



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

# Zpracování hyperspektrálních dat pomocí neuronových sítí na FPGA

## Autoreferát disertační práce

Studijní program: P2612 Elektrotechnika a informatika  
Studijní obor: 2612V045 Technická kybernetika

Doktorand: **Ing. Jiří Čech**  
Školitel: **Ing. Martin Rozkovec, Ph.D.**  
ITE FM TUL



## **Abstrakt**

Cílem práce bylo navrhnout a natrénovat konvoluční neuronové sítě pro vyhodnocování hyperspektrálních snímků, ověřit jejich vlastnosti oproti ostatním metodám a realizovat jejich výpočet na FPGA. Pro zjištění reálných vlastností hyperspektrálních snímků, byla nejprve dostupná hyperspektrální kamera podrobně zkoumána a jednotlivé vlastnosti řádně identifikovány.

## **Klíčová slova**

Hyperspektrální záznam, Konvoluční neuronové sítě, Kalibrace hyperspektrální kamery, FPGA, LWIR

## **Abstract**

The aim of this work was to design and train convolutional neural networks for the evaluation of hyperspectral images, verify their properties against other existing methods and implement their inference on FPGAs. To find out the real properties of hyperspectral images, firstly the available hyperspectral camera was investigated in detail and the different properties were properly identified.

## **Key words**

Hyperspectral imaging, Convolutional neural networks, Calibration of hyperspectral camera FPGA, LWIR

## **Poděkování**

Tato práce byla podpořena Studentskou grantovou soutěží Technické univerzity v Liberci v rámci projektu č. SGS-2020-3059.

## Obsah

Úvod.....	5
1 Hyperspektrální snímání.....	8
1.1 Spektrální charakteristika.....	8
1.2 Záznam scény.....	8
1.3 Uplatnění hyperspektrálních systémů.....	10
2 Hyperspektrální kamerový systém (RODES).....	12
2.1 Kalibrace systému.....	14
2.1.1 Kalibrace infračervené kamery.....	14
2.1.2 Korekce vadných pixelů.....	17
2.1.3 Zaostření kamery.....	17
2.1.4 Identifikace rozložení spektra na senzoru.....	19
2.2 Identifikace šumu systému.....	22
2.2.1 Identifikace šumu hyperspektrální kamery.....	23
2.3 Naměřené vzorky.....	25
3 Vyhodnocení hyperspektrální kostky.....	27
3.1 Vyhodnocované vzorky.....	27
3.1.1 Testovací vzorek „Pavia University“.....	27
3.1.2 Testovací vzorek „Indian pines“.....	28
3.2 Vyhodnocovací algoritmy.....	28
3.2.1 Redukce spektrální charakteristiky.....	31
3.3 Neuronové sítě.....	31
3.3.1 Redukce velikosti neuronové sítě.....	32
3.4 Porovnání dosažených přesností.....	33
3.4.1 Vytvořené testovací množiny.....	33
3.4.2 Vliv hloubky neuronové sítě.....	34
3.4.3 Vliv redukce spektrální charakteristiky.....	35

3.4.4	Vliv zahrnutí okolí pixelu .....	35
3.4.5	Citlivost vyhodnocení na šum vstupních dat .....	36
4	Porovnání výpočetních platforem.....	37
4.1	Příprava FPGA .....	37
4.2	Optimalizace neuronových sítí.....	39
4.3	Výkonnost platforem.....	40
4.4	Nároky systému RODES.....	42
	Závěr .....	43
	Citovaná literatura.....	46
	Seznam publikací .....	49
	Životopis .....	50

## Úvod

Technologický rozmach průmyslu a automatizace vyžaduje pro potřeby efektivního řízení stále více vstupních dat. To vede k intenzivnímu výzkumu senzorů, miniaturizaci řídicích systémů a optimalizaci vyhodnocovacích algoritmů. Stávající desktopová řešení se převádějí na úspornější a efektivnější „Cloud“ nebo „Edge“ řešení, tak aby se zkrátila latence systému nebo navýšil dostupný výpočetní výkon. Nejinak je tomu i u kamerových systémů v průmyslu nebo u bezpečnostních aplikací.

Běžné řešení kamerového systému se skládá z jedné až osmi kamer připojených k jednomu počítači, na kterém je puštěna příslušná řídicí aplikace. Moderní řešení propojují kamerový systém s ostatními přidruženými systémy. Například kamerový systém zaznamenaný obraz pouze komprimuje a odesílá do cloudového úložiště, kde se snímky archivují. Dále do stejného úložiště má přístup vyhodnocovací systém, pracující na výkonném HW, který uložené snímky zpracuje a vyhodnotí jejich kvalitu a další příznaky potřebné pro danou aplikaci. Omezením takovéto aplikace je pak zejména datová propustnost propojení mezi systémy, způsobená zvyšujícím se dostupným rozlišením a rychlostí senzorů. To je možné snížit využitím tzv. inteligentních kamer, které umožňují vyhodnocení snímku přímo v kameře [1] [2]. Může se jednat o vyhledávání objektů v obraze a následnému ořezu odesílaného snímku o nezajímavé oblasti, až po úplné vyhodnocení snímku a odeslání pouze vypočtených příznaků.

Ve své práci se zaměřuji na Hyperspektrální kamerové systémy, které narozdíl od klasických barevných kamer zaznamenávají scénu ve třech až čtyřech barevných složkách (přes Bayerovský filtr), umožňují záznam scény v řádu stovek „barevných“ pásem, čímž diametrálně narůstá zaznamenaný objem dat. Jednotlivé snímky zaznamenávané scény se skládají do struktury označované jako „Hyperspektrální kostka“ a ta je následně odesílána do nadřazeného systému k vyhodnocení. Díky získané podrobnější „barevné“ informaci lze lépe detekovat objekty a jejich vlastnosti, čehož se hojně využívá v průmyslu a farmaceutických procesech [3].

Cílem mé práce je upravit proces vyhodnocení hyperspektrálních snímků tak, aby byl realizovatelný v řídicím systému hyperspektrální kamery, obdobně jak je tomu u inteligentních černobílých a barevných kamer. Využitý kamerový systém je složen ze senzoru a programovatelného hradlového pole (FPGA), pro které hledám vhodný způsob vyhodnocení hyperspektrálních snímků a ověřuji jeho vlastnosti oproti stávajícímu řešení využívající výkonnou výpočetní platformu mimo kameru. Díky tomu může být výsledný systém kompaktní

a úsporný, což má příznivý vliv na jeho provozní vlastnosti, především při jeho používání v terénu.

Primární účel využitého hyperspektrálního kamerového systému je pro detekci nebezpečných hořlavých plynných látek [4]. To je možné díky použití bolometrických senzorů schopných detekovat dlouhovlnné infračervené záření (long wave infrared, LWIR). Tzv. termovizní systémy jsou monochromatické kamerové systémy schopné detekovat tepelné zdroje bez nutného osvětlení scény. Jejich nevýhodami jsou vysoký šum systému, vysoká teplotní závislost jednotlivých bodů senzoru a obtížná kalibrace systému.

Při vyhodnocování hyperspektrálních kostek se v současné době porovnávají naměřené spektrální charakteristiky jednotlivých bodů v obraze s množinou předem definovaných tříd pomocí několika metrik (Euklidovská vzdálenost, spektrální úhel), jejichž vzorové představitele se hledají nebo upřesňují pomocí metody K-mean [5] [6]. Tento přístup dává dobré výsledky u laboratorně měřených dat, ale nikoliv u reálných měření, které jsou zatíženy vyšším šumem a proměnným pozadím i pracovním prostředím. Rozšíření vyhodnocovacích algoritmů o metody redukce šumu a přidání korekcí pro jednotlivá pracovní prostředí významně navyšuje výpočetní náročnost.

Trendem posledních let je využití neuronových sítí, které na základě natrénovaného modelu třídí vstupní spektrální charakteristiku mezi naučené třídy. Jejich trénování je časově náročné a vyžaduje velké množství obtížně získatelných trénovacích vzorků. Výhodou je i možné učení sítě na surových výstupech ze senzoru, čímž dojde k začlenění početných korekcí kamery, které jsou jinak nutné počítat ještě před začátkem vyhodnocování. Pomineme-li zdoluhavé trénování neuronových sítí, je výsledná inferenze hyperspektrálních kostek rychlejší, přesnější a lépe implementovatelná na různých výpočetních platformách (CPU, GPU nebo FPGA), než původně používané algoritmy [7].

Porovnávání jednotlivých metod a výkonnosti platforem provádím na veřejně dostupných a ověřených vzorcích pořízených ve viditelné (visible, VIS) části spektra, které jsou již po všech korekcích obrazu kamery a nejsou zatíženy výrazným šumem a teplotními změnami. Pro adaptaci vybraných metod otestovaných pro VIS vzorky je potřeba vytvořit obdobnou testovací sadu s dostupnou LWIR hyperspektrální kamerou. To obnáší tvorbu záznamového cyklu kamery, analýzu vlastností kamery a jejich chování při změně okolního prostředí, kalibraci kamery a stanovení vhodných provozních oblastí. Spektrální charakteristiky v naměřené hyperspektrální kostce je potřeba převést na reálné fyzikální jednotky, tak aby byly ověřitelné vůči přesným měřením spektrometry, které jsou dostupné ve veřejné databázi [8].

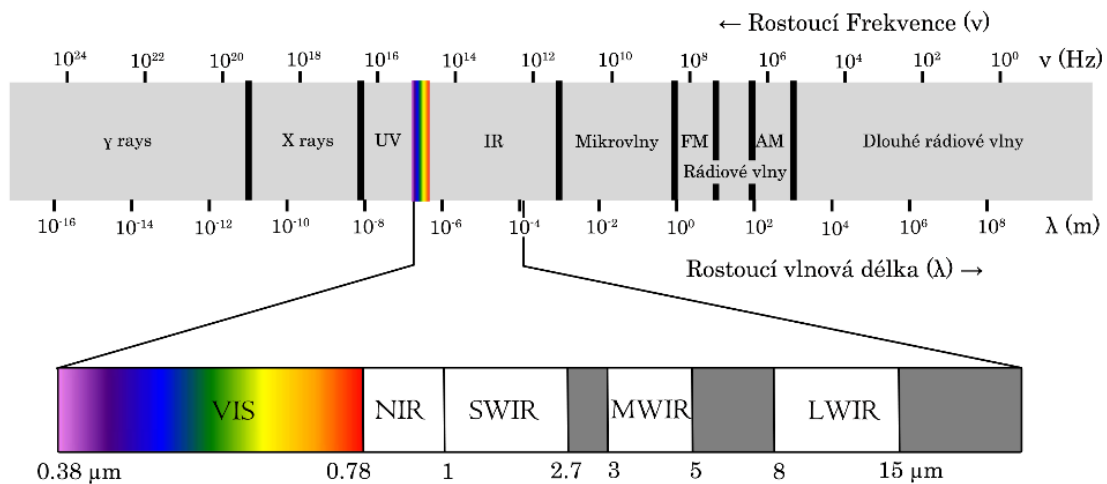
Všechny výpočty realizuji v prostředí Python s přidaným modulem pro výpočet neuronových sítí Tensorflow, který je podporovaný všemi testovanými platformami. Řešení na FPGA využívá přidaného výpočetního jádra (Deep Learning Processor Unit, DPU) [9]. V průběhu práce jsem přešel na novější nástroj Vitis-AI, který jsem testoval v několika konfiguracích. To obnáší realizaci celého procesu tvorby návrhu FPGA, integraci operačního systému PetaLinux [10] a převedení natrénovaných neuronových sítí do formátu potřebného pro výpočet na DPU.

