obr. 4.3: Vliv typu aberace na model trénovaný na neaberovaných datech pro určení 2D pozice.

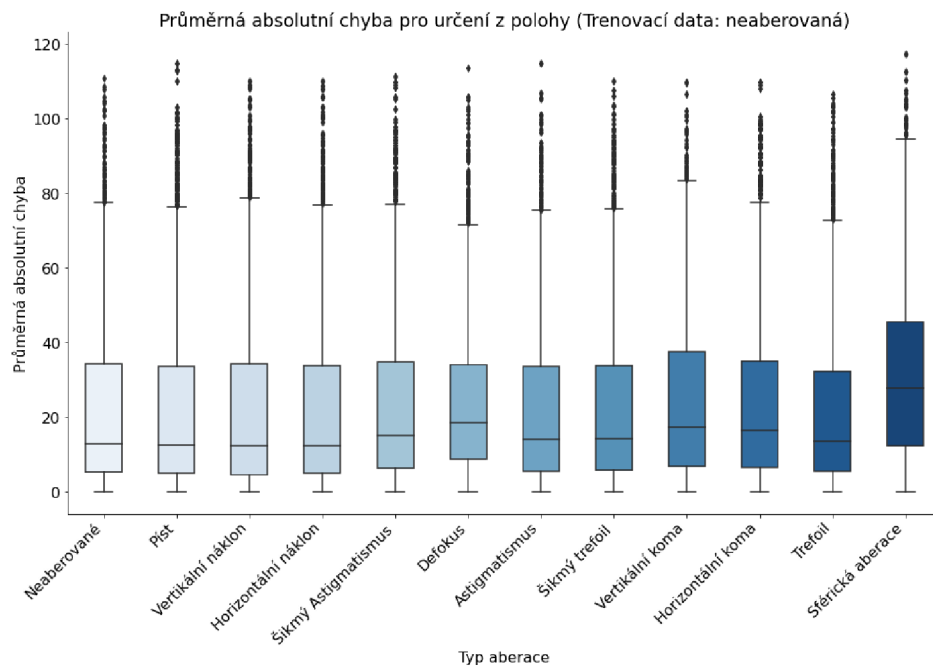
Model učený na neaberovaných datech je podle výsledků robustnější než model, který byl učený pouze na aberovaných datech. Tyto výsledky neodpovídají předpokladu. Nicméně horší model vykazuje větší robustnost na určité typy aberací, jako vertikální a horizontální koma.

4.1.2 Určení polohy z pozice

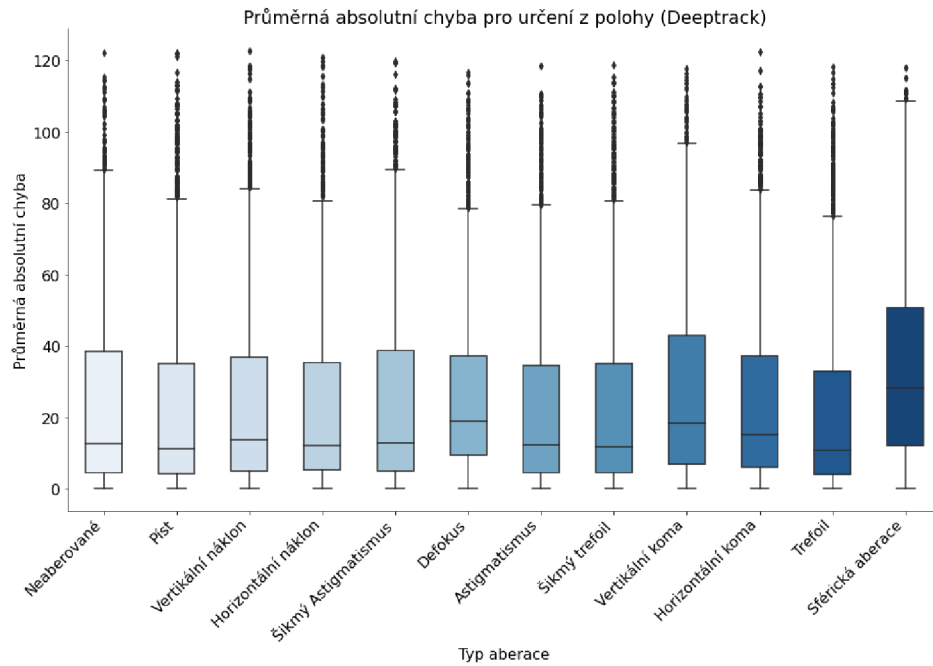
Výsledky chyby pro určení z polohy jsou normovány podle intervalu uvedeného v tab. 3.2.



Obr. 4.4: Vliv typu aberace a čistých dat na model trénovaný na aberovaných datech pro určení polohy z. Jednotky jsou uvedeny v μm .



Obr. 4.5: Vliv typu aberace a čistých dat na model trénovaný na neaberovaných datech pro určení polohy z. Jednotky jsou uvedeny v μm .



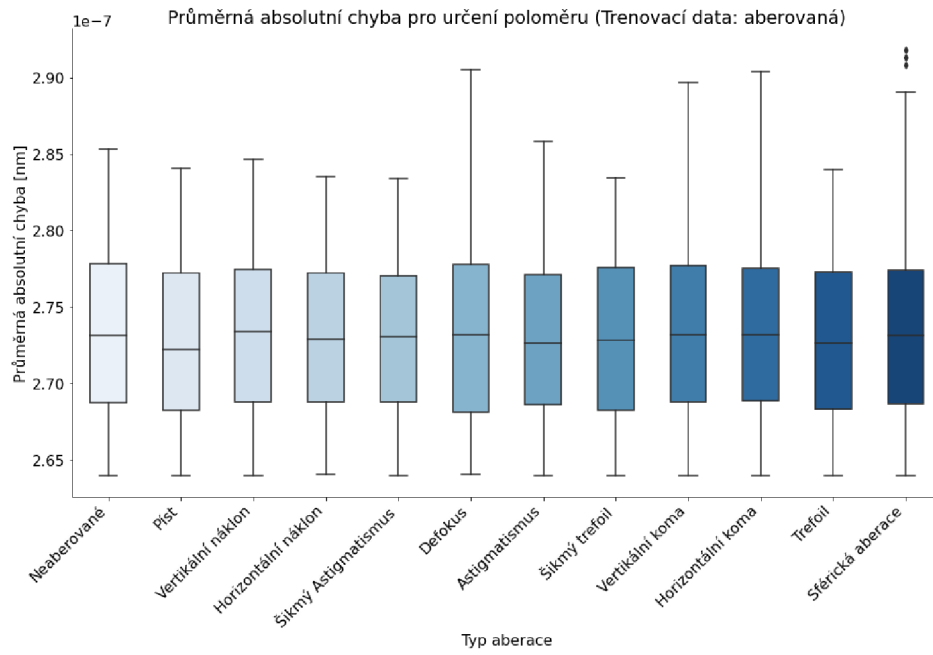
Obr. 4.6: Vliv typu aberace a čistých dat na DT model určující polohu z. Jednotky jsou uvedeny v μm .

