



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Sběr a analýza hyperspektrálních dat s využitím komprimovaného snímání

Bakalářská práce

Studijní program: B3901 – Aplikované vědy v inženýrství

Studijní obor: 3901R055 – Aplikované vědy v inženýrství

Autor práce: **Jakub Lukeš**

Vedoucí práce: Ing. Jiří Hlubuček





Zadání bakalářské práce

Sběr a analýza hyperspektrálních dat s využitím komprimovaného snímání

Jméno a příjmení: **Jakub Lukeš**
Osobní číslo: M17000101
Studijní program: B3901 Aplikované vědy v inženýrství
Studijní obor: Aplikované vědy v inženýrství
Zadávací katedra: Ústav nových technologií a aplikované informatiky
Akademický rok: **2019/2020**

Zásady pro vypracování:

1. Seznamte se s technikami hyperspektrálního zobrazování a komprimovaného snímání. Zaměřte se na metodu CASSI (coded aperture snapshot spectral imaging).
2. Provedte pokusná měření metodou CASSI a zkoumejte limity této metody z hlediska vlastností zobrazovaných scén.
3. Pomocí programu Matlab porovnejte různé přístupy zpětné rekonstrukce původního signálu.
4. Navrhněte doporučenou konfiguraci zobrazovacího systému a rekonstrukčních parametrů.
5. Vypracujte podrobnou dokumentaci a analýzu provedených měření, včetně navržených postupů kalibrace a justáže.

Rozsah grafických prací:
Rozsah pracovní zprávy:
Forma zpracování práce:
Jazyk práce:

dle potřeby
30-40 stran
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] Stern, A., Optical compressive imaging. CRC Press, 2016.
- [2] Šonka, M., Hlaváč V., Boyle. R.: Image processing, analysis, and machine vision. 3rd ed. Toronto: Thomson, 829 s. ISBN 978-0-495-08252-1, 2008.
- [3] Gamez, G., Compressed sensing in spectroscopy for chemical analysis. J. Anal. At. Spectrom., 2016, 31, 2165.
- [4] Candes, E.J., WAKIN, M.B., An Introduction to Compressive Sampling. IEEE Signal Processing Magazine. 2008, 25(2), 21-30.

Vedoucí práce:

Ing. Jiří Hlubuček
Ústav nových technologií a aplikované informatiky

Konzultant práce:

RNDr. Karel Žídek, Ph.D.
TOPTEC, ÚFP Akademie Věd ČR, v.v.i.

Datum zadání práce:

9. října 2019

Předpokládaný termín odevzdání:

18. května 2020

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

Ing. Josef Novák, Ph.D.
vedoucí ústavu

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že texty tištěné verze práce a elektronické verze práce vložené do IS STAG se shodují.

1. 6. 2020

Jakub Lukeš

Sběr a analýza hyperspektrálních dat s využitím komprimovaného snímání

Abstrakt

Tato bakalářská práce se zabývá komprimovaným snímáním a jeho uplatněním při sběru hyperspektrálních dat. Cílem práce bylo seznámit se s metodou CASSI (Coded Aperture Snapshot Spectral Imaging), která umožňuje záznam hyperspektrálních dat pomocí jednoho snímku, provést pokusná měření i s následnými rekonstrukcemi obrazu a poté pro danou sestavu navrhnout doporučenou konfiguraci rekonstrukčních parametrů. Měření byla prováděna na soustavě HDES-VIS ve výzkumném centru TOPTEC, která pracuje v rozmezí vlnových délek od 400 nm do 900 nm (přibližně viditelné spektrum). Pro porovnávání různých přístupů rekonstrukce byly z naměřených dat vybrány tři typově odlišné scény, kde ke každé scéně byla následně vytvořena simulovaná data. Nejprve byla prováděna rekonstrukce pouze na simulovaných datech, díky čemuž byla odzkoušena funkčnost systému a mohl se snadno porovnávat vliv jednotlivých parametrů na proces rekonstrukce. Poté proběhla rekonstrukce i za použití reálných dat a ze získaných poznatků byla navržena ideální konfigurace jak pro jednotlivé typy scén, tak pro celou soustavu obecně.

Klíčová slova: hyperspektrální snímání, komprimované snímání, metoda CASSI, HDES-VIS

Acquisition and analysis of hyperspectral data by using compressed sensing

Abstract

This bachelor thesis focuses on the compressed sensing and its usage in acquisition of hyperspectral data. The aim of this thesis was to get acquainted with the CASSI method (Coded Aperture Snapshot Spectral Imaging), which allows to record hyperspectral data using a single snapshot image, perform measurements with the image reconstruction and eventually suggest ideal configuration of reconstruction parameters. Measurements were performed with the system called HDES-VIS in the research centre TOPTEC, which operates in the spectrum range from 400 nm to 900 nm (approximately visible light). To compare different reconstruction approaches, three different types of scenes were selected from the measured data and for every type of scene simulated data were also created. At the beginning the reconstruction was performed only with simulated data, thanks to which the functionality of whole system could be tested and the effect of individual parameters on reconstruction could be easily compared. Subsequently, reconstruction with the real data was performed and from acquired knowledge the ideal configuration for every type of scene and also for whole system was suggested.

Keywords: hyperspectral sensing, compressed sensing, CASSI method, HDES-VIS

Poděkování

Tímto bych rád poděkoval všem, kteří mi s touto prací pomohli. Především pak děkuji svému vedoucímu Ing. Jiřímu Hlubučkovi a konzultantovi RNDr. Karlu Žídkovi, Ph.D. za jejich neocenitelné odborné rady, trpělivost, vstřícnost a celkovou pomoc při vytváření této práce.

Obsah

Seznam obrázků	8
Seznam tabulek	9
Seznam zkratk	10
1 Úvod	11
2 Rešeršní část	12
2.1 Hyperspektrální snímání	12
2.1.1 Využití hyperspektrálních dat	13
2.2 Digitální snímání obrazu	13
2.3 Komprimované snímání	15
2.3.1 Obecný popis	15
2.3.2 Základní matematický popis	17
2.4 Metoda CASSI	18
2.4.1 Rekonstrukce obrazu	20
2.4.2 Haarova transformace	21
2.5 Jednopixelová kamera	22
2.6 CMOS komprimované snímání	23
3 Hyperspektrální detekční systém	25
3.1 Základní popis	25
3.1.1 Optická soustava	25
3.1.2 Detektor	26
3.1.3 Modulační maska	27
3.1.4 Difrakční mřížka	28
3.1.5 Zpracování vstupních dat	28
3.2 Rekonstrukce obrazu	29
3.2.1 Počáteční odhad	30
3.2.2 Převod detektoru na datakrychli	31
3.2.3 Převod datakrychle na detektor	32
3.3 Justáž a kalibrace	35
4 Měření	37
4.1 Intenzita prvního a nultého řádu	37
4.2 Testovací scény	37
4.2.1 Problémy ve zobrazování scény	39

4.3	Rekonstrukce pomocí jednotlivých módů	41
4.3.1	Mód 1	43
4.3.2	Mód 2	44
4.3.3	Mód 3	47
4.3.4	Shrnutí	52
4.4	Další pokusná měření	53
5	Závěr	55
	Zdroje	57

Seznam obrázků

2.1	Fourierova transformace	14
2.2	Komprimované snímání - příklad	16
2.3	Soustava pro snímání pomocí CASSI	19
2.4	Průběh snímání pomocí CASSI	20
2.5	Haarova vlnka	22
2.6	Jednopixelová kamera	23
2.7	Soustava pro CMOS komprimované snímání	24
3.1	Disperzní část optické soustavy	26
3.2	Vstupní data z detektoru	27
3.3	Modulační maska	28
3.4	Funkce pro vytvoření počátečního odhadu	32
3.5	Funkce pro převod dat detektoru na datakrychli	33
3.6	Funkce pro převod datakrychle na detektor	34
3.7	Sestava HDES	35
4.1	Relativní intenzita obrazu na detektoru	38
4.2	Poměr intenzity mezi prvním a nultým řádem	38
4.3	Spektrum lampy	39
4.4	Vstupní data z detektorů	40
4.5	Porovnání simulovaných ideálních a rozmazaných dat	42
4.6	Simulovaná scéna s barevnými filtry osvětlená lampou	45
4.7	Scéna s křížem osvětlená lampou (mód 2)	47
4.8	Spektrum scény s filtry osvětlené lampou (mód 2)	48
4.9	Simulovaná scéna s křížem osvětlená lampou (mód 3)	49
4.10	Spektrum simulované scény s křížem osvětlené lampou (mód 3)	50
4.11	Scéna s křížem osvětlená laserem (mód 3)	51
4.12	Scéna s křížem osvětlená lampou (mód 3)	51
4.13	Rekonstrukce scény osvětlené laserem a diodou	54
4.14	Spektrum zářivky	54

Seznam tabulek

3.1	Seznam optických elementů v disperzní části	26
3.2	Přístup ke zpracování snímků	30
4.1	Podobnost datakrychlí	52
4.2	Přehled scén s parametry rekonstrukce	53

Seznam zkratek

CASSI	Coded Aperture Snapshot Spectral Imaging - metoda pro komprimované snímání
MB	Megabajt
GB	Gigabajt
RGB	Red-Green-Blue - typ míchání barev
JPEG	Metoda ztrátové komprese obrazu
LASSO	Least absolute shrinkage and selection operator - typ metody pro regresní analýzu
TwIST	Two-step Iterative Shrinkage/Thresholding Algorithm - typ algoritmu pro řešení optimalizační úlohy
DMD	Digital Micromirror Device - pole mikrozrcátek
CMOS	Complementary Metal-Oxide-Semiconductor - typ snímače
HDES-VIS	Hyperspektrální detekční systém viditelného spektra
Nd	Neodym
YAG	Yttrito-hlinitý granát
FO	First order - první difrakční řád
ZO	Zero order - nultý difrakční řád

1 Úvod

Každý objekt (látka) na Zemi vyzařuje, odráží či propouští záření na různých vlnových délkách. Tohoto jevu se v začalo hojně využívat v mnoha oborech od rozpoznávání hornin, přes využití v medicíně, až po vojenské technologie. Jedním ze způsobů jak tento jev zaznamenat a zobrazit je takzvané hyperspektrální snímání. Při hyperspektrálním snímání zachycujeme obraz na různých vlnových délkách. Ve výsledku tedy máme velké množství snímků, kde každý snímek zachycuje stejnou scénérii, avšak na jiné vlnové délce. Z těchto snímků pak vytváříme takzvanou hyperspektrální datakrychli, kde máme za sebou v ose z řazené jednotlivé snímky. Vystává zde tedy zásadní problém, kterým je velikost těchto dat. Jedna takováto datakrychle může mít velikost klidně i desítky až stovky MB, což by ještě nebyl žádný velký problém, kdybychom chtěli snímat jen jednu takovou datakrychli. Problém však nastává, když chceme takovýchto dat snímat více nebo dokonce požadujeme, abychom snímali živý záznam (či záznam s minimální odezvou). V takovémto případě je nutné využívat výkonné počítače, které jsou schopny toto velké množství dat zpracovat a poté přes různé algoritmy zmenšit.

Nicméně je zde i jiná možnost, kterou představuje komprimované snímání obrazu, pomocí kterého můžeme data zmenšit již při jejich snímání a kterým se bude zabývat tato bakalářská práce.

V rešeršní části této práce se tedy budu věnovat komprimovanému snímání a jeho aplikaci v hyperspektrálním sběru dat. Uvedu zde několik metod, kterými je možné toto komprimované snímání provádět, nicméně hlavní část práce bude zaměřena na metodu CASSI (Coded Aperture Snapshot Spectral Imaging). Následně zde popíšu systém HDES-VIS, což je soustava pracující na principu metody CASSI, pomocí které jsme prováděl všechna měření. V experimentální části pak provedu vyhodnocení výsledků pro jednotlivá měření a navrhnu vhodnou konfiguraci rekonstrukčních parametrů pro tuto soustavu.

2 Rešeršní část

2.1 Hyperspektrální snímání

Jak již bylo nastíněno v úvodu, hyperspektrální snímání v sobě kombinuje prvky spektroskopie a běžného snímání obrazu, díky čemuž můžeme najednou získat jak spektrální informaci, tak i informaci prostorovou. Výstupem takového snímání bývají běžně takzvané hyperspektrální datakrychle, kde za sebou máme v ose z seřazeny snímky jednotlivých vlnových délek. U běžně používaného hyperspektrálního snímání se většinou bavíme o spektrech viditelného záření až po záření infračervené (vlnová délka v rozmezí přibližně 400 nm až 1 mm).[1]

Níže jsou popsány tři základní principy jak můžeme snímat hyperspektrální data.

1. **Point mapping** - Scéna je při této metodě scanována po jednotlivých pixelech. Touto metodou jsme schopni docílit velmi dobrého rozlišení jak spektrálního, tak i prostorového a princip této metody se využívá kupříkladu k takzvanému whisk-broom scanu. Tento typ metody snímání si našel uplatnění například u satelitů snímajících povrch planety, kde je v takovém satelitu umístěn detektor a soustava pohyblivých zrcadel. Tyto zrcadla se naklápí a posílají tak paprsky odraženého světla na detektor, který tak postupně snímá jednotlivé pixely v linii kolmé na směr letu satelitu. Hlavní nevýhodou point mappingu je poměrně dlouhá doba scanu a konkrétně u whisk-broom scanu také náchylnost mechanických částí ovládajících pohyb zrcadel k poškození. [1, 2, 3]
2. **Line Scan** - Scéna je při této metodě scanována po jednotlivých liniích skládajících se z většího množství pixelů. Tato metoda slouží například k takzvanému push-broom scanu. Mimo jiné si tato metoda (obdobně jako whisk-broom scan) našla uplatnění ve vesmírných snímkovacích satelitech, kde je v takovém satelitu namísto jednoho detektoru, snímajícího jeden pixel, umístěn detektor schopný snímat linii pixelů. Nejsou zde tedy žádná pohyblivá zrcadla a jak satelit letí, v každý časový okamžik snímá najednou větší množství pixelů, které jsou v jedné linii kolmé na směr letu. Díky Line scanu můžeme získávat o něco silnější signál na výstupu než u metody Point mappingu a to z toho důvodu, že každý pixel snímáme delší dobu. Nicméně nevýhodou je, že všechny pixely musí být stejně zkalibrované, jinak mohou mít jinou citlivost, což se poté projeví negativně na výstupu a stejně jako u Point mappingu, takovýto scan trvá poměrně dlouho.[1, 2, 3]
3. **Plane scanning** - Této metodě se také někdy říká „staring method“. Její

součástí bývá proměnlivý spektrální filtr, přes který zobrazujeme scénu na 2D detektor, díky čemuž zaznamenáme v jednom okamžiku celou scénu na příslušné vlnové délce. Na rozdíl od předešlých dvou metod, kde musíme snímat scénu po řádcích, či dokonce po jednotlivých pixelech, zde nám na zachycení celé scény tedy stačí jeden snímek, díky čemuž je snímání mnohem rychlejší. Zároveň však jasnou nevýhodou je cena samotného detektoru, protože potřebujeme najednou snímat mnohem větší oblast než u předešlých dvou metod. Taktéž může být problém s kalibrací, neboť se nám téměř nikdy nepodaří zkalibrovat všechny pixely stejně.[1, 2]

2.1.1 Využití hyperspektrálních dat

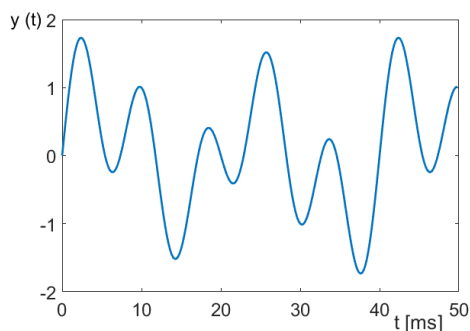
Hyperspektrální snímkování se dnes používá v mnoha různých oborech od kontroly kvality jídla [4], přes detekci rakoviny [5] až po vojenské a špiónážní aplikace [6]. Jedna z významných aplikací, která se využívá především v chemii a medicíně a s ní spojených oborech, je takzvaná infračervená spektroskopie. Tato metoda využívá toho, že každá molekula absorbuje či emituje (většinou se však zabýváme pouze absorpcí) záření na určitých vlnových délkách. Tyto vlnové délky jsou pevně dané a dají se experimentálně zjistit, neboť každý molekulární systém má díky principům kvantové mechaniky předem dané dovolené energetické hladiny, které se liší v závislosti na uspořádání systému a na jeho prvcích. Tyto energetické hladiny pak závisí na potenciální energii elektronů v systému a na molekulárních rotacích a vibracích. U infračervené spektroskopie nás však zajímají pouze molekulární rotace a vibrace, neboť se zaměřujeme na infračervené spektrum, které nemá obecně dostatečnou energii k excitaci elektronů na vyšší energetickou hladinu a může tedy ovlivnit pouze rotační a vibrační spektrum. Díky tomu tedy můžeme na hyperspektrálním snímku rozpoznat, na jakou látku se právě díváme podle toho, jaké části (v našem případě infračerveného) spektra absorbuje.[1, 7, 8]

Tuto aplikaci zde zmiňuji především proto, že je částečně spjat se soustavou, s kterou jsem pracoval a kterou zde budu popisovat. Tato soustava sice pracuje s viditelným zářením, nicméně jejím hlavním cílem je otestovat systém právě ve viditelném spektru. Díky získaným zkušenostem bude pak možné dále navrhnout nový systém, který bude pracovat s vlnovými délkami v rozmezí od infračerveného záření středních délek až po infračervené záření dlouhých vlnových délek (tudíž rozmezí okolo 5 μm až 1000 μm), což je obdobné rozmezí, ve kterém pracuje většina přístrojů využívající infračervenou spektroskopii [1].

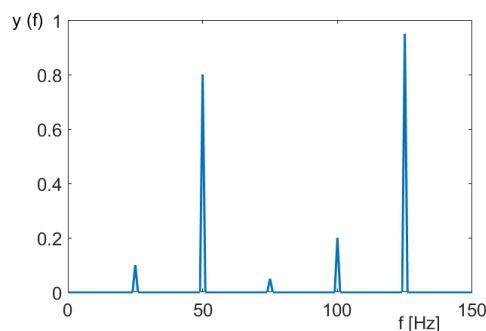
2.2 Digitální snímání obrazu

Pro pochopení motivace využití komprimovaného snímání zde nejprve nastíním, jak funguje běžné snímání obrazu, které využívají klasické fotoaparáty.