# 1 Hyperspektrální snímání

## 1.1 Spektrální charakteristika

Kamerové systémy obsahují senzor schopný detekovat fotony, tj. kvanta elektromagnetické energie. Charakteristickou vlastností fotonu je jeho vlnová délka  $\lambda$  [m], udávaná také jako frekvence  $f$  [Hz] nebo vlnové číslo  $k$  [ $\text{m}^{-1}$ ] (vlnočet).

Podle vlnové délky rozlišujeme elektromagnetické záření do několika skupin, viz Obrázek 1. Kamerové systémy nejčastěji využívají části spektra: ultrafialové (ultraviolet, UV), viditelné (visible, VIS) a infračervené (infrared, IR). Pásma IR se dále dělí na krátko vlnné (near infrared, NIR) a dlouho vlnné (long wave infrared, LWIR).

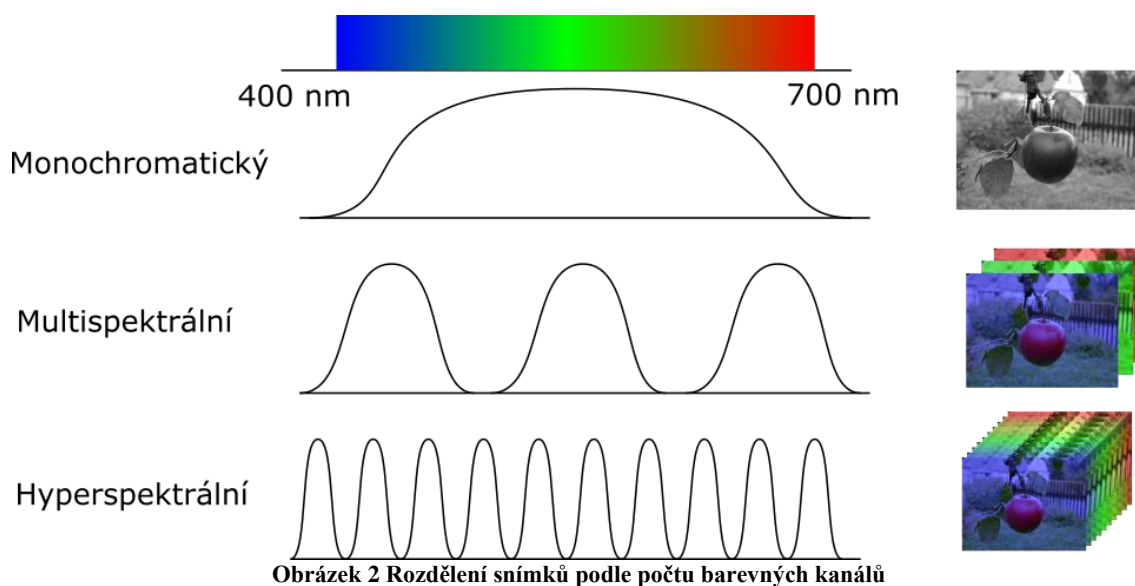


Obrázek 1 Elektromagnetické spektrum

## 1.2 Záznam scény

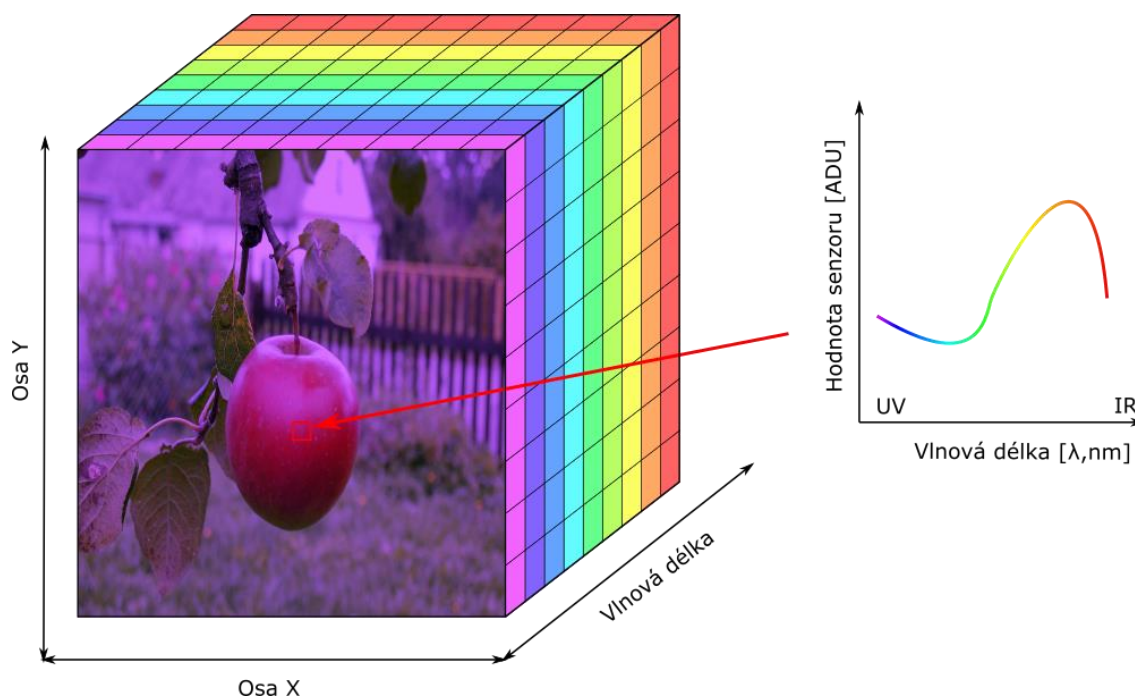
Senzor kamery pomocí fotoelektrického jevu převádí dopadající elektromagnetické záření na elektrický signál. Citlivost senzoru je vysoká jen v úzké části spektra a udává se citlivostní křivkou (quantum efficiency, QE). Výstupní snímek ze senzoru odpovídá zaznamenaným intenzitám ve spektrálním rozsahu citlivosti senzoru a je označován jako monochromatický. Přidáním úzkopásmových filtrů lze rozdělit spektrální rozsah na více částí, viz Obrázek 2, a pořízením snímku pro každou část vytvořit tzv. Hyperspektrální kostku. Použití takového filtru ale snižuje celkovou energii dopadající na pixel a snižuje tak odstup signálu od šumu (SNR), což klade vyšší požadavky na citlivost senzoru a osvětlení scény.





Podle konstrukce hyperspektrálního systému se liší rozložení spekter na senzoru a metody kompletování hyperspektrální kostky při záznamu scény. Využívá se přitom náročně výrobitelných optických elementů jako je optická mřížka, lineární optický filtr či mikroskopické čočkové pole.

Hyperspektrální kostka obsahuje informace o velikosti scény (dimenze X a Y) a o spektrálních vlastnostech scény (dimenze Z), viz Obrázek 3. Nehledě na principu záznamu scény je potřeba jednotlivé pořizené snímky aproximovat do jednotného prostorového a spektrálního rastru.



Obrázek 3 Zobrazení hyperspektrální kostky

### 1.3 Uplatnění hyperspektrálních systémů

Pro většinu průmyslových úloh rozpoznání a vyhodnocení výrobku postačují monochromatické, popřípadě barevné kamery s vhodným osvětlením. Bohužel nároky úlohy se časem zvyšují, a tak je potřeba rozšiřovat i schopnosti kamerového systému, např: pro práci v proměnných světelných podmínkách, identifikaci různobarevných výrobků nebo rozpoznání dvou téměř stejně barevných výrobků. Pro vyřešení těchto úloh se zpravidla přechází na kamerové systémy rozpoznávající UV nebo IR záření, takže rušivé nebo proměnné jevy ve VIS nesnižují spolehlivost systému. Při takovéto aplikaci ovšem vycházíme z předpokladu, že známe pozorovanou scénu. Pokud předem neznáme vlastnosti pozorované scény, musíme využít hyperspektrální kamerové systémy, které nám dají dostatečně detailní přehled o vlastnostech dané scény. Ty můžeme následně vyhodnotit a zvolit optimální řešení dané úlohy rozpoznávání výrobku, tj. které vlnové délky je potřeba zaznamenávat. Výhodou hyperspektrálního systému je snadné přizpůsobení změnám ve scéně či změně rozpoznávaného výrobku, bez nutnosti výměny kamery či použitých filtrů. Nejrozšířenější úlohy využívající hyperspektrální kamerové systémy jsou pro pozorování země ze satelitu či dronu, které sledují expanzi lidské činnosti se zaměřením na flóru, dále kontrola úrody na polích a vojenské a bezpečnostní aplikace.

Kromě principu záznamu hyperspektrální kostky můžeme kamerové systémy dělit podle velikosti a spotřeby. Při konstrukci se záznamové jádro kamery opatří příslušným krytem a doplňujícími systémy vyhodnocování, bateriového napájení apod. To umožňuje následující konfigurace.

- **Laboratorní použití** – vhodné pro vývoj kamery a testování laboratorních vzorků. Kamera je plně ovládaná z řídicího počítače a umožňuje libovolné nastavení vlastností záznamu hyperspektrální kostky. Ukládají se vždy veškeré užitečné informace o stavu kamery pro potřeby pozdějšího vyhodnocení.
- **Průmyslové využití** – je obdobné jako laboratorní využití s tím rozdílem, že kamera má robustní tělo a neměnné nastavení vlastností záznamu hyperspektrální kostky pro opakované měření. Ty se vyhodnocují na výkonném řídicím PC a ukládají se pak pouze jejich výstupy.
- **Přenosné systémy** – jsou ovládané z notebooku ze kterého jsou i napájené nebo mají vestavěnou baterii. Tělo kamery je uzpůsobené pro snadnou manipulaci v terénu s montáží na stativ. Řídicí PC musí umět rychle zobrazit měřené snímky, avšak jejich detailní vyhodnocení může být prováděné později na odpovídajícím HW.

- **Miniaturizované systémy** – jsou vylepšenou verzí přenosných systémů optimalizované pro použití na dronech, letadlech či satelitech. Systém je vybaven řídicím modulem zprostředkovávajícím automatický opakovaný záznam hyperspektrální kostky a její přenos do nadřazeného systému. Přičemž právě vysoké nároky na přenos dat mezi systémy jsou pro tuto úlohu kritické a většinou se řeší kompresí naměřených snímků. Realizací vyhodnocení naměřených snímků přímo v řídicím modulu kamery by snížilo přenosové nároky a otevřelo by nové možnosti využití systému.
- **Automatizované systémy** – opakovaně sledují a vyhodnocují scénu, přičemž upozorňují na nastalé změny. Mohou být doplněny o displej k zobrazování výsledků, nebo být vzdáleně ovládány. Pro autonomní aplikace jsou doplněny o systém úprav záznamu scény podle aktuálních podmínek.