Žádný z modelů nevykazuje uspokojivou přesnost pro určení polohy. Na vině může být velký interval hodnot, který je normovaný do intervalu od 0 do 1. Rozložením velkého intervalu vytváří malý úsek pro chybu, nutí model k získání yšší přesnosti.

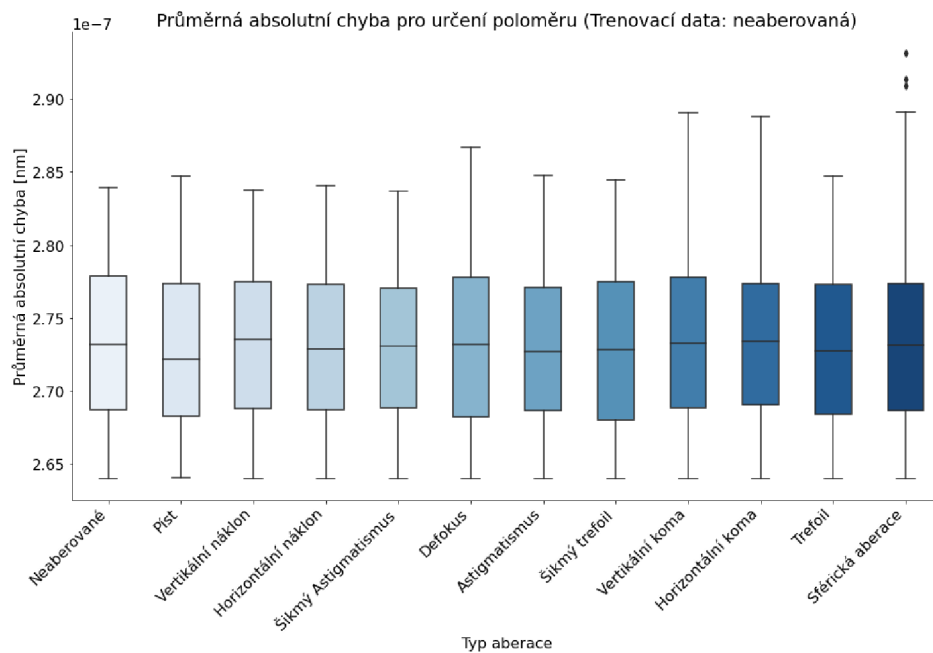
Modely učené na aberovaných a neaberovaných podobně. Pro určité typy aberací dosahoval model učený na aberacích lepší výsledky, což je předpokladatelné. Model implementovaný skrze DT dosahoval horších výsledků.

4.1.3 Určení poloměru

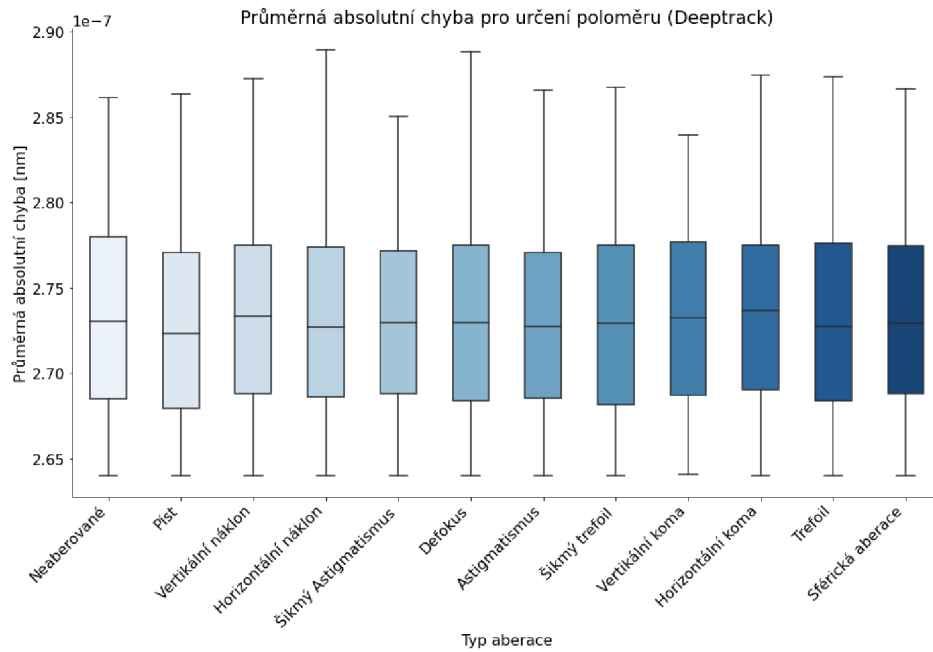
Výsledky chyby pro určení poloměru jsou normovány podle intervalu uvedeného v tab. 3.2. Pro správnou interpretaci výsledné chyby odečíst je nutné tuto hodnotu nejnižší hodnotu od výsledné chyby..



Obr. 4.7: Vliv typu aberace a čistých dat na model trénovaný na aberovaných datech pro určení poloměru.



Obr. 4.8: Vliv typu aberace a čistých dat na model trénovaný na neaberovaných datech pro určení poloměru.