První věc, kterou si musíme uvědomit, je, že snímek je signál, který se dá popsat nějakou spojitou funkcí. Tato funkce může být buď skalární (například pro popis monochromatického snímku) nebo vektorová (pro popis obrázku složeného z více



(a) Signál v časové doméně



(b) Signál ve frekvenční doméně

Obrázek 2.1: Ukázka signálu v časové doméně převedeného fourierovou transformací do frekvenční domény

barev - běžně například RGB), se dvěma proměnnými ($f_{(x,y)}$ pro určení polohy v ploše) nebo třemi proměnnými ($f_{(x,y,z)}$), kde přibývá buď čas, nebo, jak bylo zmíněno u hyperspektrálního snímání, vlnová délka. Po zmáčknutí spouště fotoaparátu je tedy scéna pomocí detektoru navzorkována na matici $M \times N$ a následně je provedena kvantizace (spojitý průběh funkce f je rozdělen na K kroků a každému pixelu v matici pak je díky tomu možné přidělit odpovídající hodnotu, představující většinou intenzitu dopadajícího světla).[9]

To však ještě není konec celého procesu, neboť takto získaná data mohou mít u běžného fotoaparátu velikost až do desítek MB, což je většinou nežádoucí a je nutné jejich velikost nějakým způsobem zredukovat. Jednou z možností je využít Fourierovu (případně kosínovou nebo vlnkovou) transformaci, která nám signál převede z časové domény (respektive u obrazových signálů z domény prostorové) na doménu frekvenční. Díky tomu pak můžeme vidět, jaké všechny frekvence jsou v signálu zastoupeny a případně určit ty, které nemají na celkový průběh zásadní vliv. Jak můžeme vidět na obrázku 2.1, poměrně složitý průběh funkce (signálu) byl pomocí Fourierovy transformace převeden na zobrazení ve frekvenční doméně. V této doméně je tento signál takzvaně řídký, což znamená, že celý jeho průběh je reprezentován jen malým počtem nenulových hodnot (v tomto případě pěti různými frekvencemi). Z obrázku 2.1b taktéž můžeme určit, jak velký vliv mají jednotlivé frekvence na celkový průběh signálu. V tomto případě vidíme, že frekvence 50 Hz a 125 Hz jsou nejvýznamnější a teoreticky by tedy stačilo ponechat pouze tyto dvě frekvence a průběh signálu by se téměř nezměnil. Na obdobném principu pracuje například metoda komprese JPEG, při které pomocí kosínové transformace odstraníme přes 90% dat, avšak na výsledný snímek to nemá téměř žádný vliv.[1, 9]

Takovéto „plýtvání“ daty není v běžných aplikacích problémem, zpracování snímků podobnými metodami je velmi rychlé a pro běžného uživatele nepostřehnutelné. Problém však vyvstává, když se snažíme takovýchto snímků dělat velké množství v krátkém časovém intervalu (např. vysokorychlostní snímání) nebo se pokoušíme v přiměřeném čase s cenově dostupnou technikou vytvářet kupříkladu datakrychle z hyperspektrálního snímání. V podobných případech je tato neefektivita pořizování snímků již značný problém a musíme tedy přistoupit na jinou metodu, jak snímky za-

znamenávat. Touto metodou je právě komprimované snímání, při kterém se snažíme zaznamenávat jen podstatné části signálu, abychom nemuseli zbytečně odstraňovat velké množství nepotřebných dat a urychlili tak celý proces.[1]

2.3 Komprimované snímání

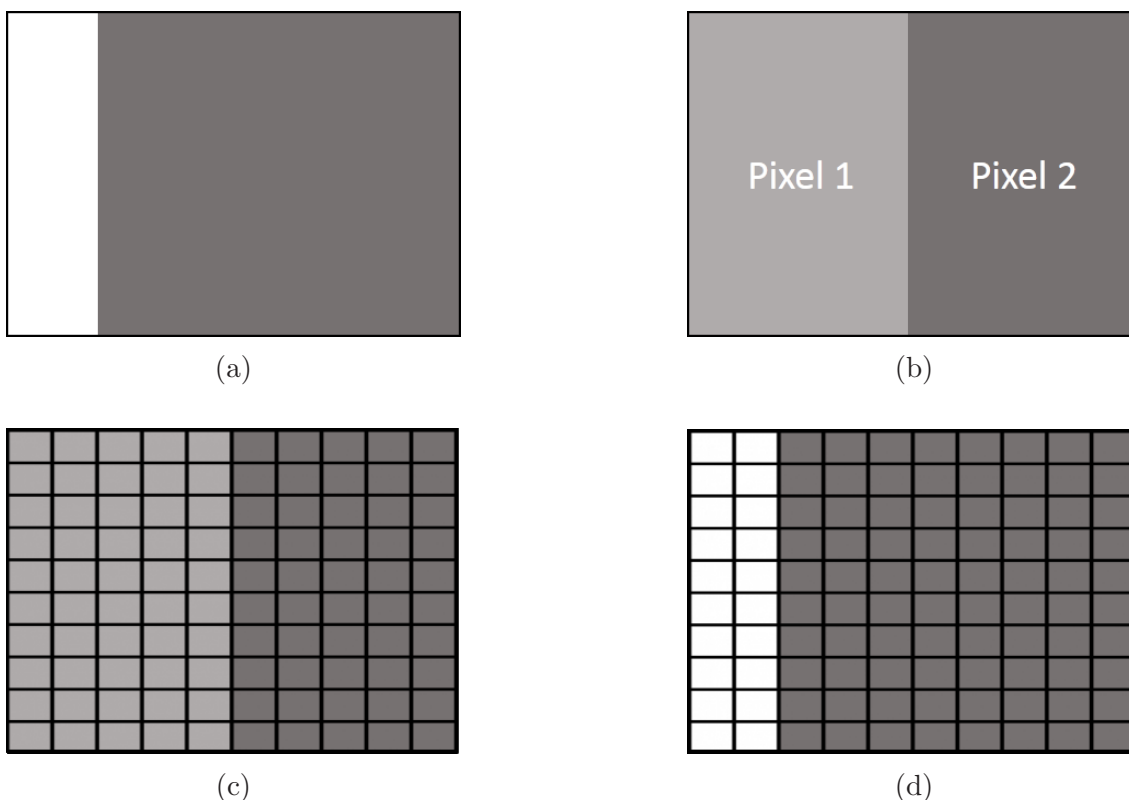
2.3.1 Obecný popis

Obecně se dá o téměř jakémkoliv snímání říct, že je do jisté míry komprimované. Důvod je prostý, v podstatě jakýkoliv fyzikální proces, který chceme snímat, má ve skutečnosti spojitý průběh (ať už se jedná o měření elektromagnetických vln nebo třeba měření teploty a tak podobně), avšak samotné měření je diskrétní. To znamená, že se snažíme přes senzor, který zvládne zachytit jen konečný počet bodů, zaznamenat proces, který má ve své podstatě nekonečně mnoho těchto bodů. Z toho tedy logicky vyplývá, že musí dojít k jisté kompresi, aby takovéto snímání bylo možné. Tato komprese, ve chvíli kdy splníme určité podmínky, však bývá ve většině případů nepostřehnutelná. Jednou z hlavních podmínek je takzvaný Nyquistův–Shannonův vzorkovací teorém, který nám ve zkratce říká, že vzorkovací frekvence musí být minimálně dvakrát větší než frekvence signálu, který chceme zaznamenat, jinak bude docházet k aliasingu (nasnímáme jiné frekvence, než které tam skutečně jsou). Pokud se tedy bavíme o „skutečném“ komprimovaném snímání, většinou máme na mysli metody, ve kterých tento teorém obcházíme a tudíž jsme schopni snímat stejná data rychleji a s menší náročností na datová úložiště.[10]

Pro základní pochopení toho, jak komprimované snímání zjednodušeně funguje, zde uvedu příklad převzatý z [10].

Představme si, že se snažíme nasnímat zeď, jejíž část je namalována na bílo a zbytek je nabarven černě (viz obrázek 2.2a). Bílý povrch odráží $1 \text{ mJ/s}/\mu\text{m}^2$ a tmavý povrch neodráží žádné světlo. Naším úkolem je nasnímat tuto scénu na výstupní obrazovku, která má 10×10 pixelů tak, aby každý pixel reprezentoval $1 \mu\text{m}^2$ snímané plochy. První možností, která nás asi napadne, je změřit každý pixel a přenést obraz na výstup. V takovémto případě bychom udělali 100 měření (ať už najednou jedním velkým detektorem nebo postupně například jednopixelovým) a dostali bychom kýžený výsledek. Nicméně z nějakého důvodu (ať už třeba z finančního nebo kvůli technickým omezením) máme k dispozici jen detektor, který má dva pixely, snímající každý plochu $10 \mu\text{m} \times 5 \mu\text{m}$, což je $50 \times$ více, než potřebujeme. Z toho tedy plyne, že nemůžeme udělat přímé měření, ale můžeme vcelku přesně odhadnout hodnoty pro jednotlivé pixely.[10]

Nejprve tedy provedeme měření pomocí našeho dvoupixelového detektoru, kde na jednom pixelu dostaneme směsici nízké a vysoké intenzity a druhý pixel nezachytí žádné světlo (viz obrázek 2.2b). Dále si můžeme představit mřížku tvořenou 100 požadovanými pixely, kterou překryjeme přes naše měření (viz obrázek 2.2c). Následně zavedeme neznámou C_1 , která udává počet sloupečků u Pixelu 1, které jsou bílé a obdobně zavedeme neznámou C_2 , která udává počet bílých sloupečků u Pixelu 2. Za předpokladu, že tedy známe celkové naměřené intenzity pro Pixel 1 i Pixel 2,



Obrázek 2.2: (a) Scéna kterou chceme zachytit. (b) Měření pomocí dvou pixelů. Pixel 1 zachycuje směsici nízké a vysoké intenzity, naopak Pixel 2 nezachycuje žádné světlo. (c) Pomyslná mřížka 100 pixelů, na které chceme zobrazit výstup, překrývající měření pomocí dvou pixelů. (d) Získaný stopixelový obrázek, změřený pouze pomocí dvou pixelů. (převzato z [10])

můžeme jednoduše dopočítat C_1 i C_2 , z čehož můžeme rekonstruovat původní obraz, kde prvních $C_1 + C_2$ sloupců bude bílých a zbytek zůstane tmavý (viz obrázek 2.2d). Ve výsledku tedy máme snímek 10×10 získaný pomocí jednoho měření dvoupixelovým detektorem.[10]

Samozřejmě podobný proces není možné udělat vždy. Takovéto měření bylo možné uskutečnit jen díky tomu, že jsme o scéně znali dopředu několik dodatečných informací, jako například, že se skládá pouze ze dvou konkrétních intenzit (pouze bílé části nebo tmavé), že vždy po celé vertikální ose je intenzita stejná a že v horizontální ose je nalevo vyšší intenzita než napravo. Jediné, co jsme neznali, bylo místo (sloupeček), kde se intenzita mění. Podobně tedy pracuje i skutečné komprimované snímání, kdy k rekonstrukci obrazu využíváme různých dalších znalostí o scéně a detektoru, abychom mohli výsledný snímek získat z co možná nejmenšího množství dat. Dvě takové hlavní znalosti, které nám dovolují využívat komprimovaného snímání i pro složitější případy, jsou řídkost signálu (signál, v určité bázi, má co nejmeně nenulových prvků) a vzájemná minimální koherence vzorkovací matice (sloupce této matice jsou na sobě co nejmeně závislé).[10, 11]

2.3.2 Základní matematický popis

Postup řešení, který zde byl zmíněn, lze, jak již bylo řečeno, zobecnit a aplikovat na složitější příklady. V takovémto případě již však bude zapotřebí matematického popisu.

Zůstaňme tedy stále u stejného příkladu, kde snímáme zeď, která je zčásti bílá a zčásti černá. Prvním krokem je zvolit odpovídající model dat Ψ , který v našem případě vypadá následovně.

$$\Psi = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

Každý sloupec zde popisuje možné rozložení intenzit v horizontální ose. To znamená, že případu, kdy na zdi bude pouze jeden bílý pruh a detektor by tak měl zachytit jen jeden sloupeček pixelů s nenulovou intenzitou, odpovídá první sloupec matice Ψ , naopak případu, kdy je celá stěna nabarvená bíle, odpovídá poslední sloupec Ψ skládající se pouze ze samých jedniček. Našemu případu tedy bude odpovídat druhý sloupec matice, což samozřejmě předem nevíme. Matematicky se tento výběr dá zapsat jako

$$f = \Psi\alpha \quad (2.2)$$

kde Ψ je náš model dat a α je hledaný N -složkový vektor, který má v našem případě 1 na druhé pozici (vybíráme druhý sloupec) a ostatní prvky jsou nulové. Obecně však může mít α nenulových prvků víc, z čehož vyplývá, že f bude lineární kombinací sloupců matice Ψ .^[10]

Druhým krokem je následně vytvořit matici Φ představující detektor. V našem případě o detektoru víme, že se skládal ze dvou pixelů a každý tento pixel měřil celkovou intenzitu na oblasti o velikosti $N \times N/2 \mu\text{m}^2$, kde $N = 10$. Matice tedy bude vypadat následovně.

$$\Phi = \begin{bmatrix} N & N & N & N & N & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & N & N & N & N & N \end{bmatrix} \quad (2.3)$$

První řádek představuje Pixel 1, který sčítá intenzity v prvních pěti sloupcích a všech deseti řádcích, druhý řádek představuje Pixel 2, který sčítá intenzity v druhých pěti

sloupcích a opět logicky ve všech deseti řádcích našeho snímku. V tuto chvíli jsme tedy schopni popsat naše výsledné měření D jako

$$D = \Phi\Psi\alpha + n = \Phi f + n \quad (2.4)$$

kde Φ je matice detektoru, Ψ je matice modelu dat, α je hledaný vektor a n je v reálném měření vektor představující zašumění dat. Jedná se tedy o nedourčený systém lineárních rovnic (máme více proměnných než rovnic), ve kterém máme 10 proměnných (složky vektoru α) a 2 rovnice (měření pomocí Pixelu 1 a Pixelu 2). Nicméně jak bylo ukázáno výše, za předpokladu, že máme o problému dostatečné množství dalších informací, je možné takovýto problém řešit.[10]

Obecně můžeme k tomuto problému přistupovat jako k řešení lineární soustavy, ve které se snažíme najít co možná nejřidší řešení. Na takovýto problém lze aplikovat velké množství různých algoritmů, nicméně jako nejvhodnější se ukázaly ty využívající l_1 normu (viz například publikaci [12]). Řešení problému pak přechází v konvexní optimalizační úlohu

$$\min_{\alpha} \|\alpha\|_1 \quad \text{vzhledem k} \quad D = A\alpha \quad (2.5)$$

kde matice A je vzorkovací matice a konkrétně v našem zmiňovaném případě platí $A = \Phi\Psi$. Jak již však bylo řečeno, v reálných měřeních se vyskytuje šum reprezentovaný vektorem n . Pokud má n nulovou střední hodnotu (což bude v naprosté většině našich měření pravda), lze řešení nalézt jako

$$\min_{\alpha} \|\alpha\|_1 \quad \text{vzhledem k} \quad \|D - A\alpha\|_2^2 \leq \epsilon \quad (2.6)$$

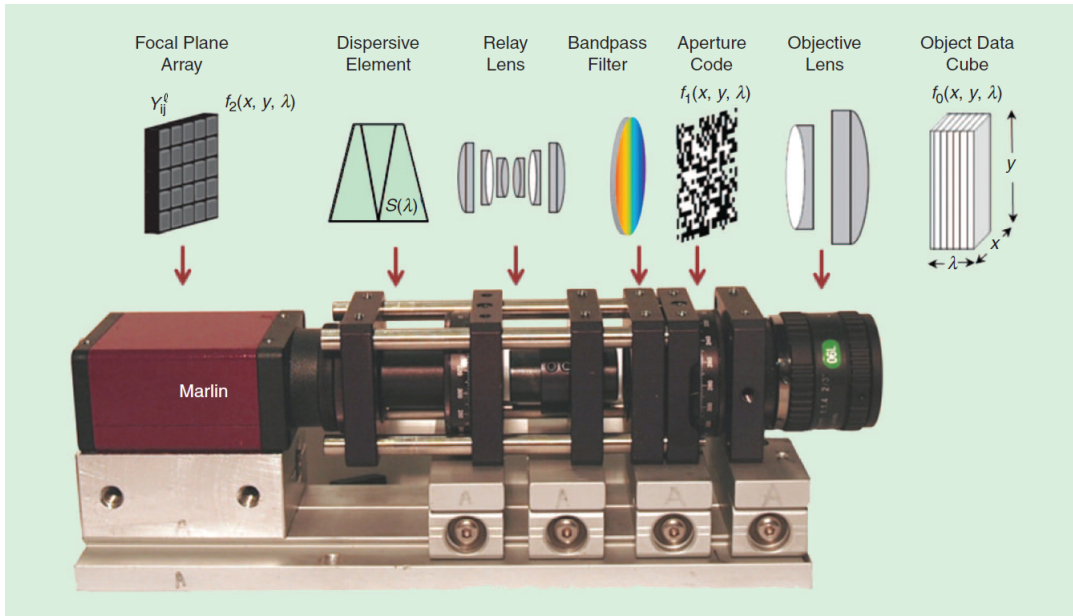
kde ϵ závisí na rozptylu šumu n . Tento problém nám tedy zjednodušeně řečeno říká, že se snažíme dosáhnout co nejpřesnějšího řešení, avšak zároveň se snažíme o to, aby řešením byl řídký vektor. Při volbě správných parametrů lze dále úlohu 2.6 převést za použití metody LASSO do následující podoby (konkrétní postup, jak toho docílit není pro tuto práci podstatný, tudíž ho zde nebudu uvádět, lze však najít například v publikaci [13])

$$\min_{\alpha} \left\{ \frac{1}{2} \|D - A\alpha\|_2^2 + \lambda \|\alpha\|_1 \right\} \quad (2.7)$$

kde λ je takzvaný regularizační parametr, který nám ovlivňuje požadovanou řídkost výsledného vektoru. Tudíž čím vyšší tento parametr bude, tím více klademe důraz na řídkost řešení (bude zde více nulových hodnot) a naopak, pokud bude mít tento parametr nízkou hodnotu, větší váhu přikládáme podobnosti modelu s měřením. [12, 13] Úloha 2.7 bude dále využívána při samotné rekonstrukci obrazu v metodě CASSI.

2.4 Metoda CASSI

Základní myšlenka CASSI je následující, snažíme se zachytit celou scénu v určitém rozmezí vlnových délek jedním snímekem a to tak, že využíváme rozkladu světla a náhodné masky. Jak taková soustava pro snímání pomocí této metody vypadá, je

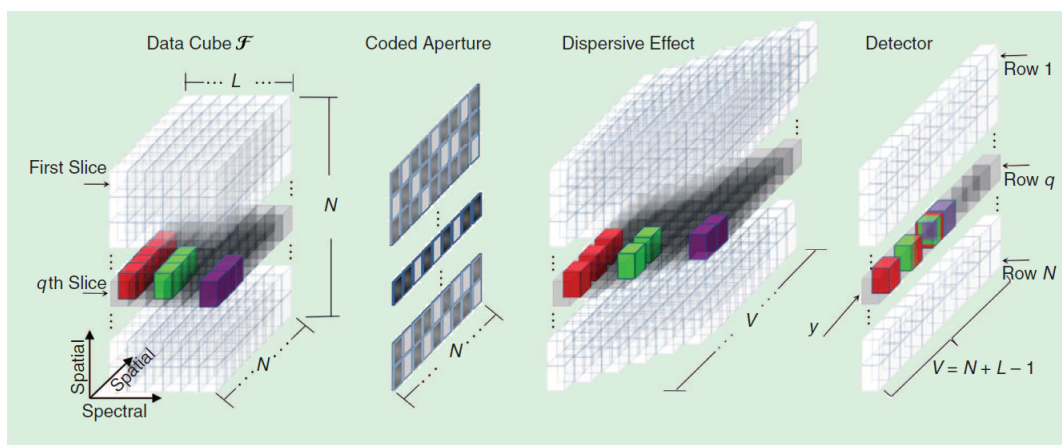


Obrázek 2.3: Soustava pro snímání pomocí CASSI (převzato z [11]) - seznam komponent zleva doprava: plošný detektor, disperzní element, soustava čoček pro invertování obrazu, barevný filtr, modulační maska, objektiv

možné vidět na obrázku 2.3 (jde pouze o ilustrační obrázek převzatý z [11] a nejedná se tedy přímo o soustavu, kterou jsem při práci využíval) .