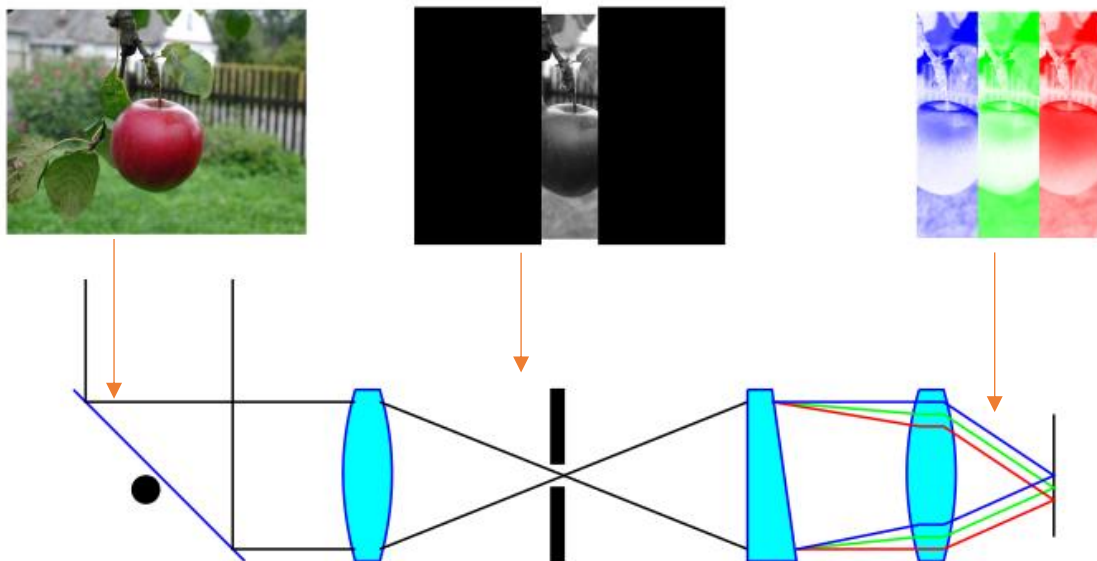
Řídicí a vyhodnocovací část kamerového systému nepočítá vše jen a pouze na procesoru (CPU), ale využívá i nejrůznějších akceleratorů (například grafické karty (GPU)) popřípadě výkonný distribuovaný výpočetní systém kombinující vše předchozí. Miniaturizované kamerové systémy využívají integrované moduly s celým systémem na jednom čipu (system on the chip, SoC), které mají speciální procesor pro zpracování obrazu [11], nebo jsou založené na FPGA platformě [12], kterou si může uživatel přizpůsobit potřebám dané úlohy.

Nástroje pro vyhodnocování hyperspektrálních snímků fungují na bázi matematických modelů či neuronových sítí, které pro svůj výpočet potřebují vysoký výpočetní výkon a vyžadují naučení systému na ohodnocené trénovací množině snímků. Používané softwarové platformy pro vývoj vyhodnocovacích procesů jsou Python či Matlab, a v průmyslu používaná vývojová studia VisionPro [2] nebo perClass Mira [13].

## 2 Hyperspektrální kamerový systém (RODES)

Ve své práci využívám prototyp hyperspektrálního kamerového systému vyvíjeného ve spolupráci s firmou APPLIC s.r.o. a výzkumným centrem speciální optiky a optoelektronických systémů TOPTEC, oddělením Ústavu fyziky plazmatu akademie věd ČR [4] [14]. Systém dokáže detekovat záření v oblasti LWIR, k čemuž využívá nechlazený bolometrický senzor s rozlišením 640×480 px [15]. Nevýhodou těchto senzorů je vyšší úroveň šumu; v porovnání s chlazenými senzory je až desetinásobná. Senzor je vestavěn v infračervené kameře IRCA [16], ke které je přimontován objektiv rozkládající zaznamenané záření do spektrální roviny.

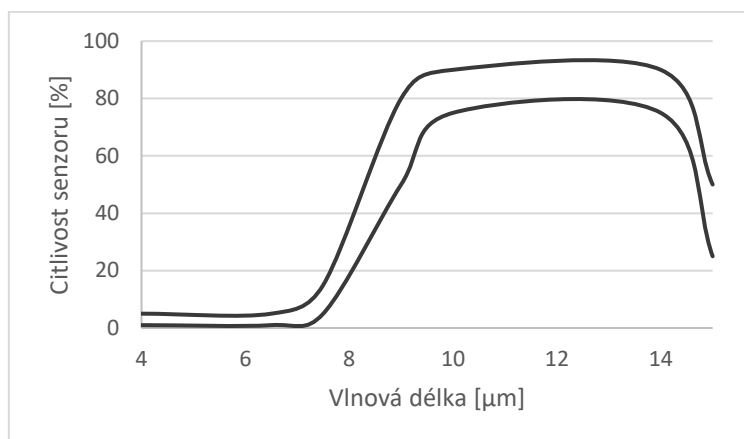
Takto zkonstruovaný hyperspektrální systém je typu „pushbroom“, tj. ke skenování scény využívá zrcadla otáčeného pomocí krokového motoru. Systém obsahuje úzkou štěrbinu, pomocí které je promítaný obraz oříznut na úzký pruh, který je dále rozložen na spektrum, jak je ilustrováno, viz Obrázek 4. Postupným otáčením zrcadla je zaznamenaná celá scéna a vytvořena hyperspektrální kostka.



**Obrázek 4 Rozbor optického systému Hyperspektrální kamery**  
Skládá se ze: skenovacího zrcadla (vlevo), úzké štěrbiny (uprostřed), optické mřížky (vpravo)

Současný používaný režim snímání rozděljuje snímanou scénu na N kroků (20 či 100), přičemž při každém kroku zaznamenává cca 20 snímků, které se následně průměrují do výsledného snímku pro skládání hyperspektrální kostky. Tímto způsobem se zaznamenají přibližně dva kroky za sekundu. Nabízí se také režim kontinuálního otáčení zrcadla, který by umožňoval záznam rychlostí až 50 snímků za sekundu. Jeho nevýhodou je ale vyšší úroveň šumu (vlivem absence průměrování snímků) a nepřesnosti v synchronizaci natočení zrcadla a času pořízení snímku.

Bolometrický senzor převádí zaznamenané záření na analogový elektrický signál, který je následně převáděn na digitální s 14bit rozsahem (výstup převodníku 0–16383 ADU, Analog Digital Unit). Spektrální citlivost senzoru je viz Obrázek 5, která odpovídá spektrálnímu rozsahu LWIR (8–14 $\mu$ m). Objektiv je vyroben převážně z germania, které je v tomto rozsahu taktéž plně propustné.



Obrázek 5 Rozmezí citlivosti senzoru napříč spektrem

Jednotlivé pixely senzoru mají rozdílný pracovní bod, který je potřeba sjednotit, tak aby byl výsledný obraz čitelný. Toho se dosahuje pomocí korekce neuniformity (nonuniformity correction, NUC) [17]. Korekce se provádí nasměrováním kamery na homogenní těleso a je ovlivnitelná změnou objektivu kamery, teploty senzoru a objektivu, a změnou nastavení senzoru. Pro získání kvalitních snímků je kvůli mnoha rušivým faktorům nutné tuto kalibraci často opakovat.

Hyperspektrální systém RODES umožňuje tuto kalibraci na dvou místech. První, před senzorem, je využita při sestavování a seřizování systému a je identická s kalibrací klasické infračervené kamery. Druhá, před objektivem, se provádí před každým záznamem hyperspektrálního snímku a při delších snímacích dobách i po záznamu snímku. Získané kalibrační snímky se poté odečítají od naměřených dat ze scény, čímž se zvýrazní teplotní rozdíly ve scéně a spektru.

Obecné vlastnosti systému RODES doplněné o výstupy provedených kalibračních měření jsou viz Tabulka 1. Obdobná koncepce tohoto systému byla představena v publikaci [D].

Tabulka 1 Vlastnosti systému RODES

Vlastnost	Hodnota	Poznámka
Rozlišení v prostoru	300 px	Úzký pruh scény
Rozlišení ve spektru	400 px	
Bitová hloubka	16 bit	Aktivních 14bitů

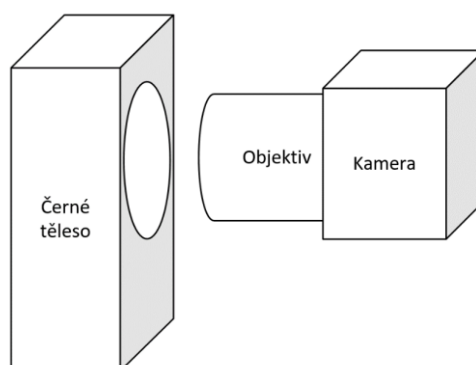
Vlastnost	Hodnota	Poznámka
Velikost senzoru	640×480 px	17 μm na pixel
Rychlost snímání	50 Hz	
Rychlost hyp. záznamu	Pod 10 s (20 pruhů na scénu)	Průměr z 20ti snímků
Viditelná scéna	40°	FOV
Světelnost objektivu	f/1	Maximální světelnost
Rozhraní	1Gb/s Ethernet	
Citlivost	Mono 500, Hyper 1 ADU/°C	
Spektrální rozsah	7500–12000 nm	LWIR
Spektrální rozlišení	12 nm/px	
Šum	NESR < 100, NETD < 70	

## 2.1 Kalibrace systému

Už při sestavování hyperspektrálního systému je nutné provést několik kalibrací. Jako první je potřeba nastavit samotnou infračervenou kameru, bez hyperspektrálního objektivu, tak aby dávala čitelný obraz. Poté se připevní část objektivu se štěrbinou a systém se zaostří. Následně lze sestavit kompletní hyperspektrální kameru, která pak využívá už jen kalibraci před objektivem na interní kalibrační černé těleso dosažitelné správným natočením skenovacího zrcátka.

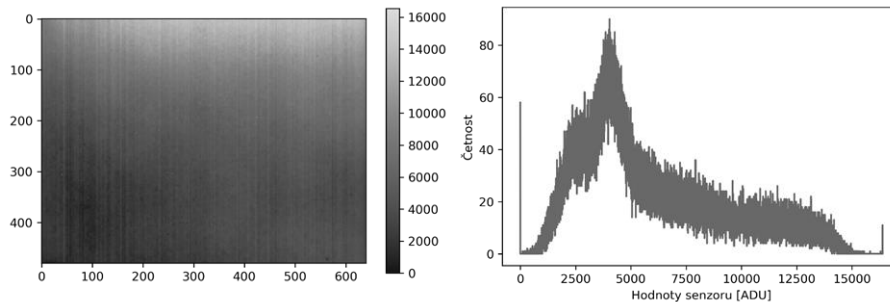
### 2.1.1 Kalibrace infračervené kamery

Infračervená kamera má integrovanou záklopku se speciálně upraveným povrchem, umožňující rychlou kalibraci senzoru. Pro přesnější korekci a analýzu vlastností senzoru je ale potřeba využít dostatečně velkého černého tělesa a naměřit korekční snímky pro několik jeho teplot (typicky pro 10, 20, 25, 30, 40, 70, 100 °C). Při měření musí být kamera relativně blízko černému tělesu, viz Obrázek 6, tak aby nebyly vidět hrany kalibrační plochy, tj. cca 2 cm od černého tělesa. Objektiv by měl být ideálně zaostřen do nekonečna. Před začátkem každého měření je potřeba dostatečně dlouhý interval (cca 20 min) pro ustálení provozní teploty kamery. Poté lze zaznamenávat série snímků (postačí 20 snímků) v radiometrickém formátu pro následné vyhodnocení.



Obrázek 6 Kalibrační scéna infrakamery

Senzor má kromě nastavení integrační doby dalších pět nastavitelných parametrů (napájecích či komparačních napětí), které ovlivňují jeho pracovní rozsah. Tento stavový prostor byl podrobně prozkoumán, analyzován a výsledky publikovány [C], přičemž byl nalezen optimální pracovní bod senzoru s maximálním zesílením. Snímek kalibračního černého tělesa při tomto nastavení je viz Obrázek 7 současně s jeho histogramem. Na něm je pozorovatelné, že rozptřeni hodnot jednotlivých pixelů při pozorování homogenní scény pokrývá téměř celý rozsah převodníku (14k ze 16k ADU).