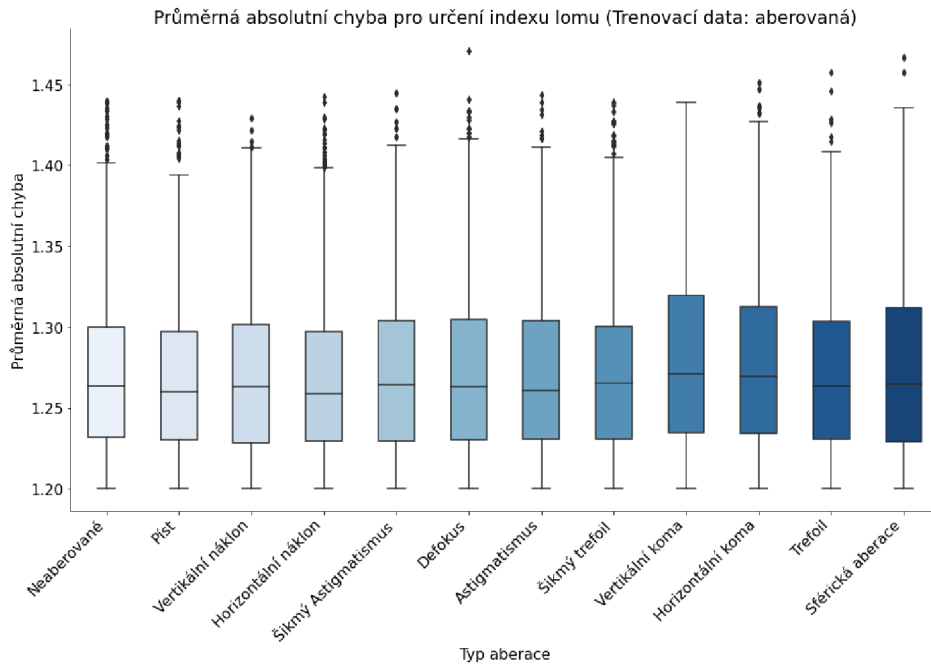


Obr. 4.9: Vliv typu aberace a čistých dat na DT model určující poloměr.

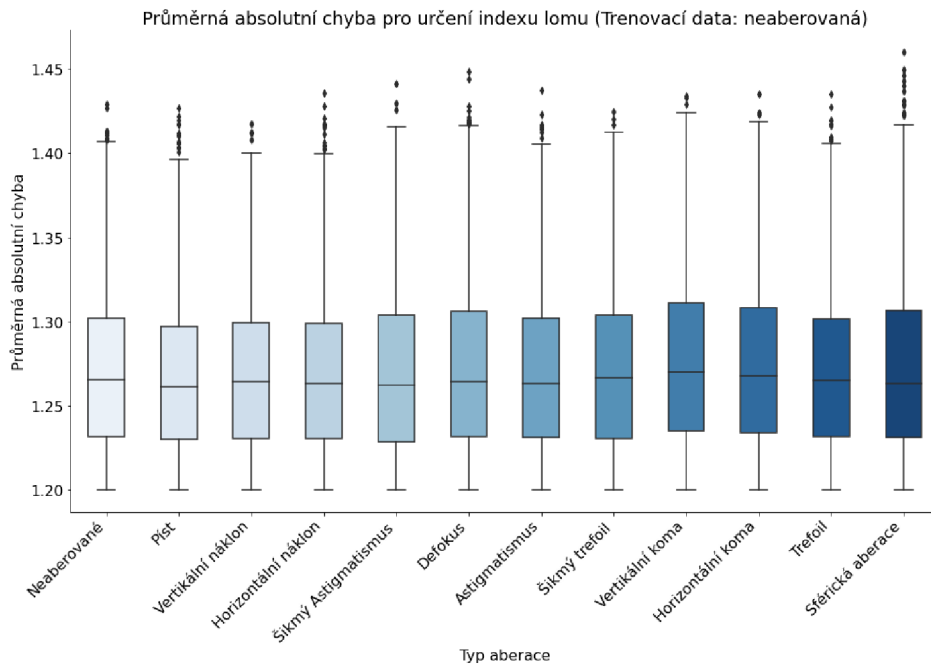
Pro určení poloměru vychází nejhůře model implementovaný v prostředí DT. Nejlepší model pro odhad parametru na čistých datech je model, který byl trénován na datech bez aberace. V Určitých typech aberací jako vertikální náklon, defokus a astigmatismus vykazuje lepší výsledky než model trénovaný na datech obsahující aberace. Nejvíce problematické je určení poloměru ze sférické aberace, defokusu, vertikální a horizontální komy.

4.1.4 Určení indexu lomu

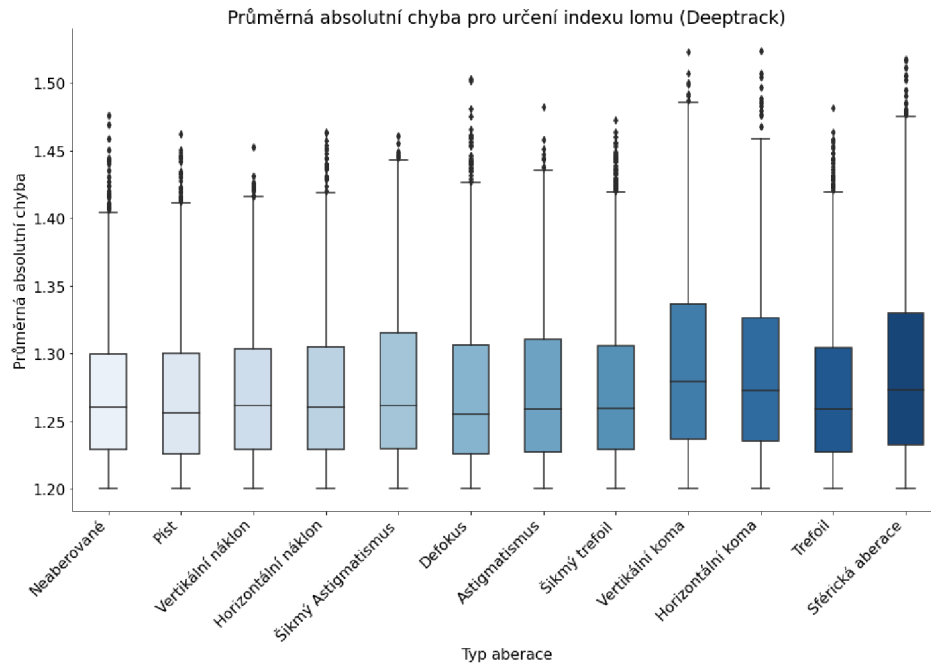
Výsledky chyby jsou normovány podle intervalu uvedeného v tab. 3.2. Z toho důvodu se nulová chyba jeví jako 1.2, pro správnou interpretaci je nutné tuto hodnotu od výsledné chyby odečíst.