Vpravo od soustavy máme v tomto případě nějaký signál (světlo), který chceme snímat, reprezentovaný datakrychlí o prostorových souřadnicích x a y a spektrální souřadnici λ . Světlo je zde skrz objektiv posíláno na modulační masku, která má na náhodných pozicích černé pixely. Tyto pixely ideálně nepropouští žádné světlo, díky čemuž dostáváme jistou výhodu při rekonstrukci obrazu, ke které se dostanu dále v této práci. Za touto maskou může být filtr, který nám propustí jen požadované rozmezí vlnových délek, skrz tento filtr projde světlo do soustavy čoček, která nám pomáhá paprsky správně fokusovat a případně i invertovat, neboť za objektivem byl obraz převrácený. Dále pak paprsky procházejí skrz element, který nám umožní rozmítnout spektrum (kupříkladu skleněný hranol pomocí disperze nebo optická mřížka pomocí difrakce). Rozmítnuté spektrum je nakonec zachyceno na detektor a následně může dojít k samotné rekonstrukci obrazu. [11, 14]

Jak takové zachycení obrazu vypadá je možno vidět na obrázku 2.4. Datakrychle F reprezentující hyperspektrální snímek o rozměrech $N \times N \times L$ (kde N jsou prostorové souřadnice a L je spektrální souřadnice) je namodulována maskou o rozměrech $N \times N$ tak, že část signálu neprojde dál (černé pixely masky, které nepropouští světlo). Dále takto upravený signál prochází skrz disperzní element a jeho spektrum je rozloženo takovým způsobem, že každý řez v ose L je od předešlého posunut o jeden pixel podél osy x . Jednotlivé řezy osy L se tedy zčásti překrývají a vytváří tak obraz o rozměrech $(N + L - 1) \times N$, kde intenzita jednotlivých pixelů odpovídá součtu intenzit příslušných pixelů z překrývajících se řezů datakrychle.[11]



Obrázek 2.4: Průběh snímání pomocí CASSI (převzato z [11])

Zde je tedy možné si všimnout, v čem spočívá ona komprese dat, o kterou se pokoušíme. Původně, kdybychom snímali běžnou metodou, bychom museli snímat $N \times N \times L$ pixelů, nicméně v tomto případě nám na rekonstrukci obrazu (data-krychle) stačí pouze snímat $(N + L - 1) \times N$ pixelů, což je skutečně velký rozdíl v rozmezí několika řádů.

2.4.1 Rekonstrukce obrazu

Jak již bylo zmíněno dříve, pro samotnou rekonstrukci lze použít mnoho různých přístupů a algoritmů, kde každý přístup má určité výhody a nevýhody. Pro nás se však ukázalo jako nejvhodnější využívat algoritmus TwIST (Two-Step Iterative Shrinkage/Thresholding Algorithm) řešící konvexní optimalizační úlohu 2.7.

Aby bylo možné tohoto přístupu využít, jsou zde dvě zásadní podmínky na data, s kterými počítáme. První podmínkou je nekoherence (sloupečky dané matice na sobě musí být co nejméně závislé) vzorkovací matice respektive masky, kterou používáme k modulaci signálu. Tato podmínka nám umožňuje z matice detektoru (matice reprezentující výstup kamery, na kterém je zaznamenáno rozmítnuté spektrum) získat s větší přesností zpět snímky jednotlivých vlnových délek a vytvořit z nich tak opět datakrychle.

Ve chvíli, kdy by jednotlivé sloupečky byly koherentní, začaly by se jednotlivé vlnové délky příliš překrývat a vytvářely by se zde jednolitě plochy (ať už světlé, či takové, kde není zaznamenáno žádné světlo). V takovémto případě bychom nebyli schopni provést rekonstrukci, protože bychom od sebe nebyli schopni odlišit jednotlivé vlnové délky. Naopak ve chvíli, kdy jsou jednotlivé sloupečky na sobě lineárně nezávislé a tudíž mají navzájem jedničky a nuly (pixely co propouští a nepropouští světlo) na jiných pozicích a za předpokladu, že víme, jak vypadá maska, můžeme od sebe do jisté míry odlišit jednotlivé vlnové délky, což nám pomůže při rekonstrukci datakrychle.

Druhou podmínkou, kterou musí data splňovat je řídkost. V tomto případě využijeme

toho, že u signálů (především pak u těch obrazových) lze často najít bázi, ve které je daný signál řídký. Takovouto bázi může být například Fourierova báze, kosínová báze, či asi nejčastěji používaná báze vlnková. V našem případě tedy využíváme vlnkovou transformaci, přesněji řečeno Haarovu transformaci, což je nejjednodušší případ vlnkové transformace.

Úlohu 2.7 lze pak přepsat na náš konkrétní problém jako

$$\min_{\mathcal{I}} \left\{ \frac{1}{2} \|D - A\mathcal{I}\|_2^2 + \lambda \|\mathcal{I}\|_1 \right\} \quad (2.8)$$

kde D představuje matici nasnímaných dat detektoru, \mathcal{I} představuje hledanou matici reprezentující rekonstruovaný snímek převedený do řídké báze (v našem případě tedy Haarova vlnková báze - viz kapitola 2.4.2), λ je regularizační parametr určující „jak moc“ řídké řešení se má algoritmus snažit hledat a A představuje nekoherentní vzorkovací matici. Pro názornost zde ještě rozepíši A jako

$$A = S \cdot M \cdot H^{-1}(\mathcal{I}) \quad (2.9)$$

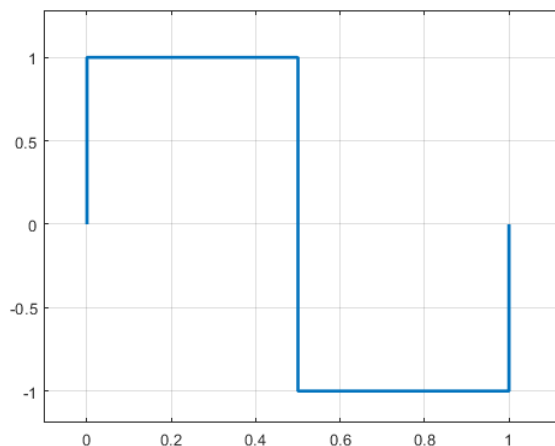
kde S je operátor, který koresponduje s rozmítnutí spektra na detektoru, M je naše nekoherentní maska a H^{-1} reprezentuje inverzní Haarovu transformaci, která převádí \mathcal{I} z vlnkové báze zpět na běžné zobrazení.

2.4.2 Haarova transformace

Haarova transformace se řadí mezi vlnkové transformace, které převádí signál do vlnkové báze. Tyto transformace využívají k převodu signálu prostorově omezené funkce, takzvané vlnky (například na rozdíl od Fourierovy transformace, která používá neomezené periodické funkce sinus). Výhodou této transformace je například to, že jsme schopni získat informaci nejen o tom, jaká frekvence se v signálu nachází, ale i o místě jejího výskytu. Zároveň lze také mnohem snáze s prostorově omezenými funkcemi interpretovat současně jak vysoké tak nízké frekvence signálu, díky čemuž jsou tyto transformace ideální pro zpracování obrazových signálů, kde se snažíme mít co nejvíce detailů, ale zároveň vnímáme celý snímek jako nějaký celek.[15]

Princip těchto transformací je zjednodušeně řečeno následovný. Nejprve určíme takzvanou „mother wave“, která nám slouží jako vzor. Následně tuto vlnku posouváme po signálu a měníme její velikost a snažíme se dosáhnout největší podobnosti takto upravené vlnky v daném místě a ukládáme koeficienty její polohy a měřítko. V takovémto případě se jedná o spojitou vlnkovou transformaci, která ze své podstaty vytváří velké množství koeficientů, nicméně v praxi se využívá diskrétní vlnková transformace, která má předem dané měřítko, které je 2^i , kde i je celé nezáporné číslo. Z tohoto důvodu je také lepší u obrazových signálů, na které chceme aplikovat vlnkovou transformaci, používat rozměry $2^i \times 2^i$ pixelů.[15]

Způsobů jak zvolit zmiňovanou „mother wave“ je spousta. Nejjednodušším typem této vlnky (a také velmi často používaným typem) je takzvaná Haarova vlnka (viz obrázek 2.5), která má dvě hlavní výhody, kterými jsou symetričnost a jednoduchá implementace.



Obrázek 2.5: Haarova vlnka („mother wave“)

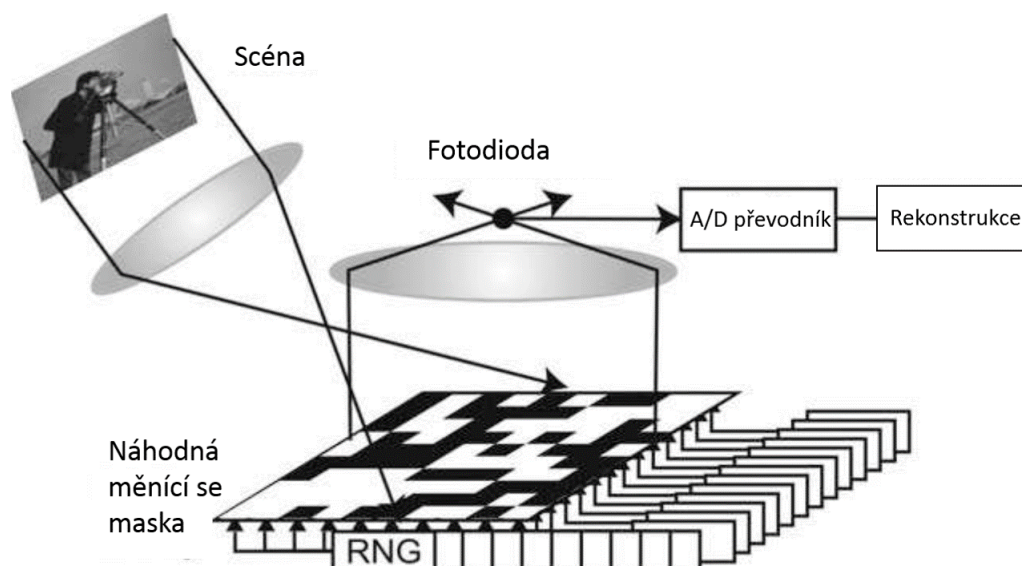
2.5 Jednapixelová kamera

Metoda CASSI samozřejmě není jediným přístupem, jak provádět komprimované snímání. Další významnou metodou je snímání pomocí takzvané jednapixelové kamery.

Při této metodě je scéna zaznamenávána na detektor o jednom pixelu skrz měnící se náhodnou masku (respektive pseudonáhodnou masku, protože podmínkou pro rekonstrukci obrazu je naše znalost jednotlivých masek), která propouští jen část záření (viz obrázek 2.6). Detektor takto postupně zaznamenává celkovou intenzitu obrazu pro několik těchto masek a my jsme následně schopni ze znalosti těchto intenzit a toho, jak vypadají příslušné masky, pomocí různých algoritmů rekonstruovat původní obraz.[16, 17]

Jednou z možností jak řešit tento přístup je například zařízení DMD (Digital micro-mirror device), kde se nachází množství malých zrcátek, které mohou být nastaveny do dvou poloh, buď odráží světlo na detektor nebo ne. Díky tomu nám vzniká maska, kterou můžeme poměrně rychle měnit a snadno zaznamenávat, v jakém místě světlo propouští a v jakém ne.

Obecně je výhodou jednapixelových kamer jejich cena, oproti CASSI zde máme detektor o jednom pixelu, který bude většinou levnější a snazší na výrobu, než plošný 2D detektor používaný u CASSI. Naopak nevýhodou je rychlost, se kterou jsme schopni snímat. Na rozdíl od CASSI, kde celou scénu zaznamenáváme jedním měřením, zde jich musíme provést větší množství. Zároveň však stále provádíme méně měření, než kdybychom používali klasické postupy a měřili scénu pixel po pixelu. Například podle publikace [17] je možné barevnou scénu o rozměrech 256×256 pixelů (celkově 65536 pixelů) rekonstruovat pomocí 6500 měření jednapixelovou kamerou.[16, 17]



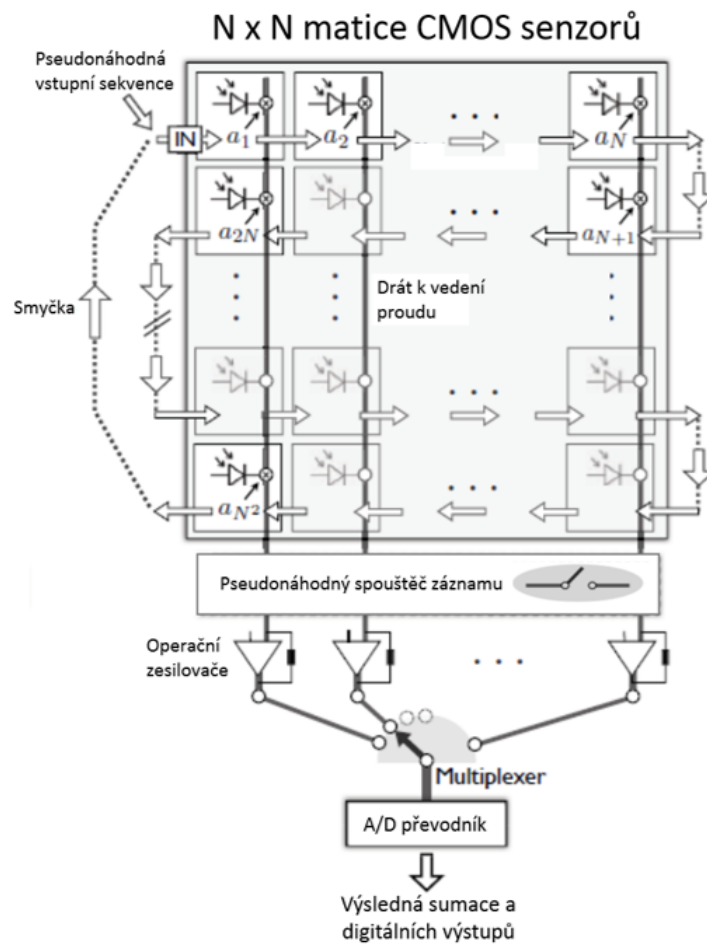
Obrázek 2.6: Konstrukce systému pro snímání jedopixelovou kamerou - jako náhodnou masku je možné využít například soustavu zrcátek (převzato z [18])

2.6 CMOS komprimované snímání

Dalším příkladem, jak aplikovat komprimované snímání, je například metoda využívající matici CMOS snímačů. Konstrukce této metody je následující (viz obrázek 2.7 převzatý z [19]), v ploše je umístěno $N \times N$ pasivních (tvořeny pouze fotodiodou) CMOS snímačů, kde každý je propojen se svým sousedem klopným obvodem (poslední je pak propojen opět s prvním) schopným uchovávat 1 bit informace, čímž celá síť tvoří jednoduchý posuvný registr. Pokud tedy na první snímač přivedeme informaci, celá sekvence uložená v posuvném registru se posune o jednu pozici (ve skutečnosti se však celá sekvence pouze „točí“ dokola, protože do prvního pixelu posíláme informaci z toho posledního). [19]

Před samotným měřením je do posuvného registru uložena pseudonáhodná sekvence, která je nám známá. Následně proběhne první měření, kde každý CMOS senzor vybudí určitý elektrický proud v závislosti na intenzitě světla, která na něho dopadá. Znaménko tohoto proudu je upraveno podle bitu, který je v té chvíli na klopném obvodu příslušícího danému pixelu. Následně je proud veden pomocí vodivého drátu spojujícího jednotlivé sloupčky pixelů do zesilovače. Výstup zesilovače je následně převeden na digitální hodnotu, sečten s ostatními výstupy zesilovačů příslušících ostatním sloupčkům a tato hodnota je uložena. V dalším kroku je pak posunuta informace z posledního klopného obvodu na první a celá sekvence uložená v registru se tak posune a snímání může proběhnout znovu. [19]

Snímání však neprobíhá po každém posunutí, ale pouze v náhodných (respektive pseudonáhodných) momentech. Výsledky z jednotlivých měření se ukládají a poté, co proběhne N^2 posunutí registru (N^2 je počet našich senzorů) a celý systém se tedy dostane do výchozího stavu, může proběhnout rekonstrukce obrazu. Celý proces je



Obrázek 2.7: Konstrukce soustavy na snímání pomocí CMOS komprimovaného snímání (převzato z [19])

v podstatě analogií k jednopixelové kameře, avšak zde odpadají některé problémy spojené s optikou jako je například různá odrazivost každého zrcadla, správná fokusace paprsků a podobně. Podle publikace [19] stačí u této metody na velmi přesnou rekonstrukci obrazu $N^2/3$ měření.[19]

3 Hyperspektrální detekční systém

3.1 Základní popis

Hyperspektrální detekční systém viditelného spektra, zkráceně HDES-VIS (dále jen HDES), je soustava využívající metodu CASSI, na které jsem prováděl všechna měření. V tuto chvíli se jedná zatím o nedokončený projekt ve výzkumném centru TOPTEC.

Pro snímání i rekonstrukci obrazu je zde používán software MATLAB.

3.1.1 Optická soustava

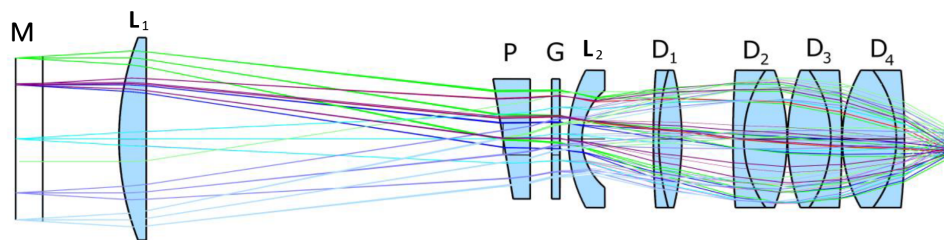
Při výrobě soustavy bylo cíleno především na dostupnost jednotlivých čoček a tudíž i jejich cenu. Většinu použitých optických elementů lze tedy běžně zakoupit.[20]

Konstrukci optické soustavy lze rozdělit do dvou částí a to části vstupní nacházející se před maskou pro modulaci signálu a části disperzní nacházející se za modulační maskou (viz obrázek 3.1).

Vstupní část se v původním návrhu skládala ze systému pěti čoček s jednou modulační maskou. Momentálně se však využívá dvou masek a namísto systému pěti čoček jako vstupní části používáme dvě konvexní čočky, před kterými se nachází nastavitelná šterbina se stínítkem a difuzor. Čočky jsou umístěny těsně nad sebe a to z toho důvodu, že využíváme dvou modulačních masek, taktéž umístěných nad sebe, a vyžadujeme, aby skrz obě masky procházel stejný obraz. Ke správnému zobrazení obou obrazů také přispívá šterbina, která musí mít správnou velikost, jinak bychom viděli buď překrývající se obrazy (v případě, že by byla příliš velká) nebo dva částečné obrazy (za předpokladu, že by byla šterbina příliš malá). Před šterbinu je také možné umístit difuzor, který využíváme při práci s laserem.

Za dvojčočkou, co nejbližší k disperzní části soustavy, je taktéž umístěn kolektor (anglicky „field lens“), což je konvexní spojná čočka, která zajišťuje, aby paprsky, které vychází z dvojčočky pod příliš velkým úhlem, dopadaly na masku, potažmo na detektor.