Obrázek 7 Kalibrační snímek a jeho histogram

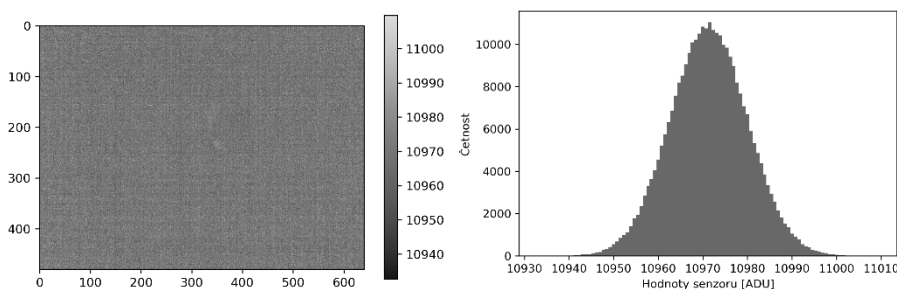
Porovnáním měření dvou blízkých teplot černého tělesa ( $T_1$ ,  $T_2$ ) lze podle (2.1) ke každému pixelu  $(x, y)$  dopočítat jeho teplotní citlivost  $G(x, y)$  [ADU/°C]. Pro nalezené optimální nastavení je průměr citlivosti za celý snímek 500 ADU/°C.

$$G(x, y) = \frac{S_2(x, y) - S_1(x, y)}{T_2 - T_1} \quad (2.1)$$

Rozsah pozorovatelné teploty odpovídá přibližně rozmezí  $\pm 2$  °C poté dochází k saturaci části obrazu. To je nevhodné pro běžné využití infračervené kamery, kde difference teplot pozorované scény je obvykle větší než 4 °C. Pro hyperspektrální použití bude v důsledku rozložení energie ve spektru citlivost významně snížena, čímž dojde k rozšíření rozsahu pozorovatelné teploty nejméně na  $\pm 50$  °C. Střední hodnota rozsahu je nastavitelná speciálním parametrem senzoru, při jehož změně je potřeba opětovná kalibrace systému.

Pro zvýraznění obrazové informace je potřeba vyrovnat obraz jednobodovou NUC, tak aby při pohledu na homogenní těleso měli všechny pixely přibližně stejnou hodnotu. Toho dosáhneme naměřením kalibračního snímku homogenní plochy černého tělesa (nebo záklopy kamery)  $S$ , od kterého odečteme jeho průměr  $\bar{S}$ , čímž získáme korekční matici  $C(x, y)$  (2.2). Korekční matici poté odečítáme od každého následujícího zaznamenaného snímku z kamery, a to do té doby, než se teplota senzoru nebo pozorovaná scéna příliš změní a bude potřeba opakované provedení kalibrace

$$C(x, y) = S(x, y) - \bar{S} \quad (2.2)$$

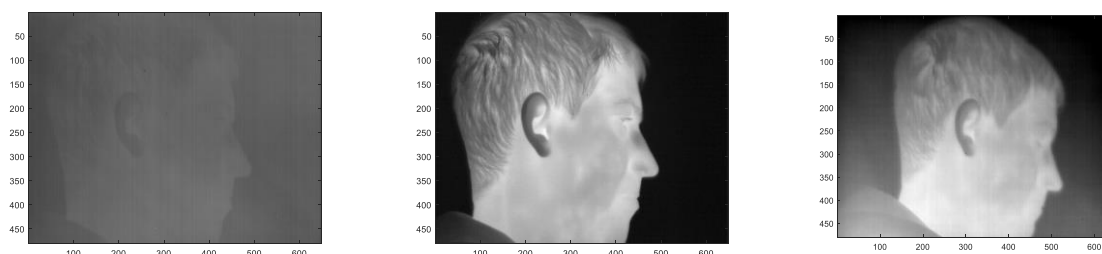


Obrázek 8 Snímek po kalibraci a jeho histogram

Infračervené kamery zaznamenávající širší rozsah teplot využívají náročnější dvoubodovou NUC zohledňující individuální zesílení každého pixelu  $G(x,y)$ . Pro každý další zaznamenaný snímek  $P(x,y)$  platí výpočet (2.3) obrazu po korekci  $DC(x, y)$ .

$$DC(x, y) = G(x, y)P(x, y) - C(x, y) \quad (2.3)$$

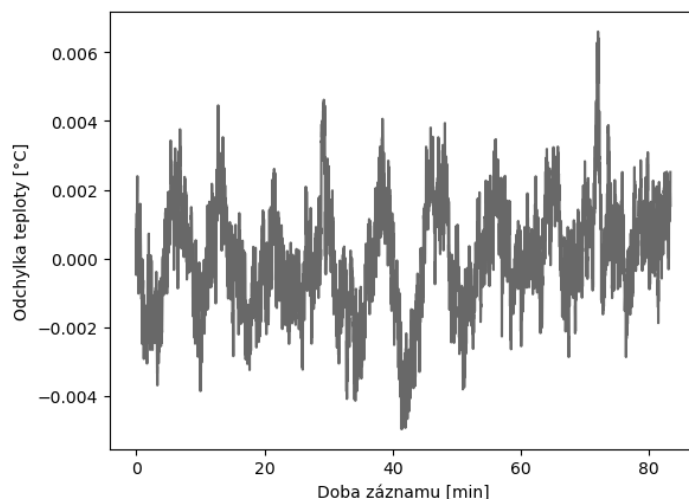
Pro korekci senzoru v celém rozsahu hodnot každého pixelu, pak místo více bodové korekce, byla navržena přesnější polynomiální korekce [A]. Ta je však používána jen u systémů s vysokou dynamikou obrazu. Výsledek aplikace NUC na pořízený snímek z kamery je viz Obrázek 9, na kterém je znázorněna i degradace kvality snímku/korekce po určitém čase.



Obrázek 9 Snímek infračervené kamery  
Před korekcí (vlevo), po korekci (uprostřed), a po půl hodině (vpravo)

Pro zajištění opakovatelnosti měření černého tělesa při jeho různých teplotách, je bezpodmínečně nutné celý proces plně automatizovat. Nepříznivým jevem pro náměr korekce je samovolná změna teploty senzoru, který je nechlazený a teplotně neregulovaný. Řešením je využití více kalibračních černých těles, předem stabilizovaných na různých teplotách, a jejich rychlou záměnou minimalizovat čas měření, nebo přidání teplotní stabilizace přímo na senzor např. pomocí Peltierova článku. Kromě ukládání snímků ze senzoru a nastavené teploty černého tělesa, je vhodné zaznamenávat i aktuální teploty černého tělesa a samotného senzoru při pořízení každého snímku. Vysoce citlivé systémy totiž rozpoznají rozdíly teplot v řádu  $0.001\text{ }^{\circ}\text{C}$  a lze tak zaznamenat například i regulační proces černého tělesa, viz Obrázek 10.



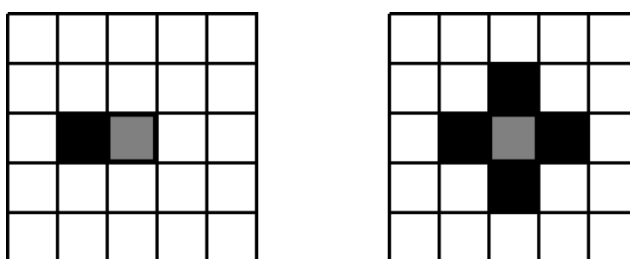


**Obrázek 10** Časová stabilizace kalibračního černého tělesa na teplotě 30 °C

Průmyslové infračervené kamery prochází dvoubodovou kalibrací při výrobě s doporučenou opětovnou kalibrací po jednom roce používání. Při jejich používání se průběžně aktualizuje už jen jednobodová složka korekce.

### 2.1.2 Korekce vadných pixelů

Bohužel, malé procento pixelů senzoru (popřípadě i celý řádek či sloupec) dávají výrazně odlišný signál, který nelze pomocí NUC dostatečně vyrovnat. Tyto pixely jsou poškozeny už z výroby a dávají saturovaný nebo nerelevantní signál. Lze je odstranit ze snímku přímo při výčtu hodnoty daného vadného pixelu ze senzoru a nahradit jej hodnotou předchozího vyčteného pixelu, nebo později při zpracování snímku dopočtením jeho náhradní hodnoty podle hodnot okolních pixelů, viz Obrázek 11.



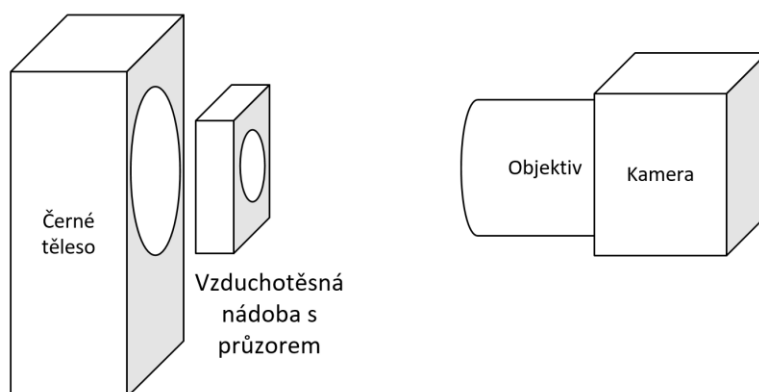
**Obrázek 11** Korekce vadných pixelů senzoru

Nahrazení hodnotou předchozího pixelu (vlevo), dopočet hodnoty z okolních pixelů (vpravo)

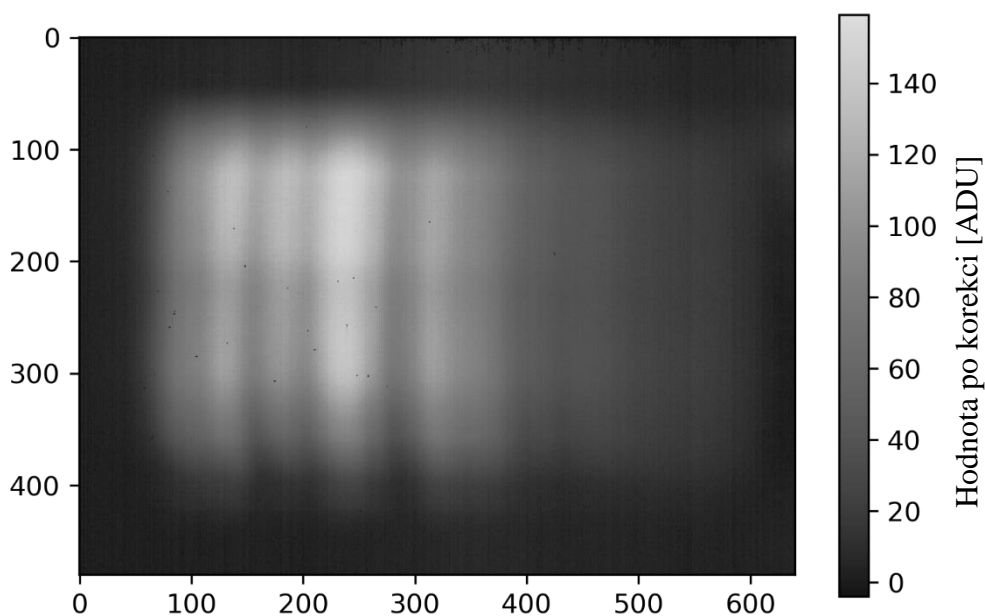
### 2.1.3 Zaostření kamery

Objektiv hyperspektrální kamery je potřeba seřadit tak aby byly dobře pozorovatelné jednotlivé spektrální vrcholy kalibračního přípravku. To je možné pomocí části infračerveného objektivu a správné pozice štěrbin. Pro kalibraci využívám chemickou látku Isopropyl alkohol ve speciální vzduchotěsné nádobě s germaniovými průzory. Schéma sestavení soustavy je vyobrazené viz Obrázek 12. Černé těleso sloužící jako pozadí scény je nastaveno na 100 °C,

což umožňuje snadné pozorování spektrálních průběhů vzorku viz Obrázek 13. Pro zvýraznění viditelnosti spektra chemické látky je potřeba provádět NUC obrazu infračervené kamery.