Obr. 4.10: Vliv typu aberace a čistých dat na model trénovaný na aberovaných datech pro určení indexu lomu.



Obr. 4.11: Vliv typu aberace a čistých dat na model trénovaný na neaberovaných datech pro určení indexu lomu.



Obr. 4.12: Vliv typu aberace na DT model určující index lomu.

Na data nepostížené aberacemi nejlépe reagoval model navržený prostřednictvím DT, nicméně tato chyba není na tolik rozdílná od ostatních modelů. Nejhorší na neaberované data reagoval model učený na aberovaných datech. Na různé typy aberace byl nejlepší model navržený DT, poté model učený na aberovaných a datech a nakonec učený na neaberovaných datech. Aberace vykazující největší chybovost jsou sférická aberace a vertikální koma.

Celkově nejlepším model pro určení indexu lomu je model naprogramovaný skrze DT.

4.2 Aplikace modelů na experimentální data

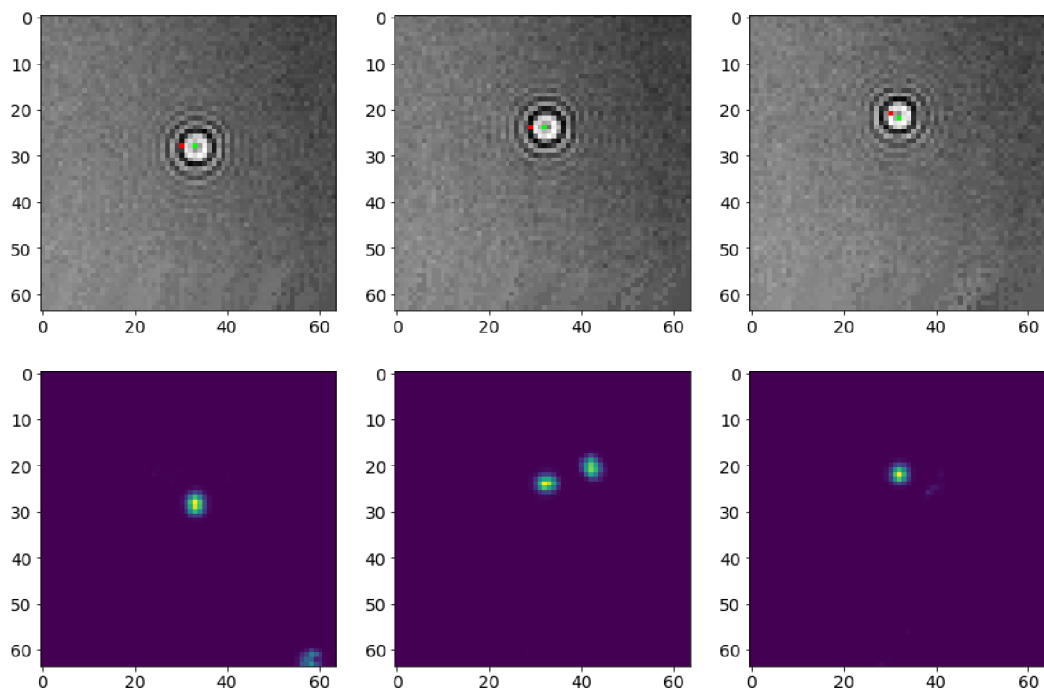
Jelikož videa byla nasnímaná v rozlišení 64x56 a modely byly učený na obraz o rozměrech 64x64, byly snímky z videa před predikcí doplněny do toho tvaru maskou 3x3 z prostředí videa. Tento přístup není nejlepší z důvodu výskytu částice na okraji videa bude informace částečně poškozena. Aby se zvýšila přesnost odhadu hodnoty je od každého snímku odečteno pozadí (viz. obr. 4.13).

Metoda používaná pro predikci polohy z , poloměru a indexu lomu na ÚTP AV ČR vychází z principu fitování křivky pomocí metody nejmenších čtverců. Princip této metody byl přiblížen v podkap. 1.4.2

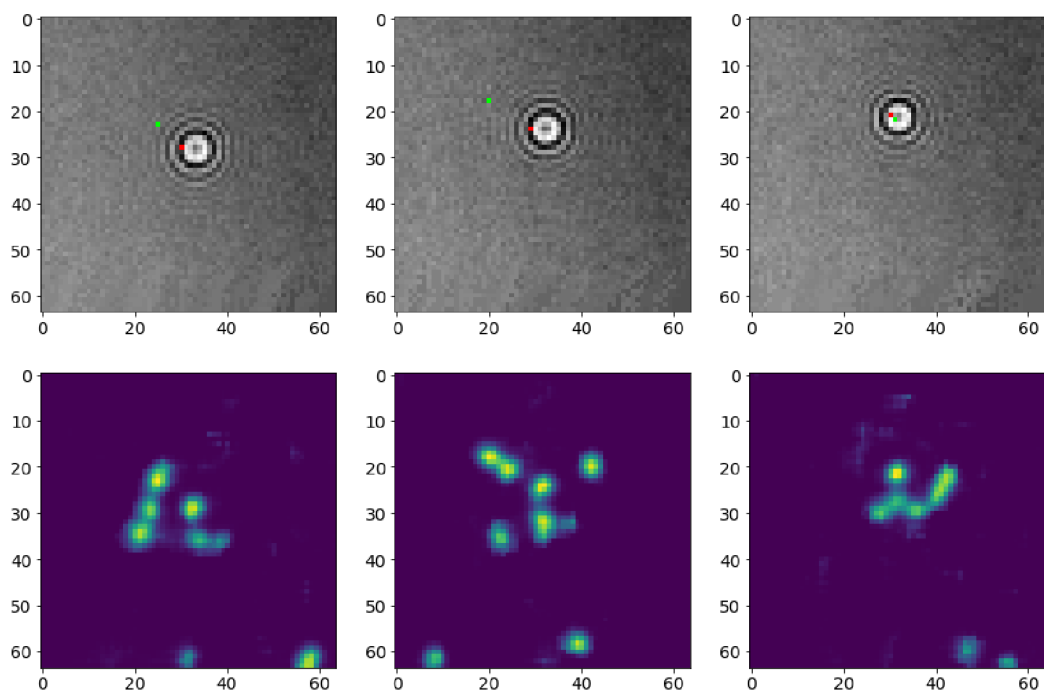


Obr. 4.13: Subtrakce nahraného pozadí bez částice od snímku s částicí.