Druhou část soustavy tvoří disperzní část (viz obrázek 3.1 a tabulka 3.1), kde je zachycený obraz, který prošel skrz modulační masku, pomocí difrakční mřížky rozmítnut na detektor obdobně jako je ukázáno na obrázku 2.4. Mimo difrakční mřížky je zde také optický hranol, který však neslouží primárně pro rozklad spektra, ale pro zalomení paprsků tak, aby byly v ideálním případě opět rovnoběžné



Obrázek 3.1: Disperzní část optické soustavy (L - čočka, D - optický dublet, P - hranol, G - difrakční mřížka, M - modulační maska) - převzato z [20]

Tabulka 3.1: Seznam optických elementů v disperzní části optické soustavy (L - čočka, P - hranol, G - difrakční mřížka, D - dublet), „Inf“ znamená nekonečný poloměr, tudíž rovný povrch (převzato z [20])

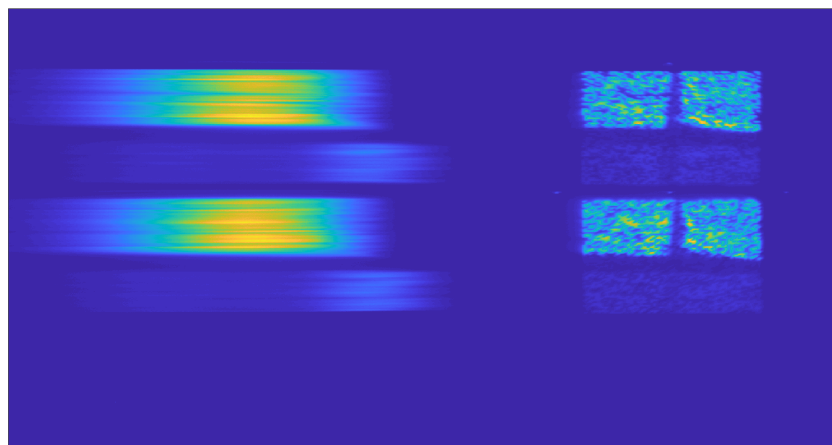
	Přední poloměr křivosti (mm)	Zadní poloměr křivosti (mm)	Materiál
L1	103,01	Inf.	N-BK7
P	-	-	SF11
G	-	-	N-BK7
L2	53,70	23,45	N-BK7
D1	376,25	93,11	SF2
		-109,86	N-BK7
D2	363,10	44,17	SF10
		-71,12	N-BAF10
D3	71,12	-44,17	N-BAF10
		-363,10	SF10
D4	50,80	-41,69	E-BAF11
		-247,70	N-SF11

s optickou osou. Stejně jako u vstupní části i zde byl kladen důraz na dostupnost a relativně nízkou cenu všech optických elementů a mimo hranol a čočku L2, které byly vytvořeny na zakázku, je zbytek optických elementů běžně dostupných. Na výstupu této části je pak detektor kamery.

3.1.2 Detektor

Jak již bylo zmíněno výše, v aktuální sestavě využíváme dvou obrazů nad sebou, což sice znamená, že zaznamenáváme větší množství dat, nicméně stále se pohybujeme řádově v mnohem menších číslech, než kdybychom nevyužívali komprimovaného snímání a snímali scénu pixel po pixelu. Výhoda dvou obrazů nám tedy výrazně zlepšuje samotnou rekonstrukci za cenu nepatrného zpomalení celého procesu rekonstrukce.

Dále také u obou obrazů zaznamenáváme mimo prvního řádu difrakce také nultý



Obrázek 3.2: Ukázka vstupních dat z detektoru - v levé části je vidět spektrálně rozmítnutý první řád a v pravé části je vidět nultý řád

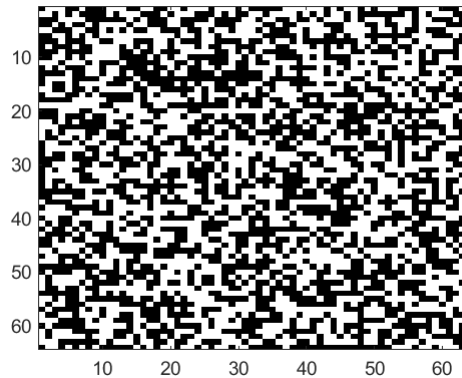
řád, který v zásadě vizuálně odpovídá scéně, kterou snímáme, což nám opět poskytuje jistou výhodu při rekonstrukci.

Jako detektor používáme kameru Manta G-507B od výrobce Allied Vision o rozlišení 2464×2056 , která využívá CMOS senzory o velikosti $3,45 \mu\text{m} \times 3,45 \mu\text{m}$ [21]. Samotné propojení kamery s počítačem je zajištěno softwarem Matlab.

Jak vypadají vstupní data, která zaznamenáme pomocí tohoto detektoru, je možné vidět na obrázku 3.2.

3.1.3 Modulační maska

V HDESu je využíváno dvou masek o velikosti 64×64 pixelů, které byly vytvořeny pomocí fotolitografie nanesením tenké chromové vrstvy na skleněný podklad. Masky jsou umístěny nad sebe a jejich předlohou byla náhodná binární matice 64×64 , kde polovina pixelů jsou nuly a zbytek jedničky. Obě masky jsou navzájem komplementární, což znamená, že vrchní maska je positivem této matice a spodní maska je jejím negativem (tzn. tam kde v původní matici byly jedničky jsou pro spodní masku nuly a opačně). Matici modulační masky pro vrchní část obrazu je možné vidět na obrázku 3.3. Rozměry každé z masek jsou přibližně $13,55 \text{ mm} \times 13,55 \text{ mm}$, což znamená, že jeden pixel takovéto masky má rozměry $0,21 \text{ mm} \times 0,21 \text{ mm}$. [20] Využití masek, které jsou navzájem positivem a negativem má velkou výhodu při samotné rekonstrukci. U nultého řádu můžeme této skutečnosti využít tak, že oba obrazy sečteme, čímž dostaneme matici plnou jedniček, což je původní obraz, jak bychom ho viděli, kdybychom sečetli všechna spektra (v našem případě, kdy pracujeme s viditelným světlem, bychom tak dostali obraz, který ve skutečnosti vidíme). Naopak u prvního řádu se vyplatí oba obrazy odečíst, protože tak pro danou vlnovou délku částečně eliminujeme vliv ostatních vlnových délek.



Obrázek 3.3: Modulační maska pro vrchní část obrazu - bílá místa představují pixely, které propouští světlo, tmavá pak ty pixely, které jsou neprůhledné

3.1.4 Difrakční mřížka

K rozdělení spektra využíváme, jak již bylo řečeno, difrakční mřížku. V našem případě je použita mřížka Thorlabs 500 nm blaze z borosilikátového skla, která má 300 vrypů na milimetr. Výhodou využití mřížky oproti hranolu jako disperzního elementu je především rovnoměrné rozložení spektra na detektoru. Jde tedy o to, že v případě disperzního spektra (tedy při použití hranolu) je vzdálenost mezi bodem reprezentujícím vlnovou délku kupříkladu 500 nm a druhým bodem představujícím 600 nm rozdílná než vzdálenost mezi body 600 nm a 700 nm. Naopak u difrakčního spektra (tedy při použití difrakční mřížky) budou tyto vzdálenosti vždy stejné v celém spektru. Výhodou je tedy jednodušší kalibrace, neboť víme, že pozice jednotlivých částí spektra na detektoru je přímo úměrná jejich vlnové délce.

Nevýhodou oproti hranolu však je rozdílná propustnost světla na rozdílných vlnových délkách.

3.1.5 Zpracování vstupních dat

Než započne samotná rekonstrukce, je nutné vstupní data získaná z kamery nejprve zpracovat, to znamená ořezat a zmenšit. Ořezání se provádí tak, že z matice surových dat, kterou je možné vidět například na obrázku 3.2, postupně v Matlabu vyjme submatice odpovídající jednotlivým řádům. Souřadnice submatic jsou pro všechny měření stejné a získávají se při kalibraci soustavy. Ve výsledku tedy získáme čtyři matice, kde dvě odpovídají prvním řádům a dvě odpovídají nultým řádům. Velikosti jednotlivých matic se mohou po každém zkalibrování lišit o několik pixelů, nicméně velikost matice nultého řádu se pohybuje přibližně kolem 520×530 pixelů kamery a velikost matice prvního řádu kolem 1520×520 pixelů kamery.

Následné zmenšení matic nultých řádů na námi požadovanou velikost 64×64 pixelů a matic prvních řádů na velikost 185×64 pixelů (velikost prvního řádu v horizontální ose se může lišit v závislosti na kalibraci) je prováděno pomocí funkce „shrinking“.

Tato funkce pracuje obdobně jako již implementovaná funkce v Matlabu „imresize“, nicméně její popis by byl zdouhavý a pro tuto práci nepodstatný, tudíž ho zde nebudu uvádět. Jejím výstupem jsou pak čtyři zmenšené matice, kde každý pixel těchto matic představuje přibližně 8×8 pixelů kamery.

3.2 Rekonstrukce obrazu

Rekonstrukce obrazu je založena na optimalizační úloze 2.8 a jejím řešení pomocí TwIST algoritmu. Kvůli zpřehlednění zde nejprve uvedu značení jednotlivých matic a operátorů, které bude dále používáno.

Vstupní data detektoru D budou značena jako „vstupní detektor D “, rekonstruovaná hypespektrální data rychle ve vlnkové doméně \mathcal{I} bude značena jako „data rychle \mathcal{I} “ a výraz $A\mathcal{I}$, který v úloze 2.8 odpovídá odhadu detektoru, bude značen jako „detektor \bar{D} “.

První důležitou poznámkou je, že k matici A ze zmíněné úlohy 2.8, která nám zajišťuje převod data rychle \mathcal{I} na detektor \bar{D} , lze přistupovat dvěma způsoby.

Prvním přímočařejším způsobem je přistupovat k ní jako ke skutečné matici. V takovém případě bychom si vstupní detektor D a data rychli \mathcal{I} z rovnice 2.8 převedli na vektory a tudíž bychom měli vektor D o $N(N + L - 1)$ složkách a vektor \mathcal{I} o N^2L složkách. Matice A , která zajišťuje spektrální rozmítnutí, modulaci a zpětné převedení z vlnkové domény do domény prostorové (viz rovnice 2.9), by pak musela mít rozměry $N(N + L - 1) \times N^2L$. Teoreticky zde tedy žádný problém není a převod data rychle \mathcal{I} na detektor \bar{D} máme provedený v jednom kroku a algoritmus by si matematicky s takovýmto postupem byl schopen poradit. Nicméně prakticky narážíme na zásadní problém, kterým je velikost matice A , protože je pravděpodobné, že i při poměrně malých velikostech N a L (rozměry naší data rychle) by se nám tato matice nemusela vejít do operační paměti počítače. Například v případě, že bychom pracovali s rozměry $N = 64$ a $L = 122$ (jako je tomu v našem případě) měla by matice A přibližně $6 \cdot 10^9$ pixelů a za předpokladu, že by se jednalo o formát double a každý pixel by tedy zabíral 8 bajtů, celkově by nám matice zabírala 48 GB operační paměti, což je pro obyčejné počítače zatím naprosto mimo jejich kapacity.

Druhým způsobem, který z očividných důvodů využíváme i my, je převádět data rychli \mathcal{I} na detektor \bar{D} „po částech“. To znamená, že se nesnažíme převést celou data rychli v jednom kroku, ale počítáme každý řez data rychle (vlnovou délku) zvlášť. Zároveň také nepřistupujeme k matici A jako ke skutečné matici, ale spíše jako k funkci, která nám tento převod zajišťuje, což bude vysvětleno dále.

K samotné rekonstrukci tedy využíváme soubor funkcí v Matlabu, které slouží jak k prvotnímu zpracování dat z vstupního detektoru D (jejich načtení, ořezání, atd.) tak i pro rekonstrukci založenou na TwIST algoritmu. Celý proces rekonstrukce je ovládán pomocí skriptu (pro jednoduchost označený jako „main“) sloužícího jako takové uživatelské rozhraní, díky kterému můžeme jednoduše nastavovat parametry, které do celého procesu vstupují. Tento skript se nám také stará o prvotní úpravu dat a také konečné zobrazení výsledků. Důležitou funkcí ve které se odehrává proces

Tabulka 3.2: Přístup ke zpracování snímků (hranaté závorky značí jednu matici, tudíž například [pozitiv - negativ] značí jednu výslednou matici, ve které se navzájem oba snímky odečetly)

Mód 1	[pozitiv]
Mód 2	[pozitiv] [negativ]
Mód 3	[pozitiv - negativ]

samotné rekonstrukce je funkce označená jako „reconstruct“, která slouží primárně k předání parametrů TwIST algoritmu a zároveň jeho spuštění v požadovaném počtu iterací. Dále se zde pak nachází soubor funkcí pro samotný TwIST algoritmus. Jádro tohoto algoritmu a jeho matematická podstata je mimo rozsah této práce, a proto jí tu nebudu uvádět.

Mimo parametrů, kterými ovlivňujeme rekonstrukci (jako například požadovaná přesnost výsledku, maximální počet iterací, atd.) jsou stěžejní tři funkce, které značně ovlivňují průběh TwISTu a výsledky, které nám bude dávat. Jsou to funkce pro počáteční odhad (označená jako „makeInitGuess“), která ze vstupních dat vytvoří počáteční odhad datakrychle, což slouží jako výchozí bod TwIST algoritmu, dále pak funkce pro převod datakrychle \mathcal{I} na detektor \bar{D} (označená jako „imageEncode“), která odpovídá matici A v 2.8 a následně její inverze pro převod detektoru \bar{D} na datakrychli \mathcal{I} (označená jako „imageDecode“).

Dalším důležitým faktorem, který ovlivňuje způsob rekonstrukce, je jak přistupujeme k samotným datům. Jak již bylo zmíněno dříve v kapitole 3.1.1, nesnímáme pouze jeden obraz, ale dva, skrz dvě navzájem komplementární masky. Nabízí se tedy několik možností, jak k datům přistupovat. Momentálně jsou implementovány tři přístupy (módy) viz tabulka 3.2.

Mód 1 využívá pouze vrchní masku a tudíž pracujeme jen s vrchní částí detektoru. Zbylé dva módy pak používají obě masky a pracují tedy již s celým detektorem. V módu 2 po celou dobu rekonstrukce zůstávají obě části detektoru oddělené a pracujeme tedy se dvěma maticemi. Naopak v módu 3 od sebe na začátku rekonstrukce odečteme obrazy prvních řádů a přičteme k sobě obrazy nultých řádů a ve výsledku tedy pracujeme pouze s jednou maticí reprezentující detektor. Výhody a nevýhody těchto přístupů budou dále diskutovány v kapitole 4.3.

V následující části se budu zabývat třemi již zmíněnými funkcemi, je však nutné podotknout, že každá z těchto funkcí se liší v závislosti na použitém módu. Popis, který zde předložím, se tedy bude týkat módu 1, nicméně pro ostatní módy jsou tyto funkce principiálně velmi podobné a mají jen dílčí rozdíly.

3.2.1 Počáteční odhad

Algoritmus TwIST ke svému fungování potřebuje mít zadaný výchozí bod, ideálně co nejblíže samotnému řešení a z toho důvodu je nutné dělat počáteční odhad, k čemuž slouží již zmíněná funkce „makeInitGuess“. Tato funkce převádí vstupní detektor D

na datakrychli \mathcal{I} a transformuje ji do vlnkové domény pomocí haarovy transformace. Hlavní část této funkce, kterou zde budu popisovat, je možné vidět na obrázku 3.4. Celý proces probíhá po částech, namísto vytvoření datakrychle v jednom kroku používáme cyklus, ve kterém postupně vytváříme každý řez. Nejprve tedy ze vstupního detektoru D extrahujeme část snímku (actSlice), se kterou budeme dále pracovat. Dále přenásobíme matici našeho aktuálního snímku maticí masky (nejedná se však o klasické maticové násobení, operátor $.*$ v Matlabu značí násobení po prvcích, neboli Hadamardův součin - kdykoliv se tedy dále v práci budu zmiňovat o násobení dvou matic, je myšleno právě toto násobení po prvcích). Po této operaci tedy máme náš aktuální snímek v podobě, kde na místech, kde v masce byly nuly, jsou i v aktuálním snímku nuly a tam kde byly jedničky, zůstávají prvky nezměněny (to ovšem platí pouze pokud využíváme 1. mód, viz tabulka 3.2, v ostatních módech je matice masky upravena obdobně jako jsou zpracovány snímky, tudíž například u 3. módu se maska skládá z čísel 1 a -1). Dále se do rekonstrukce zapojuje i nultý řád (ZOimage), kdy k aktuálnímu snímku přičítáme nultý řád vydělený poměrem mezi prvním a nultým řádem na dané vlnové délce (FOvZO). V podstatě to znamená, že když se na dané vlnové délce nultý řád téměř neprojevuje (má minimální intenzitu), bude FOvZO velké (intenzita prvního řádu je mnohonásobně vyšší než intenzita nultého řádu) a tudíž se přičte k aktuálnímu snímku jen malá část nultého řádu a naopak. Následně už může proběhnout haarova transformace do vlnkové báze a zapsání aktuálního snímku do datakrychle. Jak je vidět z posledního řádku kódu, nejedná se o 3D-datakrychli, ale o dvourozměrnou matici, kde jsou jednotlivé snímky řazeny vedle sebe a ne za sebe, což ulehčuje následné zpracování této matice TwIST algoritmem. Tento proces pak opakujeme pro každý řez datakrychle (to znamená pro každou vlnovou délku, kterou počítáme).

3.2.2 Převod detektoru na datakrychli

Pro převod dat z detektoru \bar{D} na datakrychli \mathcal{I} se využívá funkce „imageDecode“. Hlavní část kódu, kterou zde budu popisovat, je možné vidět na obrázku 3.5. Nejprve tedy opět dojde k extrahování příslušného řezu a tento řez je následně přenásoben maskou. Dále je zde možnost nahrazení záporných prvků, v tomto případě se dají buď záporné prvky vynulovat nebo jsou přepsány jejich absolutní hodnotou. Tato možnost může částečně zlepšit kvalitu rekonstrukce a je zde proto, že algoritmus TwIST může dávat ve výsledku záporné hodnoty. Matematicky je to samozřejmě správně, nicméně z fyzikálního hlediska představují hodnoty pixelů intenzitu světla, která nemůže být záporná. Následně je do výpočtu zařazen i nultý řád, který je nejprve, obdobně jako tomu bylo u počátečního odhadu, přenásoben maskou a vydělen poměrem intenzity prvního a nultého řádu a poté přičten k aktuálnímu řezu. Ve finále je stejně jako u počátečního odhadu na aktuální řez použita haarova transformace a je přidán do výsledné 2D-datakrychle \mathcal{I} . Celý tento proces je pak opakován pro každou měřenou vlnovou délku.

```

for kk = 1:c                                     % a*b*c je velikost datakrychle

    actSlice = detector(:, kk : kk+b-1 );         % extrahování jednoho řezu
    %                                              datakrychle z detektoru

    if useRMinInitG==1                           % použití masky
        actSlice = actSlice.*randomMask;
    end

    if useZOinInitG==1                           % použití nultého řádu
        if useFOvZOinCalc==1                     % použití poměru intenzit
            % mezi prvním a nultým řádem
            actZOimage = ZOimage/FOvZO(kk);
        else
            actZOimage = ZOimage;
        end
        actSlice = actSlice + actZOimage;
    end

    x0guess(:,kk*b-b+1 : kk*b )=haar_2d(actSlice); % vytvoření datakrychle
end

```

Obrázek 3.4: Část funkce v Matlabu pro vytvoření počátečního odhadu (tzn. převod vstupních dat na datakrychli)

```

for kk = 1:c                                     % a*b*c je velikost datakrychle

    actSlice = detector(:, kk : kk+b-1 );         % extrahování jednoho řezu
    %                                              datakrychle z detektoru
    actSlice = actSlice.*randomMask;             % použití masky

    switch imDecZeroing                          % nulování záporných prvků
        case 'zero'                             % místo záporných prvků nuly
            actSlice(actSlice<0)=0;
        case 'abs'
            actSlice=abs(actSlice);
    end

    if useZOinCalc==1                           % použití nultého řádu

        if useFOvZOinCalc==1                   % použití poměru intenzit mezi
            % prvním a nultým řádem
            actZOimage = ZOimage/FOvZO(kk);
        else
            actZOimage = ZOimage;
        end

        actSlice = actSlice + actZOimage;

    end

    %                                              % vytvoření datakrychle
    datacubeOutputHWT(:,kk*b-b+1 : kk*b ) = haar_2d(actSlice);
end