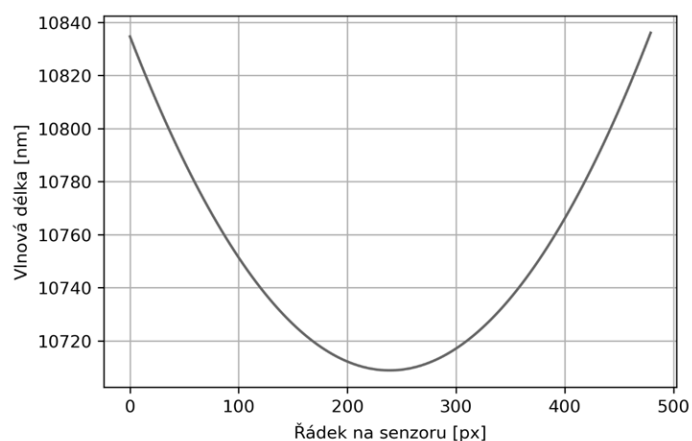


Obrázek 12 Schéma měření spektrální charakteristiky plynů



Obrázek 13 Zobrazené spektrální průběhy při kalibraci objektivu hyperspektrální kamery

Kromě ostrosti záznamu je vidět deformace snímku tzv. „smile“, při kterém jsou některé testované spektrální vrcholy prohnuté přes více sloupců. Pro lepší viditelnost je tento efekt znázorněn pro středový řádek, viz Obrázek 14, na kterém se vlnová délka mění o cca 120 nm, ta odpovídá cca 10 px, což je cca 1% zkreslení.

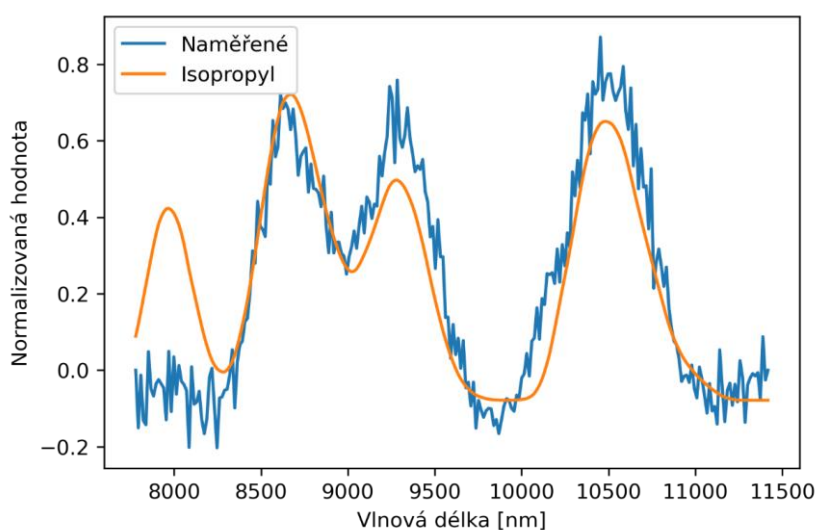


Obrázek 14 Deformace spektra na senzoru „smile“

#### 2.1.4 Identifikace rozložení spektra na senzoru

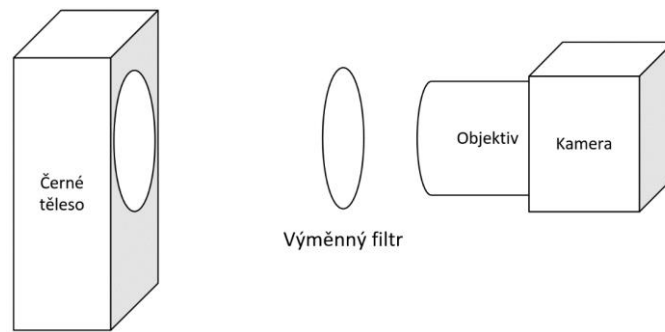
Poté co je hyperspektrální objektiv kompletně sestaven, lze přejít k měření vzorků a záznamu vzorových hyperspektrálních kostek, pomocí kterých se identifikuje rozložení spektra na senzoru. Nevyužívá se už původní kalibrace infračervené kamery, ale před každým záznamem kostky se pro kalibraci nasnímá integrované černé těleso v objektivu. Využívá se jednobodové NUC s tím rozdílem, že tentokrát je v měření zahrnutý i vliv celé hyperspektrální optické soustavy.

Spektrální charakteristika měřeného vzorku je rozložena do řádku na senzoru. Pro každý řádek se hledá nejshodnější proložení s laboratorně měřenou křivkou odpovídajícího vzorku z databáze (National Institute of Standards and Technology, NIST) [8], tak aby se jednotlivé vrcholy křivek překrývaly. Po nalezení nejlepšího proložení křivky, lze přiřadit jednotlivým pixelům v řádku reprezentativní vlnovou délku, viz Obrázek 15.

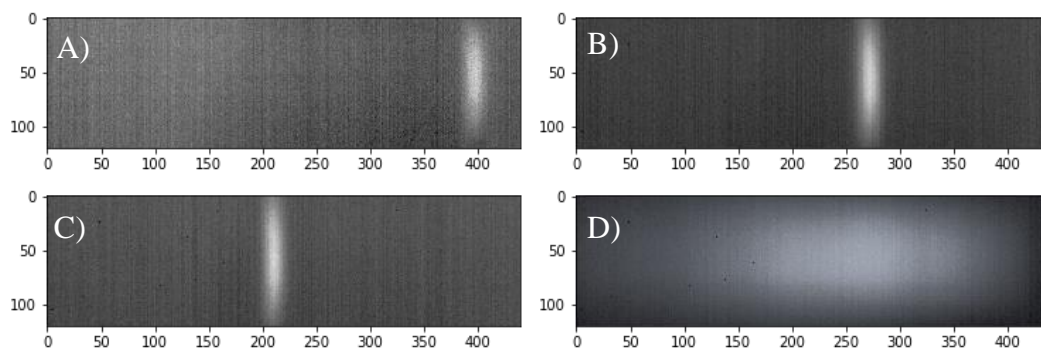


Obrázek 15 Sjednocené spektrální charakteristiky Isopropylu

Obdobně lze využít kalibrace pomocí sady úzkopásmových filtrů. Výhodou využití této metody kalibrace je jednoznačnost a viditelnost jednotlivých spektrálních vrcholů. Sestavení systému je obdobné jako pro plyny, přičemž jednotlivé filtry jsou umístěné v otočném kole, pro jejich snadnou výměnu, viz Obrázek 16. Ukázky naměřených snímků, pro jednu pozici zrcadla kamery jsou viz Obrázek 17. Měření je plně automatizováno a prováděno na optické lavici, tak aby byly zajištěné stejné podmínky pro každé z měření.



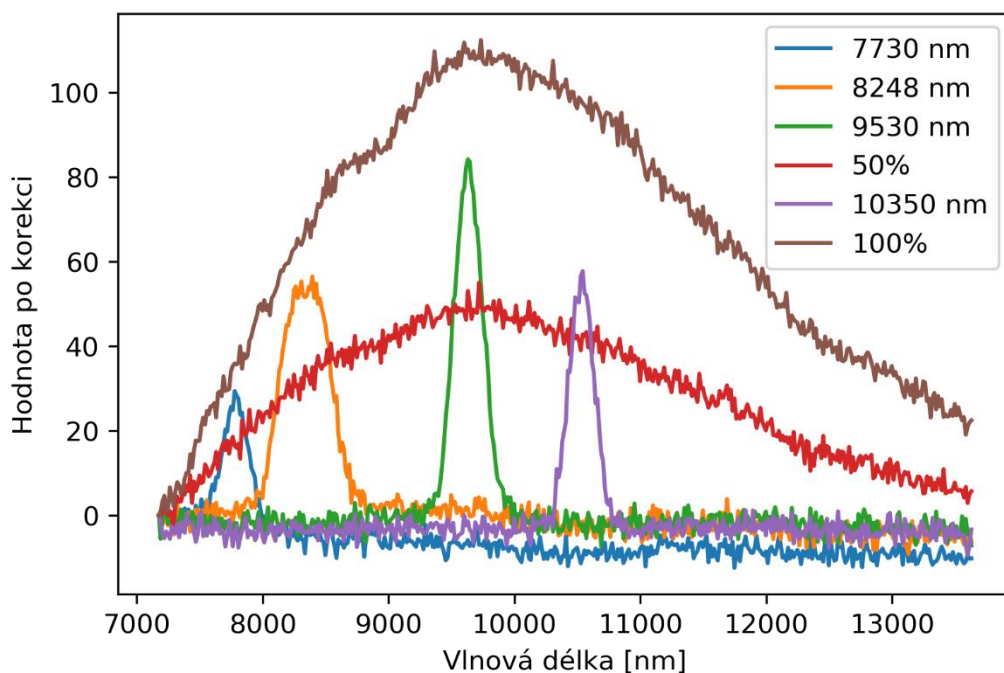
**Obrázek 16** Schéma měření scény pomocí filtrů



**Obrázek 17** Zaznamenané snímky ze senzoru pro vybrané filtry

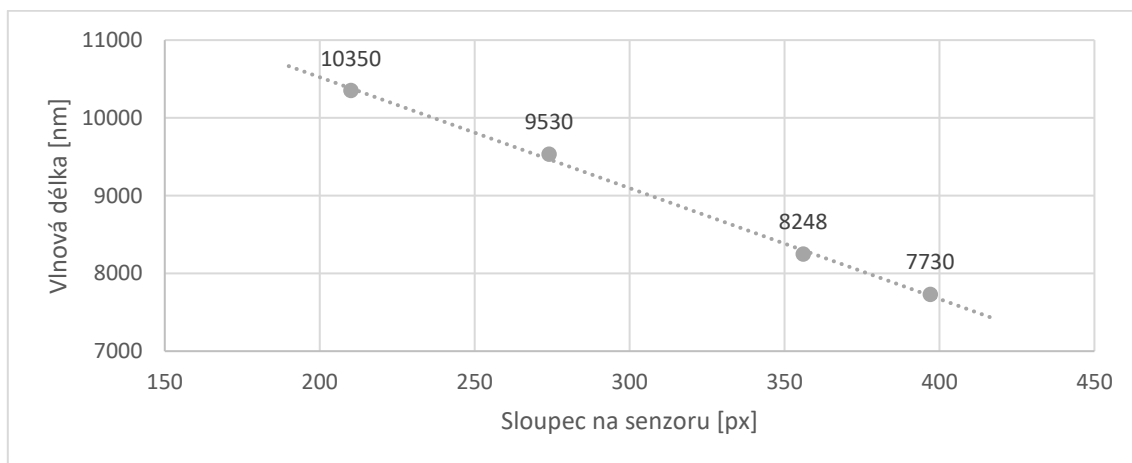
A) Filtr 7730  $\mu\text{m}$ , B) Filtr 9530  $\mu\text{m}$ , C) Filtr 10350  $\mu\text{m}$ , D) Bez filtru

Naměřené průběhy dostupné sady úzkopásmových filtrů (označeny jejich centrální vlnovou délkou), jednoho polopropustného filtru (50%) a scény bez filtru (100%) jsou vyobrazeny, viz Obrázek 18. Díky vysoké propustnosti filtrů lze snadno identifikovat vrcholy jednotlivých spektrálních charakteristik odpovídajících propustné vlnové délce filtru.