Do predikčního času je zahrnuto u modelů PyTorch převedení obrazu na tenzor, konverze na datový typ *float*, převedení z tenzoru na pole a samotná detekce. U modelu keras změna velikosti do požadovaného tvaru a samotná predikce.



Obr. 4.14: Aplikace modelu učený na aberovaných datech pro určení 2D polohy. Zelený pixel představuje predikci model. Červený pixel představuje určení polohy pomocí metody používané na ÚTP AV ČR. Obrazy výsledné predikce jsou zobrazeny pod jednotlivými snímky z videa.



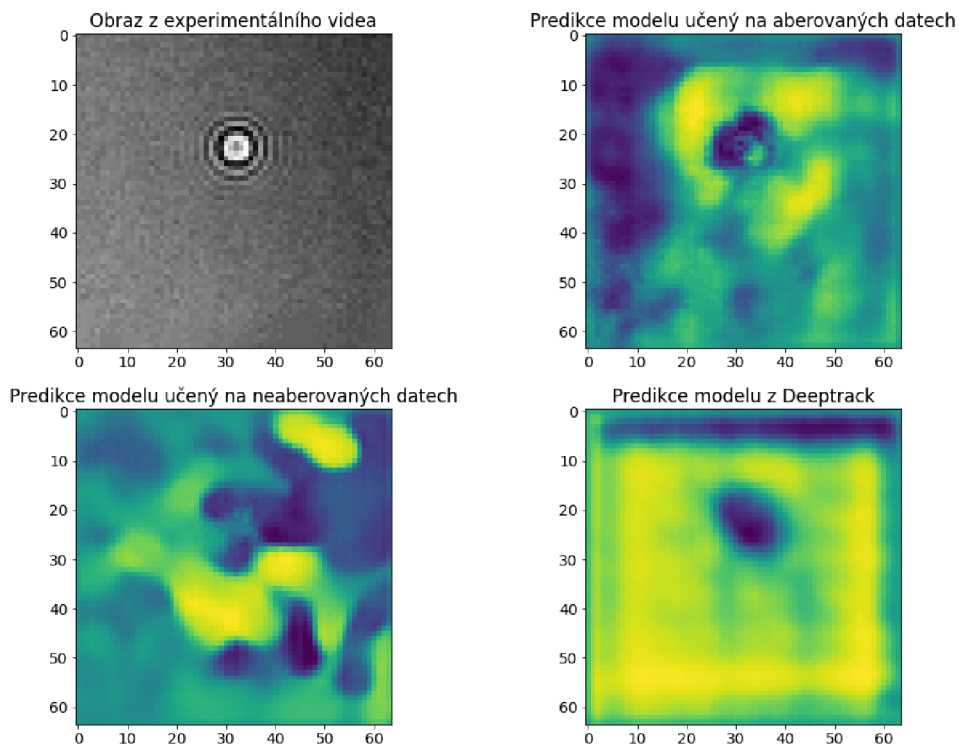
Obr. 4.15: Aplikace modelu učený na neaberovaných datech pro určení 2D polohy. Zelený pixel představuje predikci model. Červený pixel představuje určení polohy pomocí metody používané na ÚTP AV ČR. Obrazy výsledné predikce jsou zobrazeny pod jednotlivými snímky z videa.

Predikce modelu, který byl učený na aberovaných datech vykazuje daleko lepší výsledky, než pro model učený na neaberovaných a zároveň lepší výsledky než metoda používaná na ÚTP AV ČR. Z celkových výsledků pro určení 2D pozice může říct, že model s přesnější detekcí 2D polohy je vhodný pro další využití.

Tab. 4.1: Porovnání vytvořených modelů s metodou z UTP AV ČR pro určení z polohy. Hodnoty jsou uvedeny v μm .

Typ modelu	Medián	Průměr	Směrodatná odchylka.	Predikční čas
Aberované data	74.461	78.329	30.472	0.00908
Čisté data	78.766	80.077	12.249	0.01776
Deeptrack 2.1.	57.982	59.173	11.542	0.04939
Metoda ÚTP AV ČR		-13.8		

Jelikož byla vybrána experimentální data náhodně, bylo vybráno video s částicí v záporné ose z, tedy za fokusační rovinou. Pro tento případ nebyly modely trénovány, protože DT neumožňuje vytváření dat se zápornou osou z. Z tohoto důvodu je porovnání neinterpretovatelné a je zapotřebí dalších testů.