```

Obrázek 3.5: Část funkce v Matlabu pro převod dat detektoru na datakrychli

```

detector = 0; % inicializace detektoru
ZOimage = 0; % inicializace nultého řádu
for kk=1:c % a*b*c je velikost datakrychle
    % vyjmutí aktuálního řezu z
    % datakrychle
    actSlice=haar_2d_inverse(datacubeHWT(:, kk*b-b+1 : kk*b ));

    switch imEncZeroing % nulování záporných prvků
        case 'zero' % místo záporných prvků nuly
            actSlice(actSlice<0)=0;
        case 'abs' % absolutní hodnota řezu
            actSlice=abs(actSlice);
        end

    actSlice = actSlice.*randomMask; % použití masky

    if useFOvZOinCalc==1 % použití poměru intenzit mezi
        % prvním a nultým řádem
        ZOimage=ZOimage + actSlice/FOvZO(kk);
    else
        ZOimage=ZOimage+actSlice;
    end

    % vytvoření detektoru
    detector(:,kk:kk+b-1)=detector(:,kk:kk+b-1)+actSlice;
end

if useZOinCalc==1 % použití nultého řádu
    detectorOutput = [detector, ZOimage];
else
    detectorOutput = detector;
end

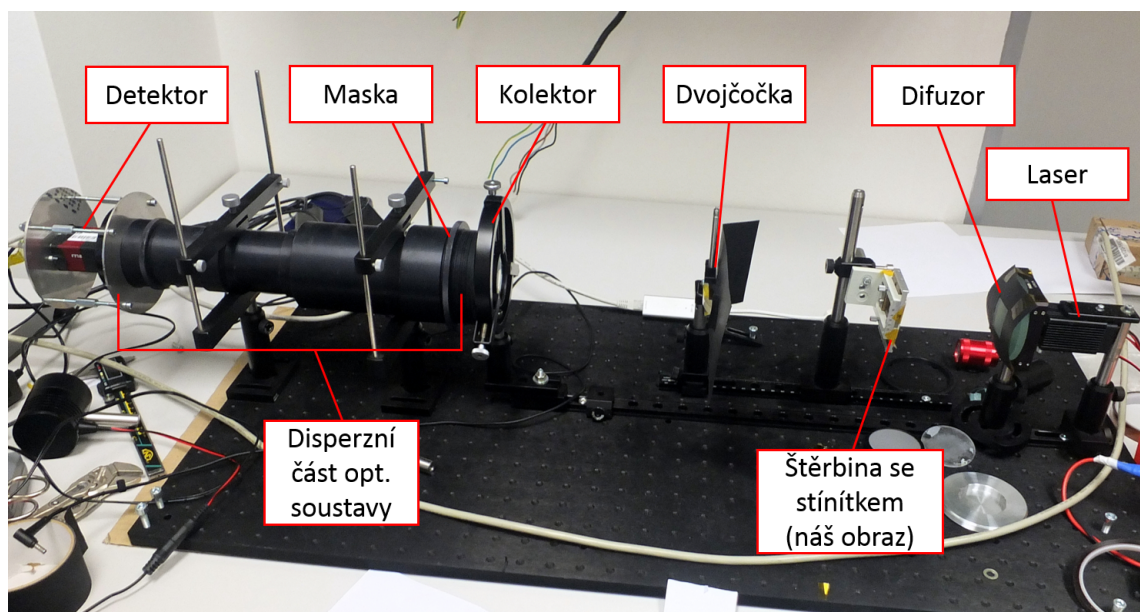
```

Obrázek 3.6: Část funkce v Matlabu pro převod datakrychle na detektor

3.2.3 Převod datakrychle na detektor

Pro převod datakrychle \mathcal{I} na detektor \bar{D} využíváme funkce „imageEncode“, která je ve své podstatě inverzní funkcí k „imageDecode“ a odpovídá matici A v rovnici 2.9. Hlavní část kódu, kterou zde budu popisovat, je možné vidět na obrázku 3.6.

Nejprve je nutné získat řez, se kterým budeme pracovat. Toho docílíme tak, že vyjmeme příslušný řez z 2D-datakrychle \mathcal{I} , která je v tuto chvíli ve vlnkové doméně, a aplikujeme na něho inverzní haarovu transformaci. Následně je zde opět možnost nahrazení záporných prvků, která sice funguje stejně jako u „imageDecode“, nicméně se jedná o dva nezávislé procesy, a proto můžeme například v „imageDecode“ nahrazovat záporné prvky absolutní hodnotou (imDecZeroing = 'abs') a v „imageEncode“ tento parametr vůbec nepoužívat (imEncZeroing = 'none'). Následně aktuální řez vynásobíme maskou a můžeme přistoupit k výpočtu nultého řádu, kde ke stávajícímu nultému řádu přičítáme náš aktuální řez vynásobený maskou a poměrem intenzit mezi prvním a nultým řádem (ten je zde použit ze stejného důvodu jako u počátečního odhadu a převodu detektoru na datakrychli). V dalším kroku přičteme aktuální řez k detektoru v místě, které je posunuto o jeden sloupeček vpravo oproti předchozímu řezu (toto posunutí odpovídá rozmítnutí spektra, respektive operátoru S v rovnici



Obrázek 3.7: Sestava HDES

2.9). Celý tento proces se opět opakuje pro každou měřenou vlnovou délku. V posledním kroku pak na výstup buď odesíláme buď samotný rozmítnutý obraz (první řád), v případě, že v rekonstrukci nechceme využívat nultý řád ($\text{useZOinCalc} = 0$), anebo matici, kde jsou za sebou spojeny jak matice prvního řádu, tak matice nultého řádu ($\text{useZOinCalc} = 1$).

3.3 Justáž a kalibrace

Pro správné fungování aparatury je nutné provést justáž a kalibrační měření. Pro jednodušší orientaci v jednotlivých komponentech a jejich uchycení v soustavě je zde přiložen obrázek 3.7.

Prvním krokem justáže je zaostření detektoru na masku. Toho docílíme tak, že nejprve uvolníme šrouby držící detektor a pomocí zaostřovacího kolečka, které se nachází před detektorem, doostříme obraz masky. V tomto kroku je však problém vyplývající z konstrukce uchycení detektoru. Při zpětném dotažení šroubů totiž dochází k malému posunutí detektoru a tudíž i rozostření obrazu, je tudíž nutné s tímto rozostřením počítat a při ostření nechat obraz nepatrně rozostřený. Následné posunutí detektoru způsobené utažením šroubů poté v ideálním případě způsobí správné zaostření a my můžeme přistoupit k dalšímu kroku.

Dalším krokem je zarovnání detektoru, masky a difrakční mřížky (maska a difrakční mřížka jsou součástí tubusu s disperzní částí soustavy, viz obrázek 3.1). Snažíme se tedy o to, aby maska, detektor i difrakční mřížka byly v jedné vodorovné ose a nebyly vůči sobě natočeny. Toho lze docílit buď natočením samotného detektoru nebo pootočením masky, samotnou difrakční mřížkou hýbat nelze neboť je pevně umístěná

v tubusu.

Jakmile jsou detektor, maska a mřížka ve správné pozici a obraz je zaostřen, zkontrolujeme dotažení šroubů u detektoru a pootočíme celým tubusem s disperzní částí optické soustavy tak, aby spodní a vrchní hrana masky byly rovnoběžné s podložkou (respektive s dvojčočkou). Následně zkontrolujeme, jestli je celý tubus ve vodorovné poloze.

Dále posuneme kolektor (čočka pro nasměrování paprsků, které vychází z dvojčočky pod příliš velkým úhlem, na detektor) tak, aby se nacházel co nejbližší masce a aby jeho střed byl ideálně v optické ose soustavy. Stejně tak nastavíme dvojčočku a štěrbinu do stejné výšky jako je zbytek komponent.

Následně posuneme dvojčočku do vzdálenosti přibližně 20 cm od masky a štěrbinu taktéž přibližně 20 cm od dvojčočky (dáno vzdáleností předmětového a obrazového ohniska). Poté posouváme dvojčočkou po kolejnici a snažíme se, aby spot, který osvětluje, byl na stejném místě masky jak na horním tak na spodním obrazu. Jakmile toho docílíme, nastavíme velikost štěrbinu tak, aby se oba obrazy navzájem nepřekrývaly.

Celá justáž je kvůli současné konstrukci vcelku časově náročná a tudíž je do budoucna plánována úprava především týkající se uchycení detektoru k tubusu s disperzní částí soustavy.

Po provedení justáže je nutné provést kalibrační měření, které sestává z měření homogenně osvětlené scény pomocí laseru a homogenně osvětlené scény pomocí širokopásmového zdroje světla (v našem případě používáme halogenovou lampu). Z těchto měření jsou následně pomocí příslušného skriptu v Matlabu zjištěny koeficienty pro ořezání obrazu a pro kompenzaci intenzit mezi horním a spodním obrazem, které jsou následně používány při rekonstrukci.

4 Měření

4.1 Intenzita prvního a nultého řádu

Jak již bylo uvedeno v kapitole 3.1.4 difrakční mřížka má na každé vlnové délce jinou propustnost světla, tudíž průběh intenzity spektra bude záviset na vlnové délce. Zároveň tato propustnost bude u každé vlnové délky rozdílná pro první a nultý řád, tudíž poměr intenzity mezi první a nultým řádem bude taktéž závislý na vlnové délce. Jedno z prvních měření, které jsem tedy prováděl, mělo za cíl zjistit, jak tato závislost poměru intenzity mezi prvním a nultým řádem na vlnové délce vypadá.

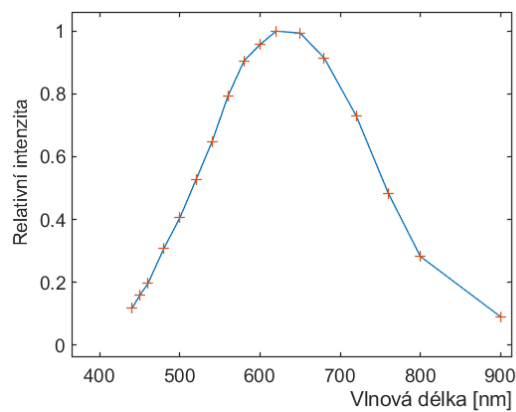
Pro měření bylo využito monochromátoru, kterým se z halogenové lampy vybíralo úzké pásmo vlnových délek o šířce přibližně 15 nm. Toto světlo bylo následně pomocí optického vlákna přivedeno na papírové stínítko a osvětlovalo zde bodovou oblast. Obraz na stínítku byl následně zaznamenán pomocí soustavy HDES. Měření probíhalo v rozsahu vlnových délek od 440 nm do 900 nm, což je přibližně maximální rozsah, který je HDES schopen zaznamenat. Velikost vlnových délek byla následně ověřována spektrometrem od Ocean Optics Flame.

Nejprve tedy pro první a nultý difrakční řád byla pro každou vlnovou délku z nasnímaných dat zjištěna intenzita obrazu. Průběh této intenzity v závislosti na vlnové délce je možné vidět na obrázku 4.1. Je možné si zde povšimnout, že pro první řád je maximální intenzita v 620 nm a u nultého řádu je toto maximum posunuto k 720 nm.

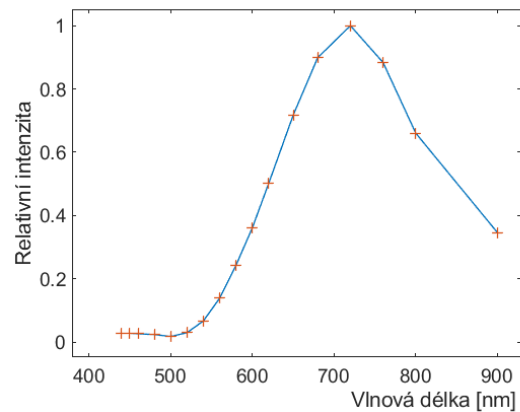
Pro každou měřenou vlnovou délku pak mohl být z těchto údajů vypočten poměr mezi intenzitou prvního a nultého řádu. Závislost tohoto poměru na vlnové délce je možné vidět na obrázku 4.2.

4.2 Testovací scény

Pro finální testování jsem si zvolil tři typově odlišné scény. Prvním typem scény byl neprůhledný kříž osvětlený zeleným Nd:YAG laserem vyzařující na vlnové délce 532 nm, jehož světlo bylo difusně rozptýleno, aby byla scéna homogenně osvětlena. Druhým typem scény byla opět scéna s křížem, tentokrát však osvětlená halogenovou lampou od firmy Thorlabs (QTH10/M). Třetí scéna se skládala ze čtyř kvadrantů, přičemž každý kvadrant obsahoval jinak barevný filtr (na každé vlnové délce má tedy každý kvadrant jinou intenzitu - viz obrázek 4.3) a jako osvětlení nám opět posloužila halogenová lampa Thorlabs.

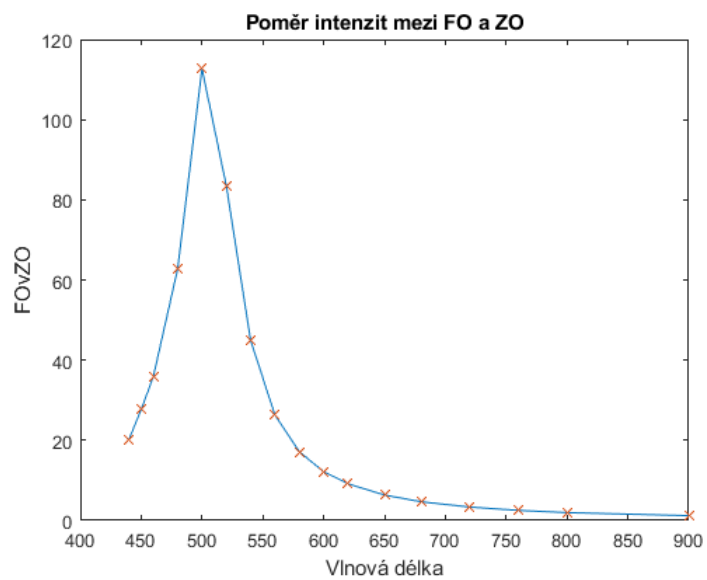


(a) První řád

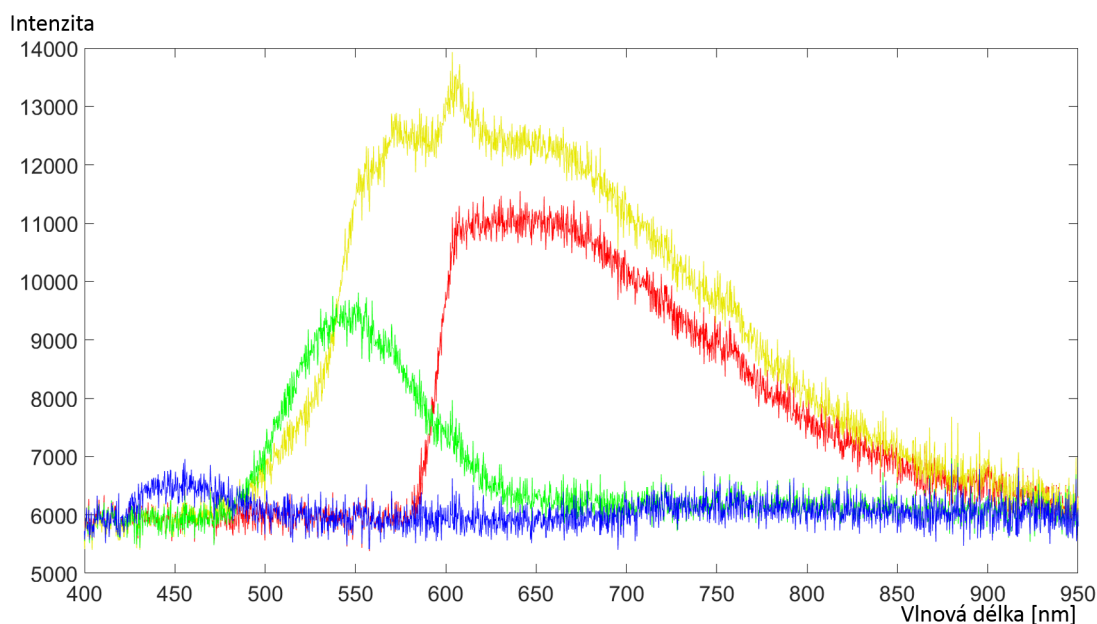


(b) Nultý řád

Obrázek 4.1: Relativní intenzita obrazu na detektoru (vztaženo k maximální hodnotě v jednotlivých obrazech)



Obrázek 4.2: Poměr intenzity mezi prvním (FO) a nultým (ZO) řádem



Obrázek 4.3: Spektrum jednotlivých barevných filtrů osvětlených halogenovou lampou Thorlabs QTH10/M (měřeno pomocí spektrometru Ocean Optics Flame) - barvy křivek odpovídají barvám filtrů

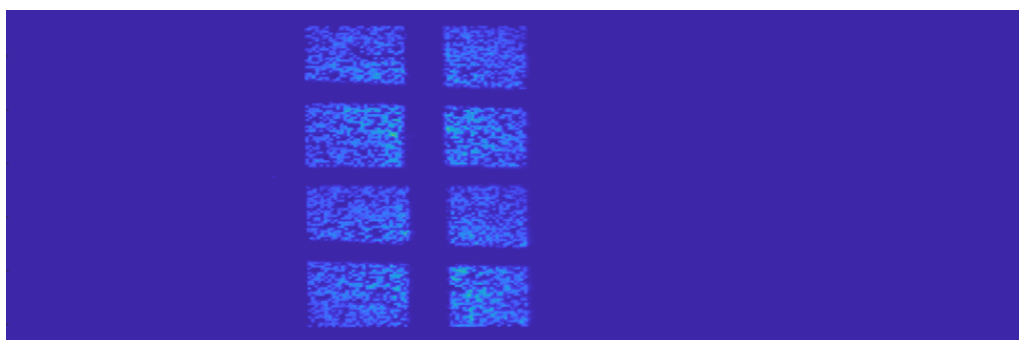
Na obrázku 4.4 je možné vidět vstupní data, která nasnímal detektor pro jednotlivé scény. Jak již bylo zmíněno například v kapitole 3.1.2, snímáme jak nultý řád (pravá část obrázku), tak i spektrálně rozmítnutý první řád (levá část obrázku). U scény osvětlené laserem (obrázek 4.4a) si však můžeme všimnout zdánlivé absence nultého řádu, což je dáno tím, že má oproti prvnímu řádu velmi slabou intenzitu a tudíž zde není vidět.

4.2.1 Problémy ve zobrazování scény

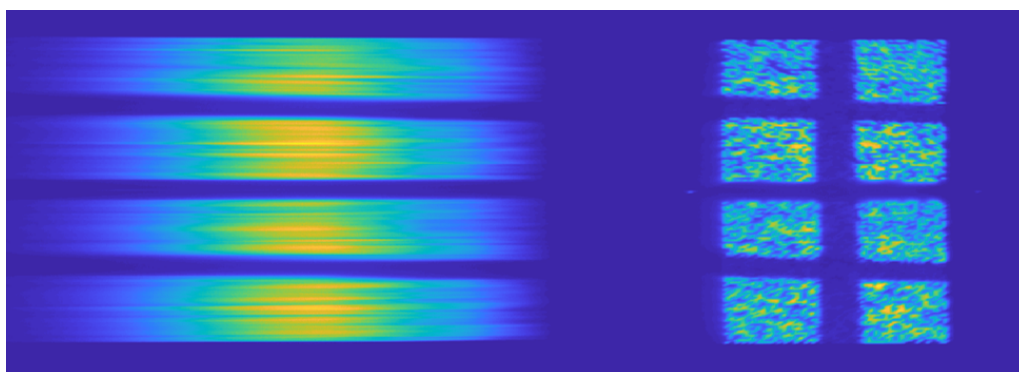
Při měřeních jsme identifikovali několik problémů, týkajících se samotných scén a i obecně celé zobrazovací soustavy, které mají do jisté míry vliv na kvalitu rekonstrukce.