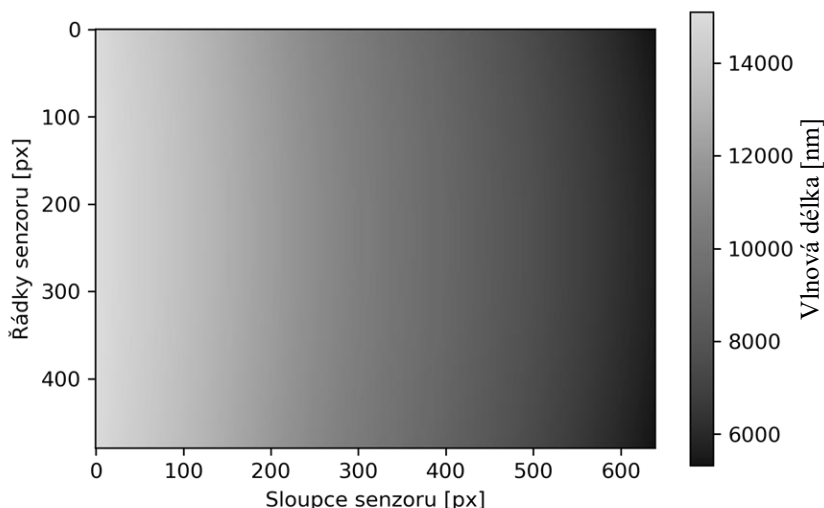
Obrázek 18 Naměřené spektrální charakteristiky filtrů

Jednotlivé pozice identifikovaných vrcholů spektrálních charakteristik na daném řádku lze proložit lineární regresí a dopočítat tak odpovídající vlnové délky pro každý pixel v daném řádku, viz Obrázek 19. Odpadá tak hledání nejlepšího proložení křivek vzorku plynu, které vlivem nečistot kalibračního plynu a šumu signálu může být zavádějící.



Obrázek 19 Lineární regrese vrcholů jednotlivých filtrů

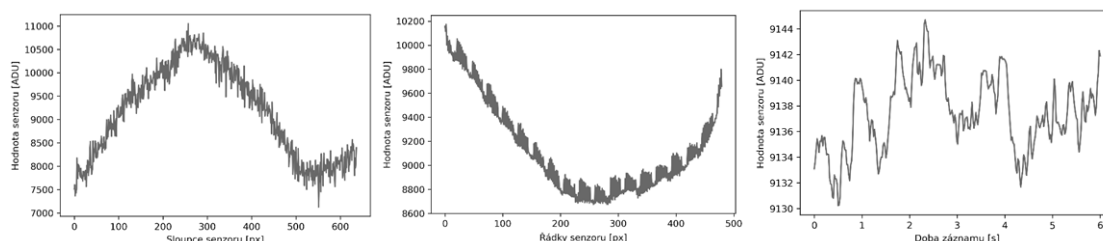
Po identifikaci vlnových délek pro jednotlivé řádky je potřeba vyhladit výslednou mapu vlnových délek i ve směru sloupců, protože optický systém ji na senzor promítá spojitě. Výsledná vyhlazená spektrální mapa senzoru je vyobrazena viz Obrázek 20.



Obrázek 20 Vyhlazená spektrální mapa senzoru

## 2.2 Identifikace šumu systému

Ačkoliv teoretické rozlišení senzoru je v řádu setin stupně, prakticky je snímek vždy zatížen šumem odpovídajícím řádu několika desetin stupně. Jedná se o šum ovlivňující celý senzor a je spjat se samotnou funkcí senzoru (pracovní bod, teplota, snímkovácí frekvence). Na šum naměřených snímků lze nahlížet ze tří rovin: prostorově v osách senzoru X a Y a pak v čase, viz Obrázek 21. Statická složka prostorového šumu se částečně vykompenzuje aplikováním NUC. Hlavním uváděným parametrem infračervených senzorů a kamer je správná reprezentace časového šumu napříč senzorem.

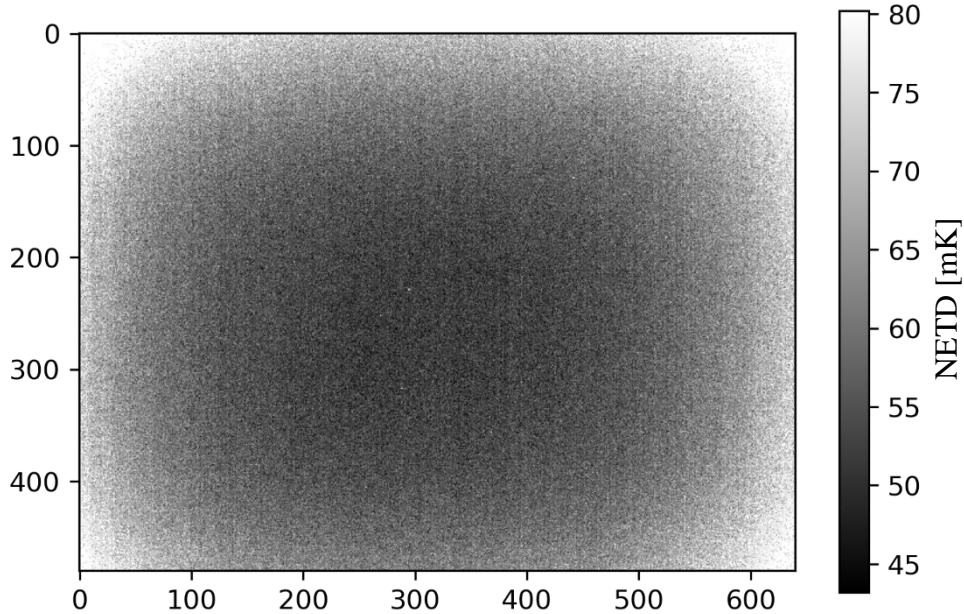


Obrázek 21 Šum kamery v ose X (vlevo), v ose Y (uprostřed), v čase (vpravo)

Ten se vyjadřuje ve vztahu k teplotnímu rozdílu (Noise Equivalent Temperature Difference, NETD). Pro jeho výpočet je potřeba provést náměr černého tělesa o třech teplotách (obvykle pro 20, 25 a 30 °C), z nichž se pro každou teplotu zaznamená 100 až 300 snímků. Měření pro krajní teploty ( $T_{20}$  a  $T_{30}$ ) slouží pro výpočet zesílení pixelu zpřesněné průměrováním přes naměřené snímky ( $\bar{S}_{20}$  a  $\bar{S}_{30}$ ) podle (2.4). Ze středové teploty ( $S_{25}$ ) se pro každý pixel vypočte směrodatná odchylka a jejím podělením vypočteným zesílením podle (2.5) vznikne mapa NETD šumu napříč senzorem, viz Obrázek 22. Udávaná výsledná hodnota je výsledný průměr šumu  $\overline{NETD}$  [mK], který se pro nechlazené senzory pohybuje v rozmezí 20–50 mK a pro chlazené senzory až 2 mK.

$$G(x, y) = \frac{\bar{S}_{30}(x, y) - \bar{S}_{20}(x, y)}{T_{30} - T_{20}} \quad (2.4)$$

$$NETD(x, y) = \frac{\sqrt{\text{var}(S_{25}(x, y))}}{G(x, y)} \quad (2.5)$$



Obrázek 22 Rozložení NETD na senzoru

Průměrná hodnota šumu NETD pro senzor je 62 mK a pro středovou oblast (využívanou systémem RODES) je výsledná hodnota 56 mK. Naměřené výsledky jsou vyšší než nominální šum senzoru, což je způsobeno elektronikou kamery a vyšší snímkovací frekvencí senzoru (nominální je 30 Hz). Dodatečným průměrováním cca 20 snímků a stabilním prostředím kamery, lze snížit šum NETD až na třetinu.

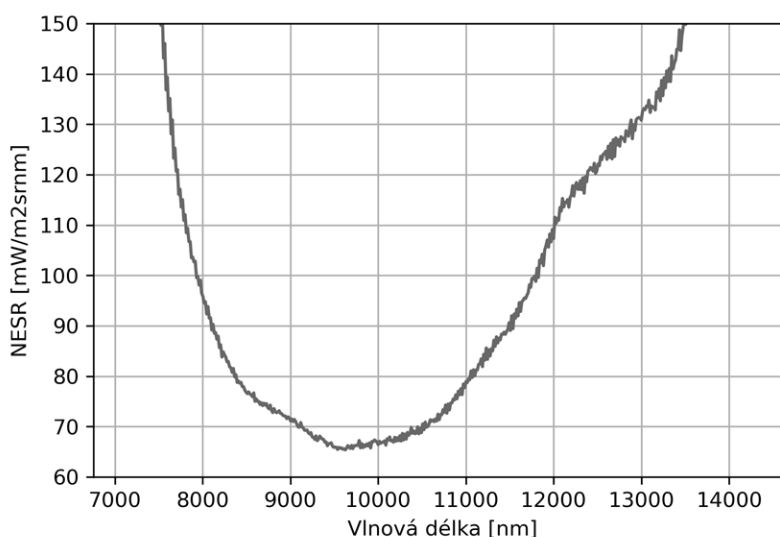
### 2.2.1 Identifikace šumu hyperspektrální kamery

Šum hyperspektrální kamery se vyjadřuje obdobně jako NETD u infračervené kamery. Rozdílný je výpočet difference teplot, které jsou přepočítány ze stupňů na energii podle Planckova zákona  $P(T, \lambda)$  a odpovídajícího rozložení vlnové délky na senzoru (Noise Equivalent Spectral Radiance, NESR) [ $\text{Wsr}^{-1}\text{m}^{-2}\text{nm}^{-1}$ ]. Vypočítá se podle rovnice (2.6), přičemž vlnová délka pro daný pixel je daná již identifikovanou spektrální mapou  $\lambda(x, y)$ .

$$NESR(x, y, \lambda) = \frac{P(T_2, \lambda(x, y)) - P(T_1, \lambda(x, y))}{S_2(x, y) - S_1(x, y)}$$

(2.6)

Výsledek NESR pro středovou část senzoru je vyobrazen viz Obrázek 23. Při zvolení maximální přípustné hodnoty NESR na  $100 \text{ mWsr}^{-1}\text{m}^{-2}\text{nm}^{-1}$  je spektrální rozsah kamery 8000–11500 nm.

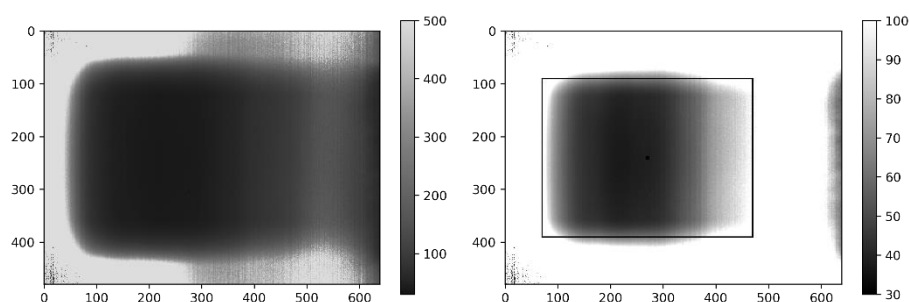


Obrázek 23 Průměr šumu NESR hyperspektrální kamery

Jiné dostupné kamery [18] založené na stejné technologii mají NESR pro stejný rozsah až  $3\times$  vyšší. Stejný výrobce má pro porovnání i kamery založené na chlazených senzorech, které mají NESR naopak  $3,5\times$  nižší.