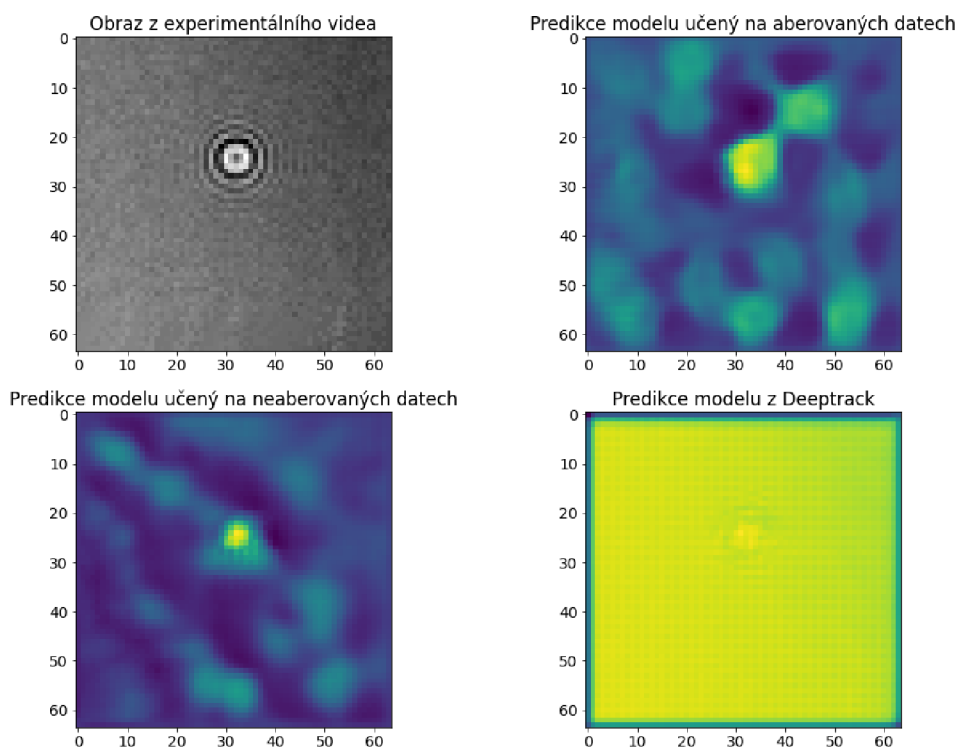


Obr. 4.16: Příklady detekce pozice z v experimentálních datech pro různé modely.

Tab. 4.2: Porovnání vytvořených modelů s metodou z ÚTP AV ČR pro určení poloměru.

Typ modelu	Medián	Průměr	Směrodatná odchylka.	Predikční čas
Aberované data	324.688	324.388	265.963	0.011129
Čisté data	323.987	324.038	266.122	0.017686
Deeprack	298.549	298.48	264.525	0.046933
Metoda ÚTP AV ČR:		329.5		

Určení poloměru je pro modely implementované v prostředí Pytorch podobné a velmi se blíží k hodnotě získané metodou používané na ÚTP AV ČR. Pokud přehlédneme fakt, že pozice v poloze z je záporná na, kterou modely nebyly trénovány, můžeme tuto metodu využít v praxi za předpokladu dalších testů.

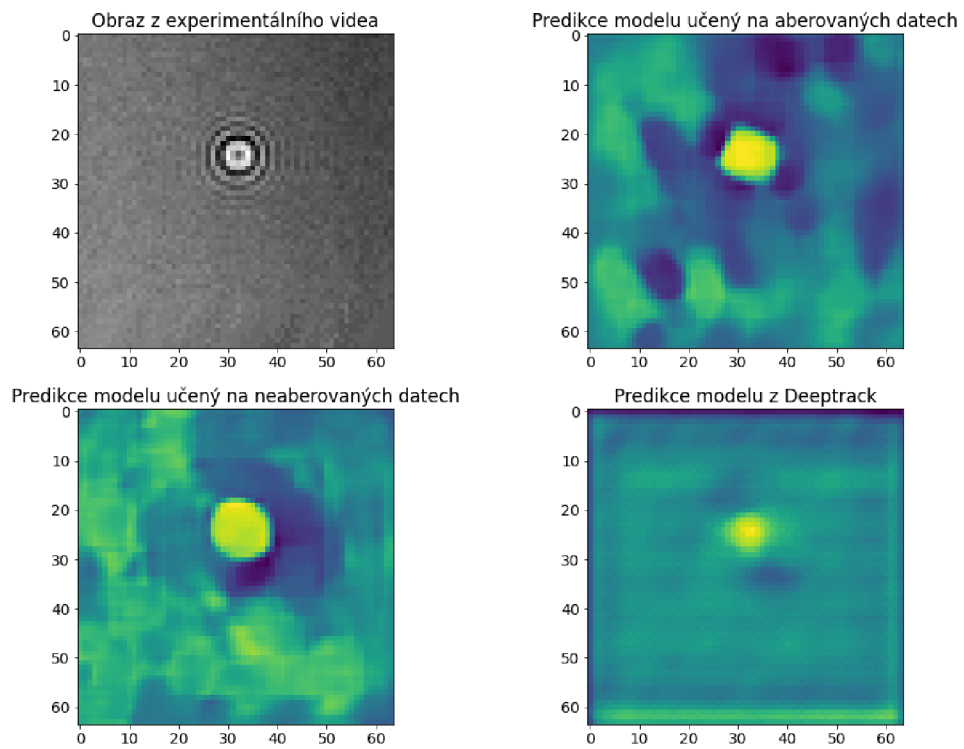


Obr. 4.17: Příklady detekce poloměru v experimentálních datech pro různé modely.

Tab. 4.3: Porovnání vytvořených modelů s metodou z ÚTP AV ČR pro určení indexu lomu.

Typ modelu	Medián	Průměr	Směrodatná odchylka.	Predikční čas
Aberované data	1.747	1.744	1.227	0.009714
Čisté data	1.706	1.704	1.222	0.034506
Deeptrack 2.1.	1.729	1.73	1.22	0.057346
Metoda ÚTP AV ČR		1.455		

Stejně, jak pro předchozí hodnoty je index lomu neinterpretovatelný, jelikož s posunutím fokusační roviny se mění i rozptyl světla, tudíž byl index lomu odhadnut na vysokou hodnotu oproti hodnotě výpočítané pomocí metody fitování použité na ÚTP AV ČR. Nicméně odhad hodnoty indexu lomu pro všechny tři modely se moc nemění, proto je potřeba další testů na experimentálních datech, jestli je možné modely využít v praxi.



Obr. 4.18: Příklady detekce indexu lomu v experimentálních datech pro různé modely.

Závěr

Tato práce se zaměřuje na popis metod strojového učení využívaných k získání kvantitativních informací ze snímků hologramu pořízených pomocí digitální holografické mikroskopie. První část je věnována stručné literární rešerši obsahující základní popis hologramu, digitální holografické mikroskopii a podobností mezi Mie teorií a hologramem. Rešerše obsahuje zároveň popis přístupů pro získání kvantitativních informací a jejich teoretického pozadí.

Z literární rešerše poté vychází metodická část. Zde jsou navrženy možné postupy pro zpracování jednotlivých parametrů nano a mikro částic. Návrhům možných řešení předchází popis použitého aplikačního rámce Deeptrack 2.1.a podkap. 3.1 popisuje možnosti využití tohoto aplikačního rámce.