Jedním z těchto problémů je fakt, že jsme nebyli schopni docílit toho, aby se na obrazech nultého řádu a prvního řádu zobrazovala přesně totožná scéna, vždy byly navzájem posunuty. Toto posunutí se pohybovalo okolo 10 pixelů kamery, což už může ovlivnit kvalitu rekonstrukce, neboť to odpovídá více jak jednomu pixelu masky. Obdobně jsme také zjistili, že vrchní a spodní hrany nultého řádu nejsou rovnoběžné. I když se nám tedy podařilo vrchní hranu vyrovnat do vodorovné polohy, spodní hrana měla na celé své délce v ose y rozdíl mezi počátkem a koncem přibližně 3 pixely kamery. Tento rozdíl tedy není ani jeden pixel masky a tudíž je možné ho považovat za zanedbatelný.

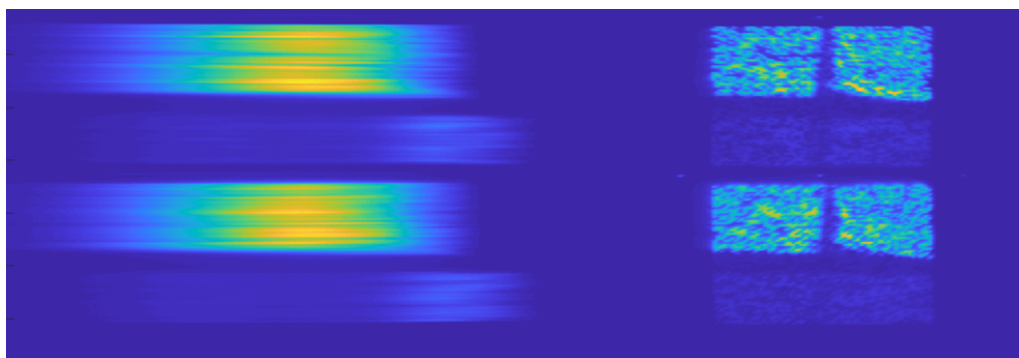
Další dva problémy se týkají nedokonalosti optické soustavy. Zaprvé k nultému



(a) Vstupní data z detektoru scény s křížem osvětlené zeleným laserem



(b) Vstupní data z detektoru scény s křížem osvětlené halogenovou lampou



(c) Vstupní data z detektoru scény s barevnými filtry osvětlené halogenovou lampou

Obrázek 4.4: Vstupní data z detektorů pro jednotlivé testované scény

řádu přistupujeme tak, že ho uvažujeme bez jakéhokoliv spektrálního rozmítnutí. Ve skutečnosti to však není zcela pravda, neboť v soustavě máme hranol (viz obrázek 3.1), který způsobuje, že je obraz nultého řádu částečně rozmítnut. Toto rozmítnutí sice není tak velké jako u prvního řádu, nicméně se projevuje rozmazáním obrazu. Zadruhé jsme narazili na problém způsobený nejspíše odstraněním vstupní části optické soustavy (viz kapitola 3.1.1), kdy nejsme schopni obraz „rovnoměrně“ zaostřit. To znamená, že když se nám podařila zaostřit vrchní část obrazu, spodní zůstala rozostřená a naopak. Toto rozostření je však poměrně malé, tudíž se nejedná o problém, který by razantně snižoval kvalitu rekonstrukce.

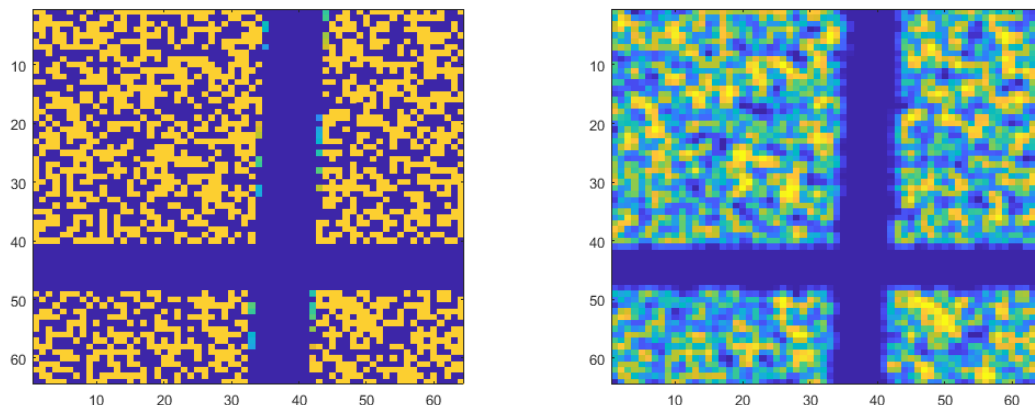
Dále jsme narazili na dva problémy týkající se především scén osvětlených lampou nebo jiným zdrojem světla se širokým spektrem. Ukázalo se, že obraz prvního řádu má „soudkovitý“ tvar. To znamená, že se směrem od středu k vyšším a nižším vlnovým délkám zužuje. V nejširším bodě má na kameře šířku přibližně 540 pixelů a v nejúžším přibližně 530 pixelů. Rozdíl mezi těmito místy je tedy 10 pixelů kamery (opět to odpovídá rozdílu více jak jednoho pixelu na masce), což už může být problém při ořezání. Druhým problémem, který se projevuje u scén osvětlených lampou, je překryv prvního řádu vysokých vlnových délek (okolo 900 nm a více) a druhého řádu nízkých vlnových délek (okolo 450 nm a méně). Kvůli tomuto překryvu mají vysoké vlnové délky vyšší intenzitu prvního řádu než tomu je ve skutečnosti. Tento problém byl vyřešen použitím žlutého filtru, který odstínil záření o vlnové délce 515 nm a méně.

U dat pro scénu osvětlenou zeleným Nd:YAG laserem jsme naopak narazili na problém, že HDES mimo vlnovou délku laseru zaznamenává i vlnovou délku okolo 800 nm s velmi slabou intenzitou. Zprvu jsme tento problém dávali za vinu špatné kalibraci, nicméně i po zkalibrování tento problém přetrvával. Až později se ukázalo, že laser v sobě má budící diodu o vlnové délce právě 808 nm, tudíž byl pro další měření použit před laserem infračervený filtr, který tento problém vyřešil.

4.3 Rekonstrukce pomocí jednotlivých módů

Pro testování jednotlivých přístupů a parametrů byly nejprve ke každému typu scény vytvořeny nasimulované datakrychle, kde byl zohledněn jak průběh spektrální intenzity samotné lampy (případně laseru), tak dříve zjištěná závislost intenzity mezi prvním a nultým řádem na různých vlnových délkách. Z těchto datakrychlí byly pak vytvořeny matice detektorů, které sloužily jako vstupy pro náš algoritmus. Výstupy (tedy opět datakrychle) se následně daly jednoduše porovnat s původní nasimulovanou datakrychlí, čímž jsme mohli určit jaký vliv mají jednotlivé parametry na rekonstrukci.

První testování bylo provedeno na ideálních datech (nebyl zde šum, nebyla rozmazaná atd.). Na těchto datech se však ukázalo, že za ideálních podmínek jsme schopni provést velmi přesné rekonstrukce za použití jakéhokoliv módu, tudíž pro přiblížení se skutečným datům bylo nutné tato data upravit tak, aby se zde projevilo rozmazání a šum. Toho bylo docíleno tak, že se každý snímek zvětšil, byla na něho



(a) Data bez rozmazání

(b) Data s rozmazáním

Obrázek 4.5: Porovnání nultého řádu ze simulovaných dat bez rozmazání a ze simulovaných dat s rozmazáním

použita konvoluce pro rozmazání a opět se zmenšil na původní velikost. Zvětšení bylo osminásobné a probíhalo pomocí funkce Matlabu „imresize“. K následné konvoluci byla použita matice 5×5 skládající se, kvůli co nejmenšímu vlivu na změnu celkové intenzity, z hodnot $\frac{1}{25}$, samotná konvoluce pak byla prováděná pomocí funkce Matlabu „conv2“. Porovnání ideálních a rozmazaných dat je možné vidět na obrázku 4.5. Nakonec proběhlo i testování na reálných datech, jejichž vstupy je možné vidět na obrázku 4.4.

Testování jsem prováděl za změny sedmi různých parametrů. Hlavním z těchto parametrů byl regularizační parametr, který udává řídkost výsledného řešení (viz parametr λ v rovnici 2.8). Dalšími parametry, jejichž použití je možné vidět v kapitolách 3.2.1 až 3.2.3, jsou „useRMinInitGuess“, který umožňuje použití masky při počátečním odhadu, „useZOinInitGuess“, který umožňuje počítat s nultým řádem v počátečním odhadu (viz obrázek 3.4), „imDecZeroing“ a „imEncZeroing“, které umožňují nulovat nebo případně vytvářet absolutní hodnoty ze záporných prvků řezu a „useZOinCalc“, který umožňuje využívat nultého řádu v průběhu rekonstrukce (viz obrázek 3.5 a 3.6). Posledním testovaným parametrem byl pak parametr „ZOmean“, který slouží k tomu, aby byl v průběhu rekonstrukce každý řez data rychle přenásoben maticí nultého řádu (tato matice je však z důvodu minimalizace vlivu na celkovou intenzitu daného řezu data rychle nejprve vydělena svým průměrem). Tento parametr nám v některých případech pomáhal zlepšit prostorovou informaci.

Většině z uvedených parametrů je možné nastavit dvě hodnoty 0 a 1. 0 znamená, že daný parametr nepoužíváme a do rekonstrukce tedy vůbec nevstupuje, naopak 1 znamená, že ho při rekonstrukci využijeme. Výjimku tvoří parametry pro nahrazování záporných hodnot, které mají tři možné hodnoty - „abs“ (záporné hodnoty nahrazujeme jejich absolutní hodnotou), „zero“ (záporné hodnoty nahrazujeme nulou) a „none“ (nahrazování záporných hodnot se nepoužívá).

V následujících třech podkapitolách se budu zabývat rekonstrukcí pomocí jednotlivých módů a konfiguracemi jejich parametrů pro jednotlivé testované scény, podkapitola 4.3.4 je pak shrnutím celého měření. Všechny testovací rekonstrukce byly prováděny při maximálním počtu 300 iterací algoritmu TwIST. Nastavení vyššího počtu iterací pak jen rekonstrukci zpomalovalo a na výsledek nemělo zásadní vliv. V mnoha případech však algoritmus konvergoval k řešení dříve než po 300 iteracích. Vyhodnocení měření probíhalo u simulovaných a reálných dat rozdílně. Pro simulovaná data jsme měli k porovnání nasimulovanou datakrychli, tudíž bylo možné poměrně přesně určit, jak dobrá rekonstrukce je, respektive jestli se se změnou parametrů zlepšila či zhoršila. Toto vyhodnocení bylo prováděno pomocí již dříve vytvořené funkce v Matlabu „evaluateSimilarity“. Tato funkce v zásadě porovnává původní a rekonstruovanou datakrychli a to takovým způsobem, že hledá minimum z funkce

$$g(x_1, x_2) = \sqrt{\sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^L (F_{ijk} - x_1 \bar{F}_{ijk} - x_2)^2} \quad (4.1)$$

,kde F a \bar{F} jsou původní a rekonstruovaná datakrychle a $N \times N \times L$ jsou jejich rozměry. Výsledné minimum pak bylo vyděleno počtem pixelů datakrychle, čímž jsme dostali hodnotu, která by se dala přirovnat průměrnému rozdílu mezi datakrychlemi na jeden pixel (dále označováno jako podobnost datakrychlí) a čím je toto číslo menší, tím přesnější rekonstrukce byla.

U reálných dat bylo porovnávání výsledků čistě subjektivní a byl brán zřetel na průběh intenzity spektra, které jsem porovnával se spektrem změřeným pomocí spektrometru Ocean Optics Flame, a na celkový vzhled rekonstruované scény.

4.3.1 Mód 1

Mód 1 je nejjednodušším z přístupů, kterými rekonstruujeme obraz. Kvůli využití pouze vrchní části detektoru nemáme tolik informace, jako u ostatních módů, a proto může, zvláště u složitějších scén, vykazovat horší výsledky než ostatní přístupy, nicméně zároveň nám to do jisté míry dovoluje porovnat dvojčočkový přístup s původním návrhem HDES, který pracoval pouze s jednou čočkou.

Obecně se dá říct, že rekonstrukci u tohoto módu zhoršují parametry „RMinInitGuess“ a nahrazování záporných prvků. Zároveň, stejně jako pro ostatní módy, k rekonstrukci vždy přispívaly parametry „useZOinInitGuess“ a „useZOinCalc“ a pomocí parametru „ZOmean“ jsme byli schopni získat lepší prostorovou informaci.

Jak již bylo zmíněno v předchozí části, při testování na simulovaných ideálních datech byly výsledky pro všechny tři módy velmi dobré a podobné, proto nemá význam je zde rozebírat. Dále se tedy budu věnovat pouze simulovaným neideálním datům a reálným měřením.

Simulovaná data:

- **Scéna s křížem osvětlená laserem** - V porovnání s ostatními módy vykazoval mód 1 u těchto dat nejhorší výsledky, nicméně stále se jednalo o velmi

přesnou rekonstrukci, kdy se podařilo přesně zachytit vzhled kříže i vlnovou délku, ve které pomyslný laser vyzařoval. Nejlepších výsledků jsem dosáhl při nastavení regularizačního parametru na hodnotu 400 a podobnost datakrychlí byla v tomto případě $5 \cdot 10^{-3}$.

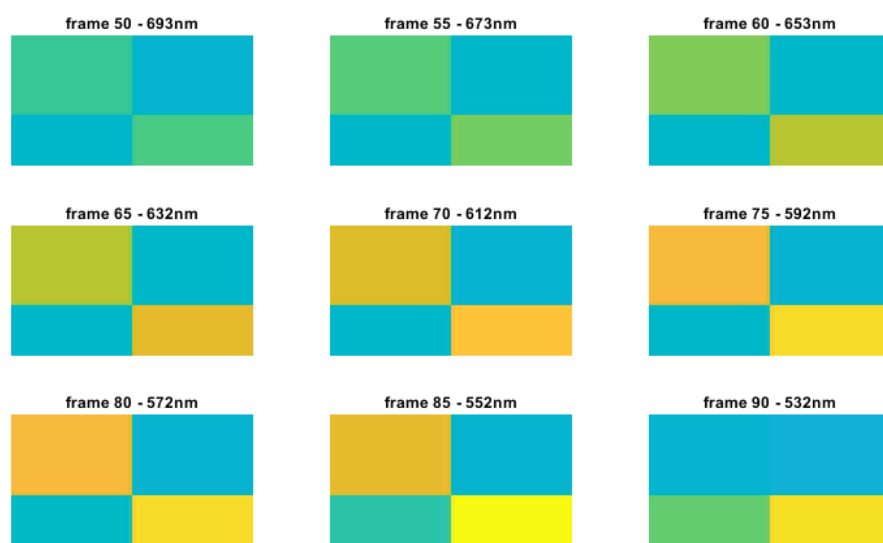
- **Scéna s křížem osvětlená lampou** - U těchto dat k rekonstrukci přispíval parametr „ZOmean“, bez kterého nebyl tento mód schopný správné rekonstrukce a ve výsledku na většině řezů zobrazil pouze horizontální čáru. Při zapnutí tohoto parametru a při nastavení regularizačního parametru na hodnotu 5000 byl však algoritmus schopný zobrazit kříž na většině řezů datakrychle a dosáhnout podobnosti datakrychlí $6,9 \cdot 10^{-4}$.
- **Scéna s barevnými filtry osvětlená lampou** - Tato data byla obecně problémem pro všechny tři módy, nicméně jako nejlepší se pro tuto scénu ukázaly módy 1 a 2, které vykazovaly téměř totožné výsledky. Nejlepších výsledků se podařilo dosáhnout při nastavení regularizačního parametru na vysoké hodnoty (5000 a více) a podobnost datakrychlí pak byla $2,6 \cdot 10^{-4}$. Na obrázku 4.6b je vidět několik řezů z rekonstruované datakrychle, na obrázku 4.6a jsou pak stejné řezy z původní nasimulované datakrychle.

Reálná data:

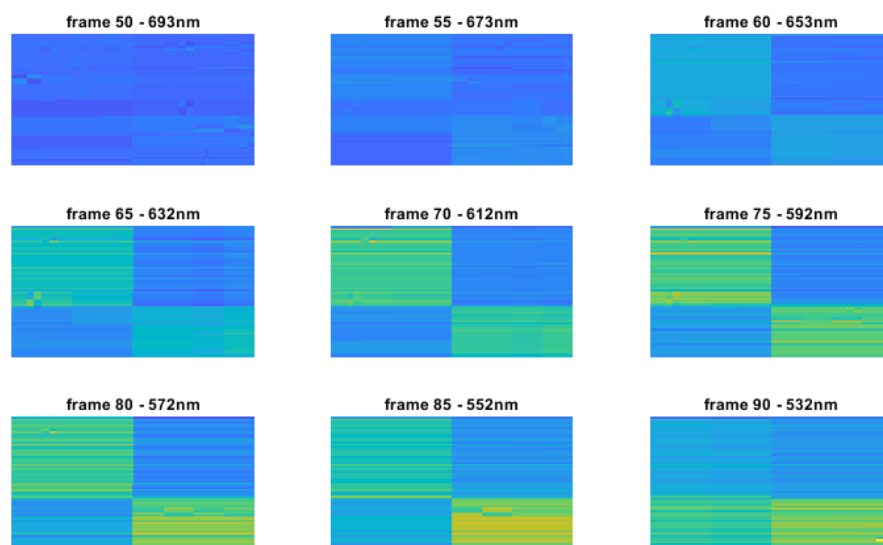
- **Scéna s křížem osvětlená laserem** - Obdobně jako u simulované scény, v porovnání s ostatními módy, vykazoval mód 1 nejhorší výsledky projevující se horší rekonstrukcí spektra (kříž se zobrazoval na více vlnových délkách). Nejlepších výsledků se podařilo dosáhnout při nastavení regularizačního parametru na hodnotu 400.
- **Scéna s křížem osvětlená lampou** - Stejně jako u simulovaných dat, i zde bylo nutné zapnout parametr „ZOmean“, bez kterého nebyl algoritmus schopen provést příliš kvalitní rekonstrukci a zobrazoval pouze horizontální čáru namísto kříže. Nejlepších výsledků se s tímto módem podařilo dosáhnout při nastavení regularizačního parametru na hodnotu 700. V takovémto případě se kříž správně zobrazil na většině vlnových délek, nicméně průběh intenzity spektra byl oproti naměřenému průběhu poměrně zkreslený.
- **Scéna s barevnými filtry osvětlená lampou** - U těchto dat jsem nejlepších výsledků dosáhl při nastavení regularizačního parametru na hodnotu 1000, při vyšších hodnotách pak už docházelo ve výsledné datakrychli ke „spojení“ jednotlivých filtrů a vytvářeli se tak jednolitě plochy. Rekonstruované spektrum pro jednotlivé barevné filtry při tomto nastavení poměrně přesně odpovídalo naměřenému spektru.

4.3.2 Mód 2

Mód 2 je, co se týče přístupu, podobný módu 1, kde však namísto pouze vrchní části detektoru, využíváme detektor celý. Máme tedy k dispozici více informace, což



(a) Původní data rychle simulovaných dat



(b) Rekonstrukce pomocí módu 1

Obrázek 4.6: Simulovaná scéna s barevnými filtry osvětlená lampou

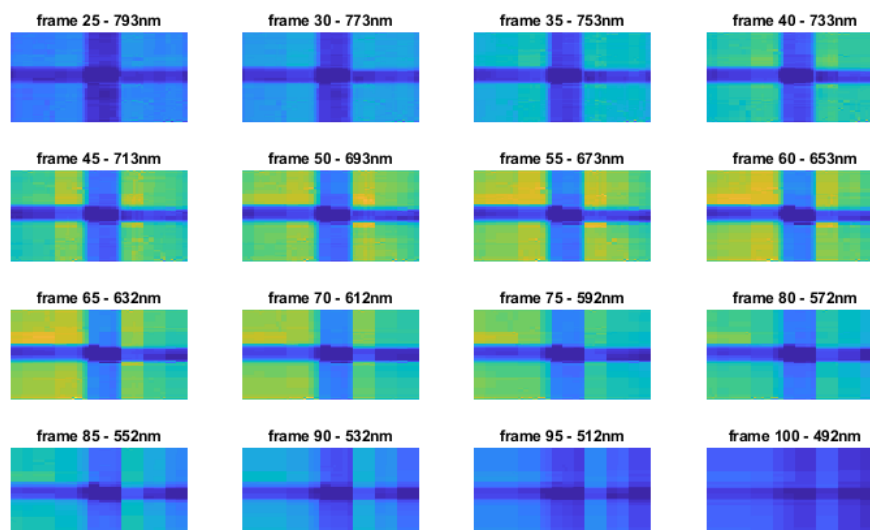
se projeví lepší rekonstrukcí scény. V některých případech u tohoto módu rekonstrukci zlepšuje jak parametr „RMinInitGuess“, tak i nahrazování záporných prvků (v tomto případě nastaveno na hodnotu „zero“, kde jsou tedy záporné prvky nahrazeny nulovými hodnotami). Stejně jako u módu 1 k rekonstrukci vždy přispívaly parametry „useZOinInitGuess“ a „useZOinCalc“.