### Výběr použitelného regionu senzoru kamery

Použitelnou plochu senzoru vybereme stanovením přípustné šumové hodnoty  $NESR < 100 \text{ mWsr}^{-1}\text{m}^{-2}\text{nm}^{-1}$ , kterou vyplníme obdélníkem. Tím jsme našli použitelný aktivní region senzoru od (70, 90) do (470, 390) pixelů, viz Obrázek 24. Efektivní využití plochy senzoru je přibližně 40 % a průměrná hodnota NESR v tomto regionu činí  $60 \text{ mWsr}^{-1}\text{m}^{-2}\text{nm}^{-1}$ .



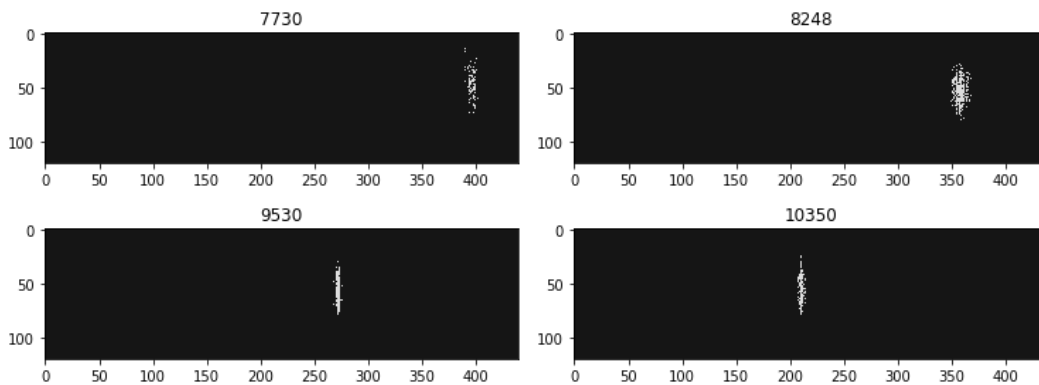
Obrázek 24 Rozložení šumu NESR na senzoru a výběr aktivní oblasti



### 2.3 Naměřené vzorky

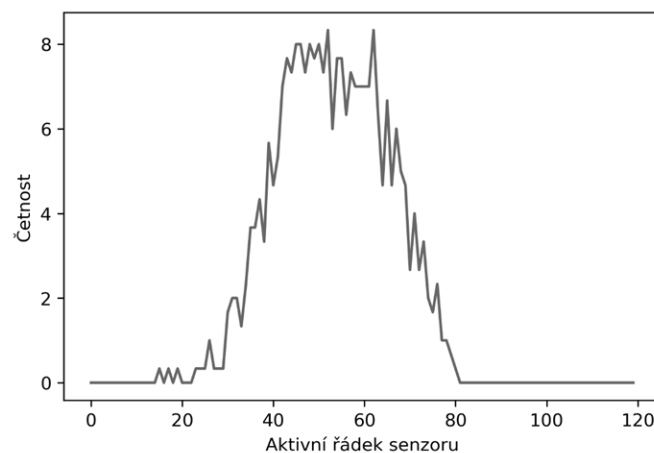
Jednotlivé vzorky naměřené pro účely kalibrace a testování funkce kamery byly znovu použity pro trénování a testování vyhodnocovacích funkcí. Díky automatizovanému měření a opakovatelnosti úlohy bylo možné vytvořit funkci pro anotování naměřených snímků.

Jednotlivé filtry byly umístěné v otočném kole na optické lavici, a tak byla pozice měřeného filtru v prostoru vždy stejná. Binarizací snímku s prahem hodnoty 80% maxima byly získané pozice charakteristického vrcholu spektrální charakteristiky filtru na senzoru, viz Obrázek 25.



Obrázek 25 Pozice vrcholů spektrálních charakteristik vybraných filtrů

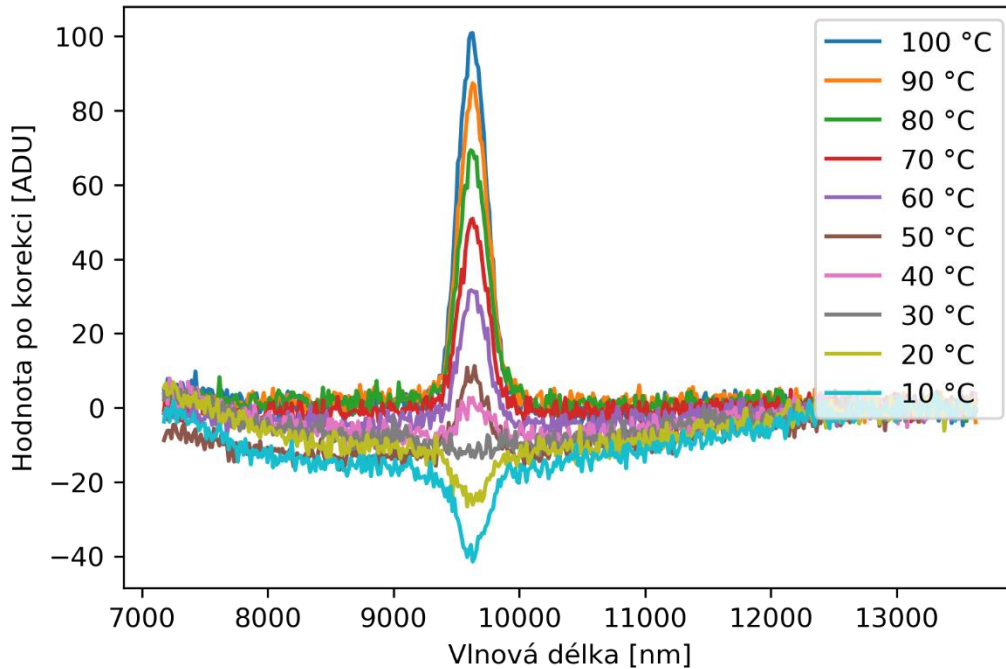
Přeložením jednotlivých binarizovaných snímků přes sebe a jejich promítnutím do roviny reprezentující skenovaný řez scény získáme pozici promítnutých filtrů na senzoru, viz Obrázek 26. Výběrem hodnot překračujících 20% maxima výsledné křivky byl identifikován region hyperspektrální kostky obsahující spektrální charakteristiky měřených filtrů (pozice 30–75).



Obrázek 26 Počet vrcholů v obrazové ose senzoru

Každý filtr byl měřen vůči teplotám černého tělesa o hodnotách 10 až 100 °C v kroku po 10 °C, viz Obrázek 27. Tím byla zaznamenána klesající naměřená energie spolu s teplotou

pozadí. Můžeme pozorovat oblast okolo  $\sim 30$  °C, při které je teplota filtru podobná teplotě pozadí. Tato oblast nemá charakteristický vrchol transmisivity filtru a měřitelná je pouze vlastní emisivita filtru.



Obrázek 27 Teplotní rozdíly průběhu filtru 9530 nm

Výsledná testovací hyperspektrální kostka „Filtry“ ze systému RODES zaznamenaná ve spektrálním rozsahu od 7500 do 12000 nm rozděleným na 400 pásem obsahuje 6000 záznamů o 4 třídách a 10 různých teplotách pozadí. Zároveň byla vytvořena i testovací hyperspektrální kostka „FiltryNC“, která obsahuje totožné průběhy, ale bez provedených korekcí.

### 3 Vyhodnocení hyperspektrální kostky

Při vyhodnocení přiřadíme každou naměřenou spektrální charakteristiku do jedné z vyhodnocovaných materiálových tříd. To je komplikované zejména při přechodech mezi materiály (hrany objektu), nebo při směsi vyhodnocovaných materiálů doplněné o proměnný vliv okolní (stíny, počasí). Pro ověřování algoritmů používám vícero různých hyperspektrálních kostek tak, aby bylo možné objektivněji posoudit jejich vlastnosti.

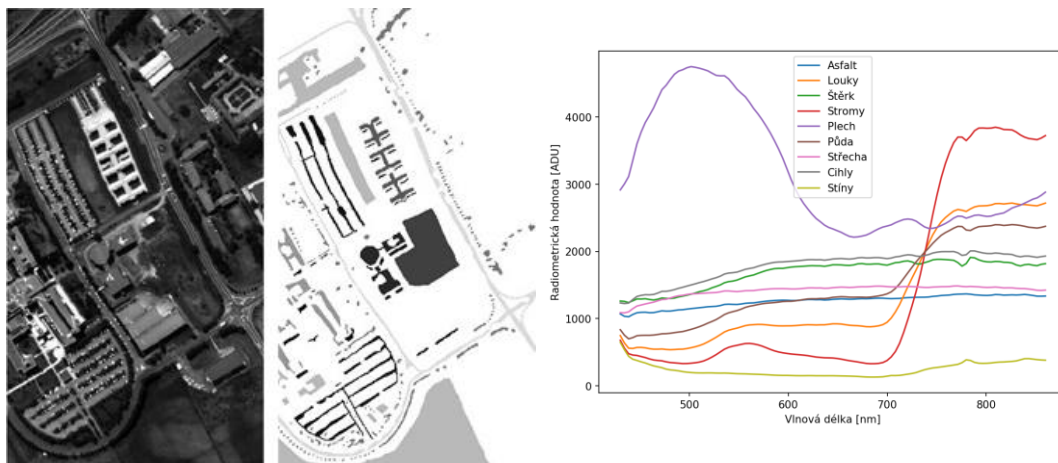
Cílová sada rozpoznávaných chemických materiálů pro LWIR systém prozatím obsahuje jen malé množství vzorků. Využívám tedy i záznamy z jiných systémů, na kterých lze porovnat náročnost vyhodnocení pro více početné a hůře rozeznatelné materiálové třídy s komplexnější spektrální charakteristikou. Podrobně byly jednotlivé metody diskutovány v publikaci [B], přičemž dále jsou diskutovány aktualizované výsledky pro novější verzi používané hyperspektrální kamery RODES.

#### 3.1 Vyhodnocované vzorky

Byly vybrány dva významné hyperspektrální vzorky, které jsou často citované a slouží pro demonstraci vyhodnocovacích algoritmů, a to vzorek „Pavia University“ a „Indian pines“. Oba vzorky jsou leteckými záznamy scény v oblasti VIS a IR a jsou veřejně dostupné [19].

##### 3.1.1 Testovací vzorek „Pavia University“

Hyperspektrální snímek pořízený optickým systémem (Reflective Optics System Imaging Spectrometer, ROSIS), který provozuje islandská výzkumná instituce (Center for Remote Sensing, CRS) s cílem záznamu spektrálních struktur zejména na pobřeží [20]. Senzor disponuje spektrálním rozsahem od 430 do 860 nm, který rozděluje na 115 pásem. Testovací vzorek obsahuje 150 tisíc klasifikovaných pixelů rozdělených nerovnoměrně na 9 tříd jejichž reprezentativní spektrální charakteristiky získané jejich průměrem jsou viz Obrázek 28.

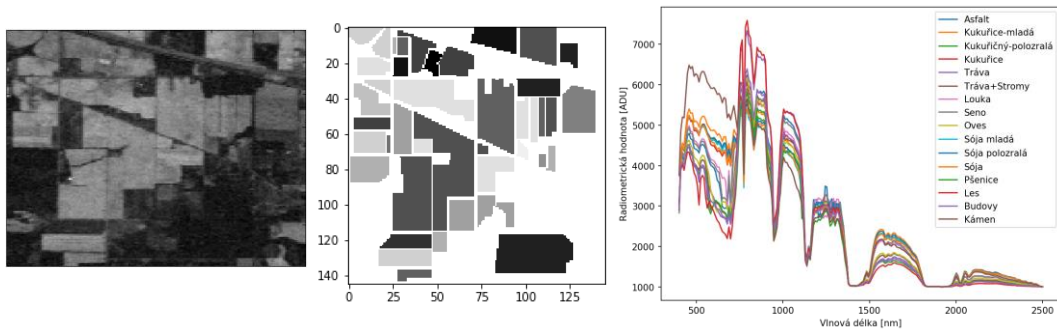


Obrázek 28 Vzorek „Pavia University“

Snímek scény (vlevo), klasifikace scény (uprostřed) a spektrální charakteristiky tříd (vpravo)

### 3.1.2 Testovací vzorek „Indian pines“

Hyperspektrální snímek pořízený optickým systémem (Airborne Visible InfraRed Imaging Spectrometer, AVIRIS), který je provozovaný kosmickou agenturou NASA s cílem záznamu povrchu země pro účely výzkumu globálního prostředí a změny klimatu [21]. Senzor disponuje spektrálním rozsahem od 400 do 2500 nm, který rozděluje na 176 pásem. Testovací vzorek obsahuje 90 tisíc klasifikovaných pixelů rozdělených nerovnoměrně na 16 tříd, viz Obrázek 29.