Na popis Deeptrack 2.1. navazuje porovnání vytvořených dat skrze tento aplikační rámec a programovým řešením generování hologramů využívaný na ÚTP AV ČR. Dále je v metodické části popsán postup návrhu optimalizace pro vybranou architekturu U-Net implementovanou v aplikačním rámci PyTorch. Je zde popsán postup tvorby map využívaných pro segmentaci polohy a regresy hloubkových map pro určení polohy z , indexu lomu a poloměru. Následně jsou představeny vybrané konfigurace pro určení těchto parametrů. Konfigurace byly vybrány na základě výsledných chyb z optimalizačního algoritmu RayTune. Na základě vybraných konfiguračních modelů byly skrze aplikační rámec DeepTrack 2.1. naučeny modely a porovnány s modely implementované skrze PyTorch.

Porovnání modelů proběhlo na testových sadách obsahující čisté data a data s různými typy aberacemi. V závěru práce jsou uvedeny výsledky aplikace implementovaných metod a metody fitování skrze metodu nejmenších čtverců používané na ÚTP AV ČR na experimentální data.

Literatura

- [1] RAYMOND K. KOSTUK. *Holography*. CRC Press, 2019. ISBN 9780367779573. Dostupné z: doi:10.1201/9780429185830
- [2] SCHNARS, Ulf a Werner JUEPTNER. *Digital holography: digital hologram recording, numerical reconstruction, and related techniques*. Berlin: Springer, 2005, 164 s. : il. ISBN 3-540-21934-X.
- [3] HARIHARAN, P. *Basics of Holography* [online]. Cambridge University Press, 2010 [cit. 2022-04-06]. ISBN 9780521807418. Dostupné z: doi:10.1017/CBO9780511755569
- [4] MOYSES, Henrique, Bhaskar KRISHNATREYA a David GRIER. Robustness of Lorenz-Mie microscopy against defects in illumination. *Optics express*. 2013/03/11, **21**, 5968-5973. Dostupné z: doi:10.1364/OE.21.005968
- [5] MIDTVEDT, Benjamin, Saga HELGADOTTIR, Aykut ARGUN, Jesús PINEDA, Daniel MIDTVEDT a Giovanni VOLPE. Quantitative digital microscopy with deep learning. *Applied Physics Reviews*. American Institute of Physics, 2021/02/19, **8**(1), 011310. Dostupné z: doi:10.1063/5.0034891
- [6] HERGERT, Wolfram a Thomas WRIEDT, ed. *The Mie Theory* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012 [cit. 2021-12-10]. Springer Series in Optical Sciences. ISBN 978-3-642-28737-4. Dostupné z: doi:10.1007/978-3-642-28738-1
- [7] ZHANG, Shunping a Hongxing XU. *Light Scattering by Small Metallic Particles: Mie Theory and More: Mie Theory and More*. 1. 1. CRC Press, 2018, s. -54, 21 s. Dostupné z: doi:10.1201/9781315196619-2
- [8] CHEONG, Fook Chiong, Bhaskar KRISHNATREYA a David GRIER. Strategies for three-dimensional particle tracking with holographic video microscopy. *Optics express*. 2010/06/21, **18**, 13563-73. Dostupné z: doi:10.1364/OE.18.013563
- [9] GUENTHER, Robert D., Duncan G. STEEL a Leopold BAYVEL. *Encyclopedia of modern optics*. Oxford: Elsevier, 2005. ISBN 9780122276002.
- [10] BATOOL, Sidra, Mehwish NISAR, Fabio MANGINI, Fabrizio FREZZA a Eugenio FAZIO. *Scattering of Light from the Systemic Circulatory System*. 10. 2020. ISSN 2075-4418. Dostupné z: doi:10.3390/diagnostics10121026

- [11] YU, Xiao, Jisoo HONG, Changgeng LIU a Myung K. KIM. Review of digital holographic microscopy for three-dimensional profiling and tracking. *Optical Engineering*. 2014/4/1, **53**(11), 1-22. Dostupné z: doi:10.1117/1.OE.53.11.112306
- [12] GURNEY, Kevin. *An Introduction to Neural Networks*. Taylor and Francis, 1997. ISBN 9781857286731.
- [13] SCHNEIDER, B., J. DAMBRE a P. BIENSTMAN. Fast particle characterization using digital holography and neural networks. *Applied Optics*. OSA, 2016/01/01, **55**(1), 133-139. Dostupné z: doi:10.1364/AO.55.000133
- [14] YEVICK, Aaron, Mark HANNEL a David G. GRIER. Machine-learning approach to holographic particle characterization. *Optics Express*. OSA, 2014/11/03, **22**(22), 26884-26890. Dostupné z: doi:10.1364/OE.22.026884
- [15] LEE, Sang-Hyuk, Yohai ROICHMAN, Gi-Ra YI, Shin-Hyun KIM, Seung-Man YANG, Alfons van BLAADEREN, Peter van OOSTRUM a David G. GRIER. Characterizing and tracking single colloidal particles with video holographic microscopy. *Optics Express*. OSA, 2007/12/24, **15**(26), 18275-18282. Dostupné z: doi:10.1364/OE.15.018275
- [16] HANNEL, Mark D., Aidan ABDULALI, Michael O-BRIEN a David G. GRIER. Machine-learning techniques for fast and accurate feature localization in holograms of colloidal particles. *Optics Express*. OSA, 2018/06/11, **26**(12), 15221-15231. Dostupné z: doi:10.1364/OE.26.015221
- [17] LEAHY, Brian, Ronald ALEXANDER, Caroline MARTIN, Solomon BARKLEY a Vinothan N. MANOHARAN. Large depth-of-field tracking of colloidal spheres in holographic microscopy by modeling the objective lens. *Optics Express*. OSA, 2020/01/20, **28**(2), 1061-1075. Dostupné z: doi:10.1364/OE.382159
- [18] GABOR, D. A New Microscopic Principle. *Nature*. 1948, **161**(4098), 777-778. ISSN 1476-4687. Dostupné z: doi:10.1038/161777a0
- [19] SCHNARS, Ulf a Werner P O J PTNER. Digital recording and numerical reconstruction of holograms. *Measurement Science and Technology*. IOP Publishing, 2002, **13**(9), R85-R101. ISSN 0957-0233. Dostupné z: doi:10.1088/0957-0233/13/9/201
- [20] ALTMAN, Lauren E. a David G. GRIER. CATCH: Characterizing and Tracking Colloids Holographically Using Deep Neural Networks: Characterizing and Tracking Colloids Holographically Using Deep Neural Networks. *The Journal of*