Simulovaná data:

- **Scéna s křížem osvětlená laserem** - U těchto dat bylo pomocí módu 2 dosaženo velmi podobných výsledků jako u módu 1. Stejně jako u módu 1 lze lepších výsledků dosáhnout nižšími hodnotami regularizačního parametru (kolem hodnoty 500), čím vyšší je pak jeho hodnota, tím hůře je patrný kříž a pokud se parametr navýší na příliš vysoké hodnoty (3000 a více), algoritmus dává na výsledku pouze jednodlitou plochu bez kříže. Zapnutí nebo vypnutí parametru „RMinInitGuess“ pak nemělo na kvalitu této rekonstrukce žádný vliv. Při nastavení regularizačního parametru na hodnotu 500 jsem dosáhl podobnosti datakrychlí $4,8 \cdot 10^{-3}$, což je lepší výsledek než u módu 1.
- **Scéna s křížem osvětlená lampou** - Pro tento typ scény se ukázal mód 2 jako nejlepší. Pro správné zobrazení kříže zde bylo stejně jako u módu 1 nutné zapnout parametr „ZOmean“ a nastavit vysokou hodnotu regularizačního parametru (kolem 5000). Podobnost datakrychlí se pak pohybovala kolem $4,7 \cdot 10^{-4}$.
- **Scéna s barevnými filtry osvětlená lampou** - U těchto dat se nejlepších výsledků podařilo dosáhnout opět při nastavení regularizačního parametru na hodnotu 5000 a při použití parametrů „RMinInitGuess“ a „imEncZeroing“ nastaveného na hodnotu „zero“. V takovémto případě byla hodnota podobnosti datakrychlí $2,5 \cdot 10^{-4}$, což je pro tato data nejlepší výsledek, jakého se podařilo dosáhnout.
Zajímavé je, že při nastavení regularizačního parametru přibližně do hodnoty 2000, celkové rekonstrukci pomáhal i parametr „ZOmean“, nicméně po překročení hranice 2000 naopak jeho použití rekonstrukci zhoršovalo.

Reálná data:

- **Scéna s křížem osvětlená laserem** - Na tomto typu dat vykazoval mód 2 obecně lepší výsledky než mód 1, projevující se především užší špičkou intenzity v okolí vlnové délky, na které laser vyzařuje (to znamená, že se kříž zobrazoval na menším počtu vlnových délek). Nejlepší rekonstrukce se podařilo dosáhnout při použití parametrů „RMinInitGuess“ a „imEncZeroing“ (opět nastaveného na hodnotu „zero“) a nastavení regularizačního parametru na hodnotu přibližně 700. Parametr „ZOmean“ zde rekonstrukci zásadně zhoršoval.
- **Scéna s křížem osvětlená lampou** - Podobně jako u simulovaných dat byl pro tato data nejlepší mód 2 a to při zapnutí parametru „RMinInitGuess“



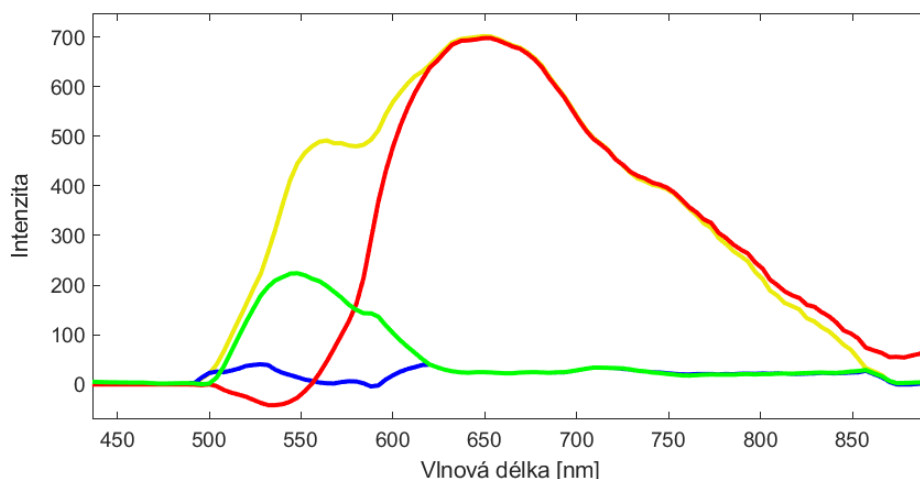
Obrázek 4.7: Scéna s křížem osvětlená halogenovou lampou - rekonstrukce pomocí módu 2

a „ZOmean“, který sice částečně zkresloval výsledné spektrum, nicméně se díky němu správně zobrazoval kříž téměř ve všech vlnových délkách. Ideální hodnota regularizačního parametru se ukázala kolem 2000, při vyšších hodnotách pak začínal být kříž hůře patrný. Několik řezů z rekonstruované data rychle je vidět na obrázku 4.7.

- **Scéna s barevnými filtry osvětlená lampou** - U těchto dat výsledný průběh spektra jednotlivých filtrů odpovídal naměřeným datům nejpřesněji z testovaných módů. Ideální hodnotou regularizačního parametru se zde ukázala hodnota kolem 1000, při vyšších hodnotách pak od sebe nešly jednotlivé filtry odlišit a výsledné spektrum bylo zkreslené. Použití parametru „RMinInit-Guess“ na rekonstrukci nemělo téměř žádný vliv a parametry pro nahrazení záporných prvků a „ZOmean“ zkreslovali výslednou intenzitu spektra. Rekonstrukci spektra pomocí této konfigurace je možné vidět na obrázku 4.8.

4.3.3 Mód 3

V módu 3 opět využíváme informaci z celého detektoru. Ve výsledku však algoritmus pracuje pouze s jednou maticí, která je složena z prvního řádu, který odpovídá odečteným prvním řádům z vrchní a spodní části, a z nultého řádu, který odpovídá sečteným nultým řádům. Pro všechny typy dat bylo nutné u tohoto módu využívat parametr „imEncZeroing“ nastavený na hodnotu „abs“ (nahrazení záporných prvků jejich absolutní hodnotou), bez kterého nebyl algoritmus schopen správně zobrazit

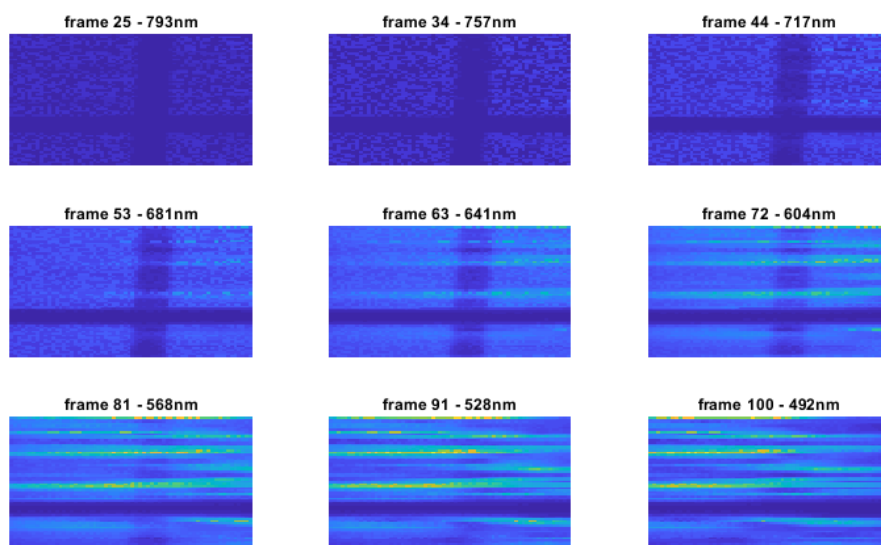


Obrázek 4.8: Rekonstruované spektrum scény s barevnými filtry osvětlené halogenovou lampou - rekonstrukce pomocí módu 2 (barvy křivek odpovídají barvám filtrů)

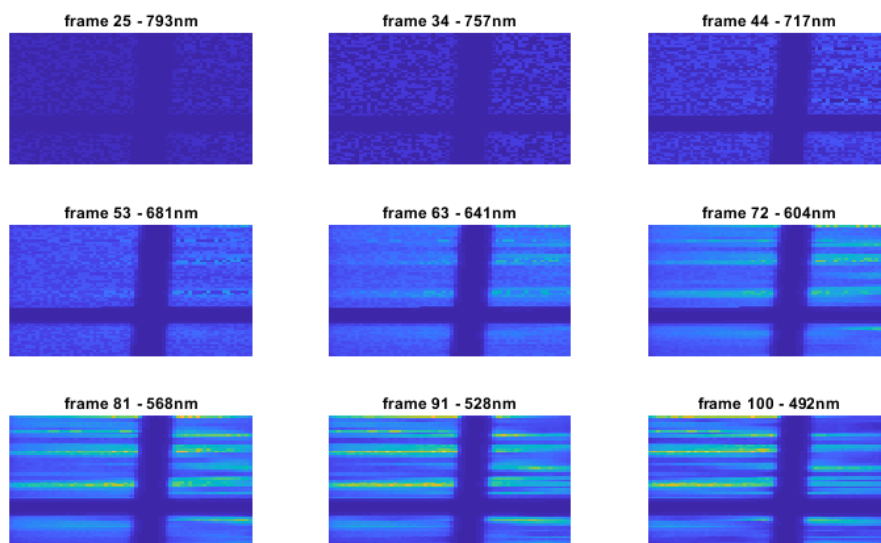
scény ani intenzitu spektra. Obdobně jako u dvou dříve zmíněných módů, i zde k rekonstrukci přispívaly parametry „useZOinInitGuess“ a „useZOinCalc“ a v některých případech i parametr „RMinInitGuess“.

Simulovaná data:

- **Scéna s křížem osvětlená laserem** - Pro laser a obecně pro rekonstrukci scén o několika vlnových délkách s výraznou intenzitou se mód 3 ukázal jako nejlepší. Konkrétně u těchto simulovaných dat se ve výsledné datakrychli kříž zobrazoval pouze na třech řezech datakrychle a na ostatních řezech byly nulové nebo téměř nulové hodnoty, což odpovídá simulovaným datům, kde byly nuluové hodnoty taktéž pouze na třech řezech datakrychle. Takovýchto výsledků jsem dosáhl při nastavení regularizačního parametru na hodnotu 1000 a při použití parametru „imEncZeroing“ nastaveného na hodnotu „abs“. V tomto případě byla podobnost datakrychlí $3,2 \cdot 10^{-3}$, což je podstatně méně než u předešlých dvou módů.
- **Scéna s křížem osvětlená lampou** - Obdobně jako u předešlých dvou módů, i zde bylo podmínkou správné rekonstrukce zapnutí parametru „ZOmean“. Porovnání vlivu tohoto parametru na rekonstrukci je možné vidět na obrázku 4.9. Nejlepších výsledků se podařilo dosáhnout při nastavení regularizačního parametru na hodnotu 5000 a použití parametru „imEncZeroing“. Výsledná podobnost datakrychlí byla $7,3 \cdot 10^{-4}$, což je výsledek horší než u obou předešlých módů. Na obrázku 4.10 je pak možné vidět porovnání rekonstruovaného spektra s původním simulovaným spektrem datakrychle.
- **Scéna s barevnými filtry osvětlená lampou** - U těchto dat se nepodařila za pomoci módu 3 provést rekonstrukci s dobrými výsledky.

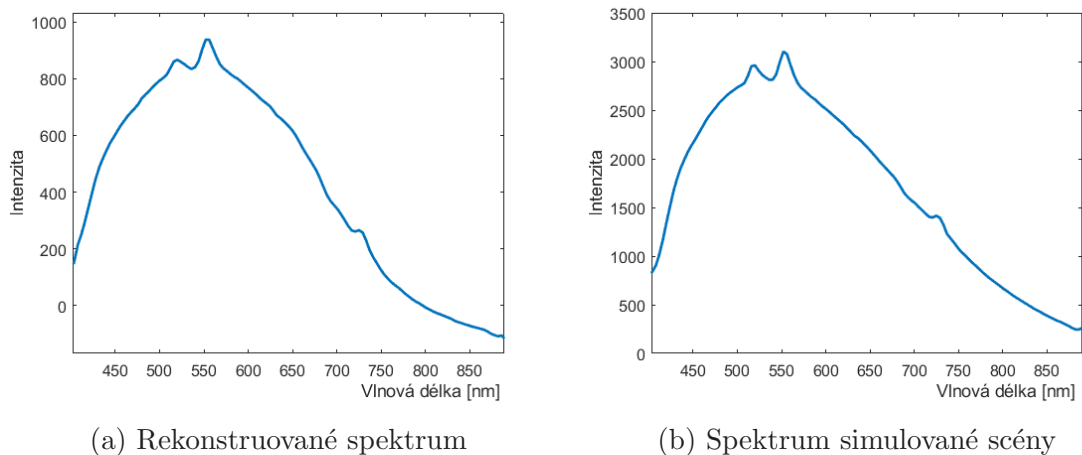


(a) Rekonstrukce bez parametru „ZOmean“ - podobnost datakrychlí $8,7 \cdot 10^{-4}$



(b) Rekonstrukce s parametrem „ZOmean“ - podobnost datakrychlí $7,3 \cdot 10^{-4}$

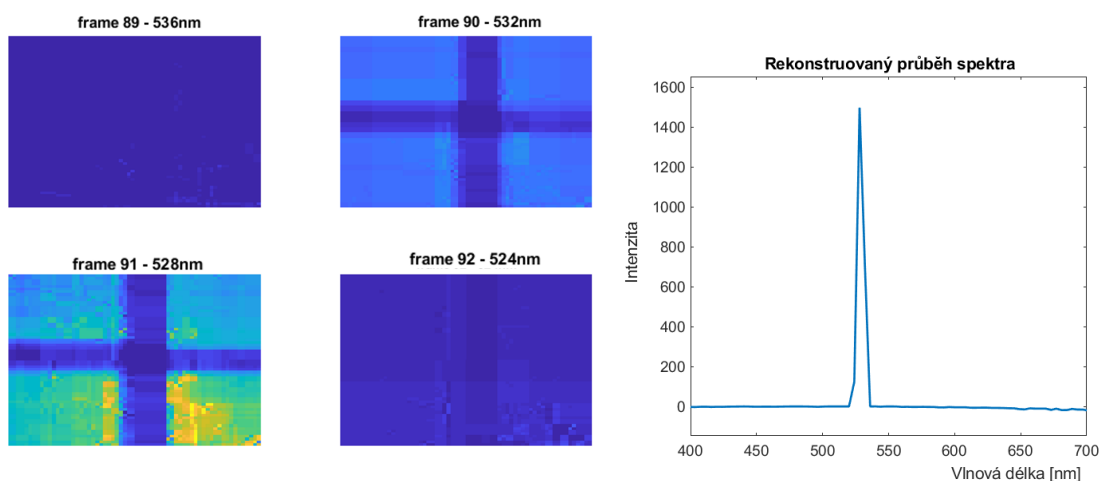
Obrázek 4.9: Ukázka rekonstrukce simulované scény (kříž osvětlený halogenovou lampou) pomocí módu 3 s vypnutým a zapnutým parametrem „ZOmean“



Obrázek 4.10: Spektrum simulované scény s křížem osvětlené lampou - rekonstrukce pomocí módu 3 (vlevo) a původní spektrum datakrychle (vpravo)

Reálná data:

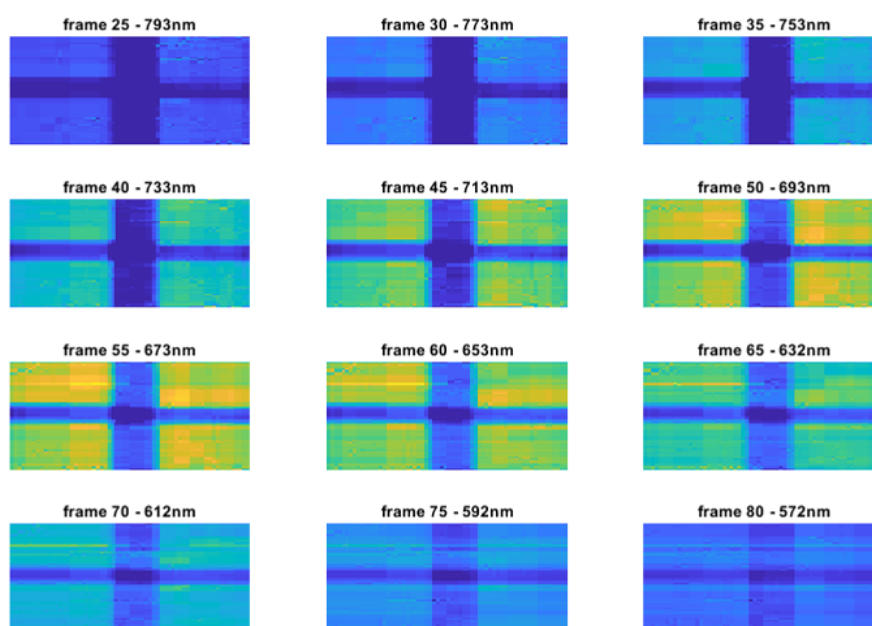
- Scéna s křížem osvětlená laserem** - Jak již bylo řečeno u simulovaných dat laseru, mód 3 se ukázal pro takovýto typ dat ideální. Nejlepších výsledků se podařilo dosáhnout obdobně jako u simulovaných dat za použití „imEncZeroing“ a nastavení regularizačního parametru na hodnotu kolem 400. V tomto případě se kříž zobrazoval pouze na dvou řezech datakrychle (vlnové délky 528 nm a 532 nm), což přibližně koresponduje s vlnovou délkou, na které laser vyzařuje. Několik řezů z výsledné datakrychle a průběh spektra je možné vidět na obrázku 4.11.
- Scéna s křížem osvětlená lampou** - U těchto dat jsem zprvu očekával, že ke správnému zobrazení kříže bude opět nutné použít parametr „ZOMean“. Nicméně se ukázalo, že mód 3 vykazuje poměrně dobré výsledky i bez využití tohoto parametru a jeho zapnutí na rekonstrukci nemá příliš velký vliv. Hlavním problémem této rekonstrukce (a obecně rekonstrukce reálných scén osvětlených lampou pomocí tohoto módu) bylo posunutí maxima intenzity k vyšším vlnovým délkám, než co jsme naměřili spektrometrem. Nejlepších výsledků se pak dosáhlo při nastavení regularizačního parametru na hodnotu 600 a zapnutí parametrů „imEncZeroing“ a „RMinInitGuess“. Několik řezů z rekonstruované datakrychle je možné vidět na obrázku 4.12.
- Scéna s barevnými filtry osvětlená lampou** - Jak se ukázalo na simulovaných datech, tento mód není příliš vhodný pro podobný typ scény. Sice jsem zde byl schopen dosáhnout lepších výsledků, než při testování na simulovaných datech, nicméně spektrum jednotlivých filtrů zde bylo stále velmi zkreslené a neodpovídalo změřenému spektru. Nejlepších výsledků se dosáhlo při použití parametru „imEncZeroing“ a nastavení regularizačního parametru k hodnotám kolem 1000.