- Physical Chemistry B*. American Chemical Society, 2020/03/05, **124**(9), 1602-1610. ISSN 1520-6106. Dostupné z: doi:10.1021/acs.jpcc.9b10463
- [21] SLABÝ, Tomáš, Pavel KOLMAN, Zbyněk DOSTÁL, Martin ANTOŠ, Martin LOŠŤÁK a Radim CHMELÍK. Off-axis setup taking full advantage of incoherent illumination in coherence-controlled holographic microscope. *Optics Express*. OSA, 2013/06/17, **21**(12), 14747-14762. Dostupné z: doi:10.1364/OE.21.014747
- [22] GIRKIN, John. *A Practical Guide to Optical Microscopy*. 2019/06/14. ISBN 9781315115351. Dostupné z: doi:10.1201/b22249
- [23] BRZOBOHATÝ, Oto, Lukáš CHVÁTAL a Alexandr JONÁŠ. *Holographic microscopy of levitating particles*. Brno: Institute of Scientific Instruments of the CAS, 2020.
- [24] RONNEBERGER, Olaf, Philipp FISCHER a Thomas BROX. *U-Net: Convolutional Networks for Biomedical Image Segmentation: Convolutional Networks for Biomedical Image Segmentation*. In: . Cham: Springer International Publishing, 2015, s. 234-241. ISBN 978-3-319-24574-4.
- [25] MIDTVEDT, Benjamin, Erik OLSÉN, Fredrik EKLUND, Fredrik HÖÖK, Caroline Beck ADIELS, Giovanni VOLPE a Daniel MIDTVEDT. Fast and Accurate Nanoparticle Characterization Using Deep-Learning-Enhanced Off-Axis Holography. *ACS Nano*. American Chemical Society, 2021/02/23, **15**(2), 2240-2250. ISSN 1936-0851. Dostupné z: doi:10.1021/acsnano.0c06902
- [26] EPIPHANY, JEBAMALAR LEAVLINE a Asir Antony DANASINGH. *On Teaching Digital Image Processing with MATLAB*. 2014/01/01.
- [27] GRANT-JACOB, James A., Benita S. MACKAY, James A. G. BAKER, Daniel J. HEATH, Yunhui XIE, Matthew LOXHAM, Robert W. EASON a Ben MILLS. Real-time particle pollution sensing using machine learning. *Optics Express*. OSA, 2018/10/15, **26**(21), 27237-27246. Dostupné z: doi:10.1364/OE.26.027237
- [28] SHAO, Siyao, Kevin MALLERY, S. Santosh KUMAR a Jiarong HONG. Machine learning holography for 3D particle field imaging. *Optics Express*. OSA, 2020/02/03, **28**(3), 2987-2999. Dostupné z: doi:10.1364/OE.379480
- [29] HELGADOTTIR, Saga, Aykut ARGUN a Giovanni VOLPE. Digital video microscopy enhanced by deep learning. *Optica*. OSA, 2019/04/20, **6**(4), 506-513. Dostupné z: doi:10.1364/OPTICA.6.000506

- [30] MANNING, C D, P RAGHAVAN a H SCHÜTZE. *Introduction to Information Retrieval*. Cambridge University Press, 2008. ISBN 9781139472104. Dostupné také z: <https://books.google.cz/books?id=t1PoSh4uwVcC>
- [31] WANG, Zhou, A. C. BOVIK, H. R. SHEIKH a E. P. SIMONCELLI. Image quality assessment: from error visibility to structural similarity: from error visibility to structural similarity. *IEEE Transactions on Image Processing*. 2004, **13**(4), 600-612. ISSN 1941-0042. Dostupné z: doi:10.1109/TIP.2003.819861
- [32] GOODFELLOW, Ian, Yoshua BENGIO a Aaron COURVILLE. *Deep Learning*. MIT Press, 2016.
- [33] LIAW, Richard, Eric LIANG, Robert NISHIHARA, Philipp MORITZ, Joseph GONZALEZ a Ion STOICA. *Tune: A Research Platform for Distributed Model Selection and Training: A Research Platform for Distributed Model Selection and Training*. 2018/07/13.
- [34] PASZKE, Adam, Sam GROSS, Francisco MASSA, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems 32*. Curran Associates, 2019, s. 8024-8035. Dostupné také z: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-p>
- [35] VIRTANEN, Pauli, Ralf GOMMERS, Travis E. OLIPHANT, et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*. 2020, **17**, 261-272. Dostupné z: doi:10.1038/s41592-019-0686-2
- [36] VAN DER WALT, Stéfan, Johannes L SCHÖNBERGER, Juan NUNEZ-IGLESIAS, et al. Scikit-image: image processing in Python: image processing in Python. *PeerJ*. 2014, **2**, e453. ISSN 2167-8359. Dostupné z: doi:10.7717/peerj.453
- [37] DEVELOPERS, TensorFlow. *TensorFlow*. Zenodo, 2022. Dostupné z: doi:10.5281/zenodo.6519082
- [38] KINGMA, Diederik a Jimmy BA. Adam: A Method for Stochastic Optimization: A Method for Stochastic Optimization. *International Conference on Learning Representations*. 2014/12/22.
- [39] JADON, Shruti. *A survey of loss functions for semantic segmentation*. In: IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB). Dostupné z: doi:10.1109/cibcb48159.2020.9277638

- [40] LI, L., K. JAMIESON, Giulia DESALVO, A. ROSTAMIZADEH a A. TALWALKAR. Hyperband: A novel bandit-based approach to hyperparameter optimization: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*. 2018/04/01, **18**, 1-52.
- [41] YU, Tong a Hong ZHU. *Hyper-Parameter Optimization: A Review of Algorithms and Applications: A Review of Algorithms and Applications*. arXiv, 2020. Dostupné z: doi:10.48550/ARXIV.2003.05689