(a) Vybrané řezy z datakrychle

(b) Rekonstruované spektrum laseru

Obrázek 4.11: Ukázka scény s křížem osvětlené zeleným laserem rekonstruované pomocí módu 3



Obrázek 4.12: Scéna s křížem osvětlená halogenovou lampou - rekonstrukce pomocí módu 3 bez zapnutého parametru „ZOmean“

Tabulka 4.1: Přehled jednotlivých simulovaných scén a módů s nejlepšími podobnostmi datakrychlí, jaké se u nich podařilo při rekonstrukci dosáhnout

Simulovaná scéna	Dosažená podobnost datakrychlí		
	Mód 1	Mód 2	Mód 3
Kříž osvětlený laserem	$5 \cdot 10^{-3}$	$4,8 \cdot 10^{-3}$	$3,2 \cdot 10^{-3}$
Kříž osvětlený lampou	$6,9 \cdot 10^{-4}$	$4,7 \cdot 10^{-4}$	$7,3 \cdot 10^{-4}$
Barevné filtry osvětlené lampou	$2,6 \cdot 10^{-4}$	$2,5 \cdot 10^{-4}$	-

4.3.4 Shrnutí

V průběhu testování se ukázalo, že není možné zvolit ideální kombinaci parametrů, která by u všech scén a módů vykazovala vždy nejlepší výsledky. Jako takový kompromis, kdy je algoritmus schopný rekonstruovat poměrně dobře všechny testované scény za použití jakéhokoliv módu, se ukázala kombinace těchto parametrů - zapnuté parametry „useZOinInitGuess“, „useZOinCalc“ a regularizační parametr nastavený na hodnotu 400. Ostatní testované parametry pak bylo obecně lepší nechat vypnuté, protože pro určité scény a módy sice k rekonstrukci přispívaly, nicméně pro jiné naopak rekonstrukci značně zhoršovaly. Výjimkou pak byl mód 3, kde bylo nutné mít parametr „imEncZeroing“ nastavený na hodnotu „abs“ (nahrazení záporných prvků jejich absolutní hodnotou), jinak nebyl algoritmus schopen provádět příliš dobré rekonstrukce, tato hodnota byla později u módu 3 nastavena jako fixní. V případě, že bylo nutné zlepšit prostorovou informaci se pak vyplatilo zapnout parametr „ZO-mean“, díky kterému však docházelo ke zkreslení spektra.

V tabulce 4.2 je možné vidět přehled jednotlivých scén s nejlepší volbou přístupů a parametrů, v tabulce 4.1 je pak přehled simulovaných scén a módů s nejlepšími podobnostmi datakrychlí, jaké se u nich podařilo dosáhnout. Jak bylo řečeno na začátku této kapitoly, kvalita rekonstrukce se hodnotila u simulovaných dat podle podobnosti datakrychlí (hodnota představující minimum funkce 4.1) a u reálných dat subjektivně z porovnání průběhu intenzity spektra dané scény a podle vizuálního vzhledu scény na jednotlivých vlnových délkách.

Tabulka 4.2: Přehled jednotlivých scén s parametry rekonstrukce, které se pro danou scénu nejvíce osvědčily (λ je regularizační parametr a parametry, které nejsou v tabulce u jednotlivých scén uvedeny, jsou u dané konfigurace vypnuté)

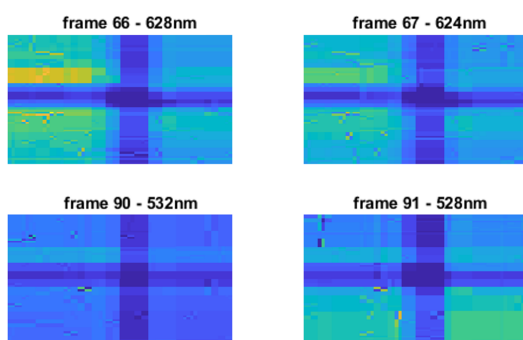
Scéna	Parametry	
	Simulovaná data	Reálná data
Kříž osvětlený laserem	Mód 3 $\lambda = 1000$ useZOinInitGuess useZOinCalc imEncZeroing = abs	Mód 3 $\lambda = 400$ useZOinInitGuess useZOinCalc imEncZeroing = abs
Kříž osvětlený lampou	Mód 2 $\lambda = 5000$ useZOinInitGuess useZOinCalc RMinlitGuess ZOmean	Mód 2 $\lambda = 2000$ useZOinInitGuess useZOinCalc RMinlitGuess ZOmean
Barevné filtry osvětlené lampou	Mód 2 $\lambda = 5000$ useZOinInitGuess useZOinCalc RMinlitGuess imEncZeroing = zero	Mód 2 $\lambda = 1000$ useZOinInitGuess useZOinCalc

4.4 Další pokusná měření

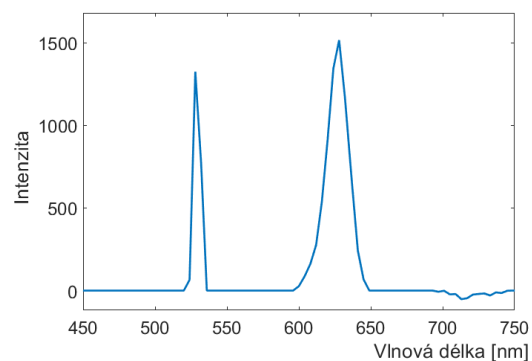
Kromě dříve zmíněných scén se pokusná měření dělala také na dalších reálných scénách. K zajímavějším patří například rekonstrukce scény osvětlené zářivkou nebo rekonstrukce scény s červenou diodou a zeleným laserem. Jak již bylo řečeno v předešlé kapitole, k rekonstrukci scén s několika výraznými vlnovými délkami se jako nejlepší ukázal mód 3 a na těchto datech se tento závěr jen potvrdil.

Na obrázku 4.14 lze vidět několik řezů datakrychle (s příslušnými vlnovými délkami) a rekonstruované spektrum scény osvětlené červenou diodou a zeleným laserem. Je zde opět patrné, že se scéna zobrazuje pouze na několika řezech datakrychle (u diody je jich více, což je způsobeno tím, že dioda vyzařuje v širším spektru než laser), což svědčí o poměrně přesné spektrální rekonstrukci. Zároveň si také můžeme ověřit, že vlnová délka skutečně odpovídá přibližně zelené a červené barvě. Co se týče prostorové rekonstrukce, opět se jednalo o scénu s křížem, který je zde taktéž dobře patrný.

Na obrázku 4.14a je vidět rekonstruované spektrum zářivky (rekonstrukce opět pomocí módu 3). Pro porovnání je zde také obrázek 4.14b, kde je spektrum změřené pomocí spektrometru Ocean Optics Flame.

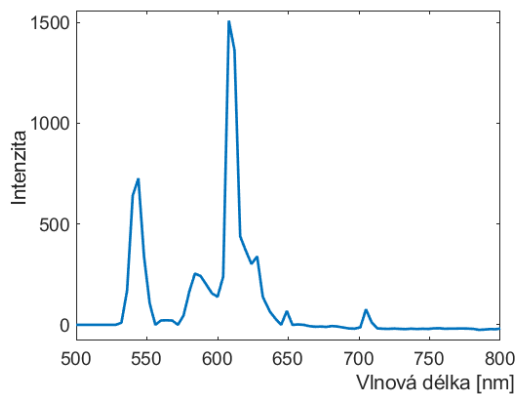


(a) Vybrané řezy z datakrychle

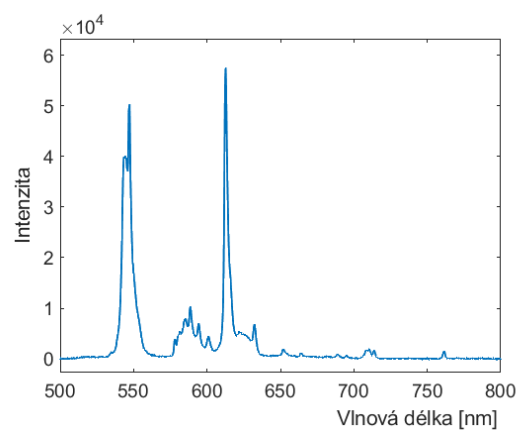


(b) Rekonstruované spektrum

Obrázek 4.13: Scéna s křížem osvětlená zeleným laserem a červenou diodou - rekonstrukce pomocí módu 3



(a) Rekonstruované spektrum



(b) Změřené spektrum

Obrázek 4.14: Scéna osvětlená zářivkou - rekonstrukce spektra pomocí módu 3 (vlevo) a spektrum změřené spektrometrem (vpravo)

5 Závěr

Cílem této práce bylo zkoumat metodu CASSI z hlediska schopnosti rekonstrukce různých scén a za použití různých přístupů k rekonstrukci. V průběhu měření jsme identifikovali několik problémů týkajících se samotné zobrazovací soustavy. Část těchto problémů byla nejspíše způsobena odstraněním vstupní části optické soustavy sloužící jako objektiv a jejím nahrazením dvojčočkou a šterbinou, které jsou pro testovací účely naprosto dostačující, nicméně nemohou se vyrovnat optické soustavě o pěti čočkách. Zbytek byl pravděpodobně způsoben nedokonalostmi samotné soustavy, případně vnějšími vlivy (například budící dioda u laseru).

Celkově není možné určit, jaké parametry nebo přístupy by byly nejlepší, neboť jak bylo ukázáno na pokusných měřeních, každý přístup má své výhody a nevýhody podle scény, kterou zpracováváme. Pokud však o scéně nemáme žádnou informaci předem, doporučená konfigurace, která se poměrně dobře osvědčila na všech testovaných scénách, je použití módu 2 (přístup, ve kterém pracujeme se dvěma maticemi reprezentujícími vrchní a spodní část detektoru) s hodnotou regularizačního parametru 400 a za použití nultého řádu jak v počátečním odhadu (parametr „useZOinInitGuess“), tak i v průběhu rekonstrukce (parametr „useZOinCalc“). V případě, že budeme potřebovat zlepšit prostorovou informaci se obecně vyplatí zapnout u tohoto módu i parametr „ZOmean“, díky kterému jsou jednotlivé řezy data rychle v průběhu rekonstrukce přenášeny nultým řádem, nicméně v takovém případě musíme počítat se zkreslením spektrální informace. Pokud však víme, že spektrum scény bude pravděpodobně složené jen z několika vlnových délek s vysokou intenzitou, vyplatí se využít módu 1 (přístup ve kterém pracujeme s jednou maticí složenou z odečtených prvních řádů a sečtených nultých řádů).

Hlavním parametrem, který pak výrazně ovlivňoval kvalitu rekonstrukce, byl regularizační parametr. U jednodušších dat, kde jsou převážně větší jednolitě plochy o stejné intenzitě (jako například u námi simulovaných dat) se osvědčilo tento parametr nastavit na vysokou hodnotu (5000 a více). Je to dáno tím, že v takovémto případě se snaží algoritmus TwIST dosáhnout takového výsledku, kde bude ve vlnkové doméně co nejméně nenulových prvků. Takovýmto malým počtem prvků jsme pak schopni tato jednodušší data dobře reprezentovat, nicméně ve chvíli, kdy jsou data, co se týče prostorové informace, složitější, je potřeba více nenulových prvků pro jejich reprezentaci a tudíž je nutné snížit i regularizační parametr.

Při měřeních jsem narazil na několik limitujících faktorů týkajících se snímaných scén. Algoritmus není schopen správně rekonstruovat spektrum, ve kterém je několik špiček intenzit vedle sebe, což je například vidět u rekonstruovaného spektra zářivky, kdy v naměřeném spektru můžeme vidět v rozmezí 550 nm až 600 nm několik

výraznějších špiček intenzit, nicméně v rekonstruovaném spektru jednotlivé špičky „splývají“ a již nejsou patrné. Je to dáno zaprvé samotným principem metody, kdy se na detektoru jednotlivé vlnové délky částečně překrývají a zadruhé je to dáno spektrálním rozlišením, které je v našem případě přibližně 4 nm. Dále se také ukázalo, že je rekonstrukce obecně citlivější na změnu prostorové informace v ose y než na změnu v ose x (to znamená, že se nám podaří lépe rekonstruovat horizontální čáru, nežli vertikální), což je možné pozorovat kupříkladu u scény s křížem. Opět je to dáno překrýváním snímků jednotlivých vlnových délek.

Celkově však lze konstatovat, že algoritmus byl na námi testovaných datech poměrně úspěšný a při použití vhodné kombinace parametrů byl schopen přesné rekonstrukce jak spektrální tak prostorové informace.

Zdroje

- [1] Jíří Hlubuček a Karel Židek. “Hyperspectral imaging in infrared region using compressed sensing methods”. In: *ACC Journal* 24.1 (2018-6-30), s. 24–32. ISSN: 18039782. DOI: [10.15240/tul/004/2018-1-003](https://doi.org/10.15240/tul/004/2018-1-003).
- [2] Marena Manley. “Near-infrared spectroscopy and hyperspectral imaging: non-destructive analysis of biological materials”. In: *Chem. Soc. Rev.* 43 (24 2014), s. 8200–8214. DOI: [10.1039/C4CS00062E](https://doi.org/10.1039/C4CS00062E). URL: <http://dx.doi.org/10.1039/C4CS00062E>.
- [3] Peg Shippert. *Push Broom and Whisk Broom Sensors*. © 2020. URL: <https://www.harrisgeospatial.com/Support/Self-Help-Tools/Help-Articles/Help-Articles-Detail/ArtMID/10220/ArticleID/16262/Push-Broom-and-Whisk-Broom-Sensors> (cit. 23.01.2020).
- [4] Bart M. Nicolai et al. “Non-destructive measurement of bitter pit in apple fruit using NIR hyperspectral imaging”. In: *Postharvest Biology and Technology* 40.1 (2006), s. 1–6. ISSN: 0925-5214. DOI: <https://doi.org/10.1016/j.postharvbio.2005.12.006>. URL: <http://www.sciencedirect.com/science/article/pii/S0925521405002723>.
- [5] Hamed Akbari et al. “Hyperspectral imaging and quantitative analysis for prostate cancer detection”. In: *Journal of Biomedical Optics* 17.7 (2012), s. 1–11. DOI: [10.1117/1.JBO.17.7.076005](https://doi.org/10.1117/1.JBO.17.7.076005). URL: <https://doi.org/10.1117/1.JBO.17.7.076005>.
- [6] Neelam Gupta. “Hyperspectral imager development at Army Research Laboratory”. In: *Infrared Technology and Applications XXXIV*. Ed. Bjørn F. Andresen, Gabor F. Fulop a Paul R. Norton. Sv. 6940. International Society for Optics a Photonics. SPIE, 2008, s. 573–582. DOI: [10.1117/12.777110](https://doi.org/10.1117/12.777110). URL: <https://doi.org/10.1117/12.777110>.
- [7] Siegfried Wartewig a Reinhard H.H. Neubert. “Pharmaceutical applications of Mid-IR and Raman spectroscopy”. In: *Advanced Drug Delivery Reviews* 57.8 (2005). Non-Invasive Spectroscopic and Imaging Techniques in Drug Delivery, s. 1144–1170. ISSN: 0169-409X. DOI: <https://doi.org/10.1016/j.addr.2005.01.022>. URL: <http://www.sciencedirect.com/science/article/pii/S0169409X0500058X>.
- [8] Bahaa E. A. Saleh a Malvin Carl Teich. *Základy fotoniky*. Vyd. 1. Praha: Matfyzpress, 1995. ISBN: 80-85863-05-7.

- [9] Milan Šonka, Václav Hlaváč a Roger Boyle. *Image processing, analysis, and machine vision*. 3rd ed. Toronto: Thompson Learning, c2008. ISBN: 978-049-5082-521.
- [10] Adrian Stern. *Optical compressive imaging*. 1st Edition. Boca Raton: CRC Press, Taylor & Francis Group, [2017]. ISBN: 978-1-4987-0806-7.
- [11] Gonzalo R. Arce et al. “Compressive Coded Aperture Spectral Imaging. An Introduction”. In: *IEEE Signal Processing Magazine* 31.1 (2014), s. 105–115. ISSN: 1053-5888. DOI: [10.1109/MSP.2013.2278763](https://doi.org/10.1109/MSP.2013.2278763). URL: <http://ieeexplore.ieee.org/document/6678264/>.
- [12] Pavel Rajmíc a Marie Daňková. *Úvod do řídkých reprezentací signálů a komprimovaného snímání*. první. Brno: Vysoké učení technické v Brně, 2014. ISBN: 978-80-214-5169-8. URL: <http://www.utko.feec.vutbr.cz/~rajmic/skripta/Skripta-Ridke-reprezentace-Rajmic-Dankova.pdf> (cit. 22. 03. 2020).
- [13] D. Angelosante, G. B. Giannakis a E. Grossi. “Compressed sensing of time-varying signals”. In: *2009 16th International Conference on Digital Signal Processing* (2009), s. 1–8. DOI: [10.1109/ICDSP.2009.5201168](https://doi.org/10.1109/ICDSP.2009.5201168). URL: <http://ieeexplore.ieee.org/document/5201168/> (cit. 02. 03. 2020).
- [14] Ashwin Wagadarikar et al. “Single disperser design for coded aperture snapshot spectral imaging”. In: *Appl. Opt.* 47.10 (2008), B44–B51. DOI: [10.1364/AO.47.000B44](https://doi.org/10.1364/AO.47.000B44). URL: <http://ao.osa.org/abstract.cfm?URI=ao-47-10-B44>.
- [15] Elena Anisimova, Jan Bednář a Petr Páta. “Zpracování obrazu pomocí vlnkové transformace”. In: *Elektrorevue* 15.4 (srpen 2013), s. 238–246. ISSN: 1213-1539. URL: <http://www.elektrorevue.cz/cz/download/zpracovani-obrazu-pomoci-vlnkove-transformace--image-processing-using-the-wavelet-transform-/> (cit. 16. 05. 2020).
- [16] Gerardo Gamez. “Compressed sensing in spectroscopy for chemical analysis”. In: *Journal of Analytical Atomic Spectrometry* 31.11 (2016), s. 2165–2174. ISSN: 0267-9477. DOI: [10.1039/C6JA00262E](https://doi.org/10.1039/C6JA00262E). URL: <http://xlink.rsc.org/?DOI=C6JA00262E> (cit. 12. 04. 2020).
- [17] M. F. Duarte et al. “Single-pixel imaging via compressive sampling”. In: *IEEE Signal Processing Magazine* 25.2 (2008), s. 83–91. ISSN: 1558-0792.
- [18] Dana Mackenzie. *What’s Happening in the Mathematical Sciences, Volume 7*. 1st Edition. Providence, Rhode Island, USA: American Mathematical Society, 2008. ISBN: 978-0-8218-4478-6.
- [19] L. Jacques et al. “CMOS compressed imaging by Random Convolution”. In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing* 2009.CFP09ICA (2009), s. 1113–1116. ISSN: 1520-6149. DOI: [10.1109/ICASSP.2009.4959783](https://doi.org/10.1109/ICASSP.2009.4959783). URL: <http://ieeexplore.ieee.org/document/4959783/> (cit. 13. 04. 2020).

- [20] Karel Židek et al. “Compact and robust hyperspectral camera based on compressed sensing”. In: *Optics and Measurement International Conference 2016*. Ed. Jana Kovacicinova. Sv. 10151. International Society for Optics a Photonics. SPIE, 2016, s. 165 –171. DOI: [10.1117/12.2250268](https://doi.org/10.1117/12.2250268). URL: <https://doi.org/10.1117/12.2250268>.
- [21] *G-507*. © 2020. URL: <https://www.alliedvision.com/en/products/cameras/detail/Manta/G-507.html> (cit. 22.05.2020).