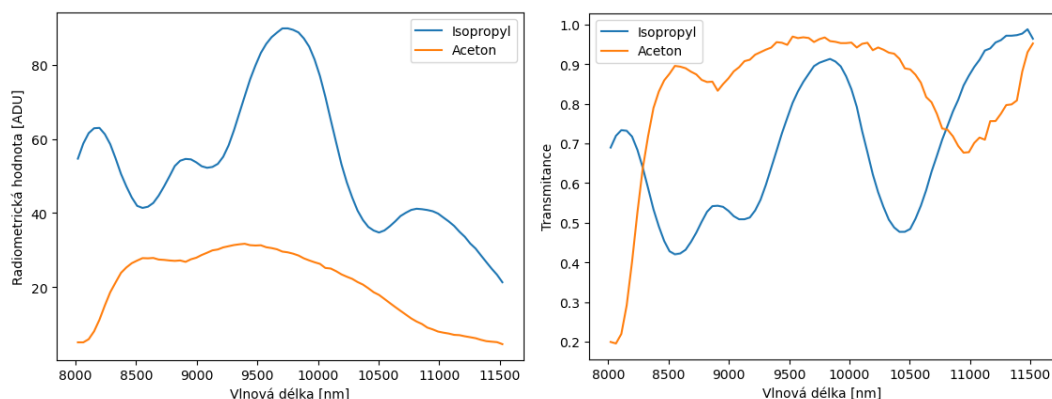
**Obrázek 29** Vzorek „Indian pines“  
Snímek scény (vlevo), klasifikace scény (uprostřed) a spektrální charakteristiky tříd (vpravo)

## 3.2 Vyhodnocovací algoritmy

Nejprve se určí podobnost spektrální charakteristiky každého prostorového pixelu s rozpoznávanou množinou tříd např. pomocí metriky Euklidovské vzdálenosti. Výsledky lze pak interpretovat, buď jako mapu podobnosti pro vybranou třídu, nebo pro každý pixel scény přiřadit k nejpodobnější třídě.

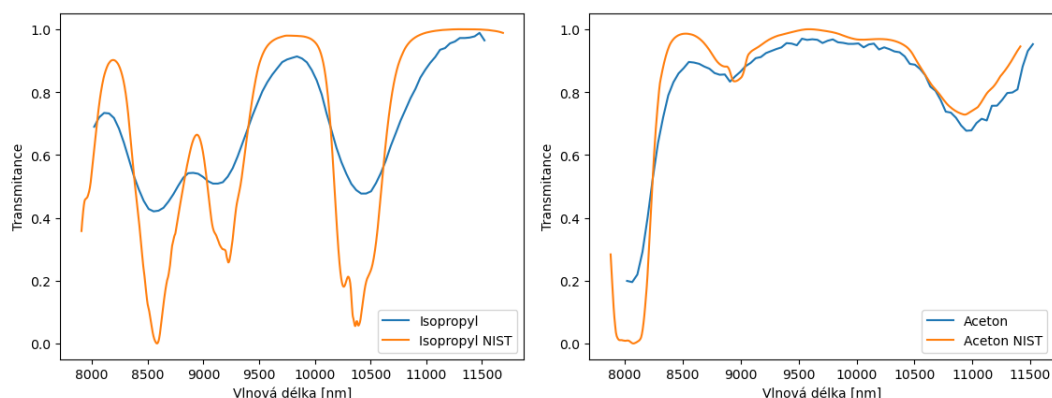
Každá vyhodnocovaná třída má svoji reprezentativní spektrální charakteristiku, získanou průměrem vybraných členů dané třídy. Ty by měly být ze středu shluku stejných pixelů dané třídy, nikoliv z jejího okraje, tak aby se minimalizovalo zkreslení charakteristik kombinací s jinými třídami.

Reprezentativní spektrální charakteristiky daných tříd, lze porovnat s detailními laboratorně naměřenými charakteristikami získané z veřejné databáze NIST [8]. Ty jsou dostupné v hodnotách Transmisivity nebo Emisivity, tudíž je potřeba naměřené radiometrické průběhy převést na příslušnou fyzikální veličinu, viz Obrázek 30. Toho je dosaženo odečtením kalibračních snímků a porovnáním s odhadem teploty pozadí.



**Obrázek 30 Vstupní naměřené průběhy chemikálií a jejich přepočítání na Transmittanci**  
 Naměřená hodnota chemikálií (vlevo), přepočtený průběh na Transmittanci (vpravo)

Bohužel, kvůli rozdílným podmínkám měření a nepřesnostem ve vyhodnocení (odhad teploty pozadí, atmosférický vliv, nečistoty materiálu ...) jsou naměřené průběhy a laboratorní průběhy značně rozdílné, viz Obrázek 31. Proto není vhodné využít dostupné laboratorní vzorky přímo k vyhodnocování celé scény, ale pouze pro ověření jednotlivých tříd, nebo pro kalibraci systému.



**Obrázek 31 Porovnání vypočtených transmittančních průběhů s databází NIST**  
 Porovnání průběhů Isopropylu (vlevo), Acetonu (vpravo)

Porovnávat lze místo pouze jednoho pixelu i jeho okolí, čímž se zlepšuje přesnost na hranách objektů a snižuje se vliv šumu ve větších plochách. Nevýhodou je značný nárůst velikosti vstupního vektoru k vyhodnocení.

Před samotným vyhodnocením lze ztrátově redukovat velikost vstupní spektrální charakteristiku a využívat jen ty její části, které obsahují významnou informaci pro vyhodnocení. Při jejich použití je potřeba zvážit, zda způsobená ztráta informací výrazně neovlivní dosahovanou přesnost vyhodnocení.

Porovnání dvou spektrálních charakteristik (naměřené  $x$  a zástupce dané třídy  $y$ ,  $f(x, y)$ ) lze snadno vyhodnotit pomocí metrik Euklidovské vzdálenosti (3.1) nebo výpočtem Spektrálního úhlu (3.2). Pokročilejší metriky nejprve analyzují vyhodnocovanou množinu tříd a vypočtou jejich tzv. „kovarianční matici  $C$ “ (3.3), která reprezentuje vztahy mezi

nimi. Ta je využívána metrikou Mahalanobisovi vzdálenosti (3.4). Používaná je také metoda vyhodnocení K-mean, která nejprve analyzuje celou množinu dat, rozdělí ji na daný počet tříd a iteračním procesem rozdělení zpřesňuje. Obdobně lze využít statistickou klasifikační metodu Gaussian Maximum Likelihood [5].

### Euklidovská vzdálenost

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (3.1)$$

### Spektrální úhel

$$\theta(\mathbf{x}, \mathbf{y}) = \cos^{-1} \left( \frac{\sum_{i=1}^n x_i y_i}{\left(\sum_{i=1}^n x_i^2\right)^{\frac{1}{2}} \left(\sum_{i=1}^n y_i^2\right)^{\frac{1}{2}}} \right) \quad (3.2)$$

### Mahalanobisova vzdálenost

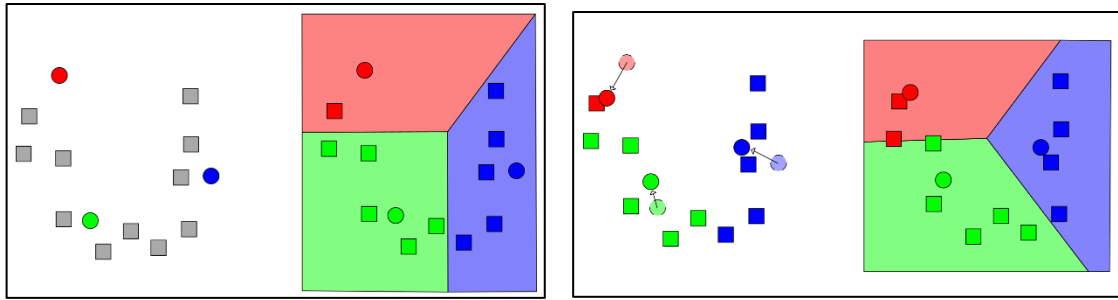
$$\mathbf{C} = \mathbf{X}^T \mathbf{X} = \mathbf{P} \mathbf{\Lambda} \mathbf{P}^T \quad \mathbf{P}^T \mathbf{P} = \mathbf{I} \quad (3.3)$$

$$\mathbf{MD}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^t \mathbf{C}^{-1} (\mathbf{x} - \mathbf{y})} \quad (3.4)$$

Matice průběhů zástupců tříd  $\mathbf{X}$  je rozložitelná na ortonormální matici  $\mathbf{P}$  (vlastní vektory kovarianční matice  $\mathbf{C}$ ) a matici  $\mathbf{\Lambda}$  (diagonální matice vlastních čísel matice  $\mathbf{C}$ ).

### K-mean

Analyzuje rozdělení množiny dat do předem daného počtu podmnožin a v několika iteracích zpřesní pozici těžiště podmnožiny (centroid), viz Obrázek 32 (pro tři centroidy). Nalezené centroidy lze porovnat se známými spektrálními průběhy z databáze NIST a identifikovat tak jednotlivé nalezené třídy. Výhodou tohoto algoritmu je, že nalezené centroidy, lépe odpovídají naměřeným spektrálním charakteristikám než ty, které používají u ostatní metriky.



**Obrázek 32 Ukázka metody K-mean**

Ukázka metody K-mean na množině dat (čtverečky) pro tři centroidy (kolečka) a jejich rozdělení při inicializační fázi (vlevo) a pro iterační krok (vpravo), při kterém se přepočítává poloha centroidů

### 3.2.1 Redukce spektrální charakteristiky

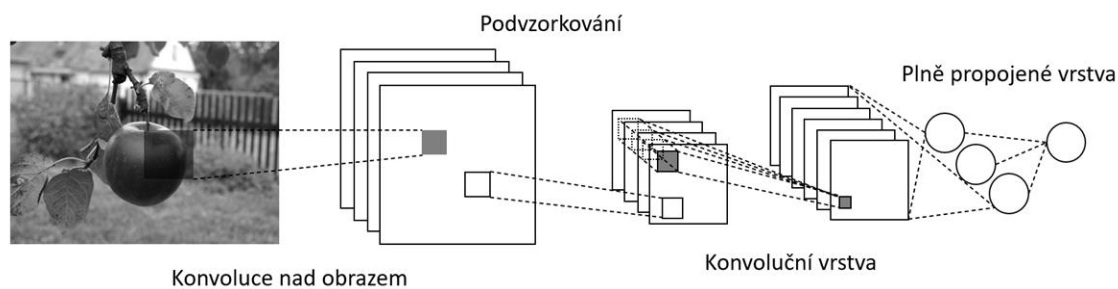
Naměřená spektrální charakteristika poskytuje detailní záznam v rámci celého měřeného spektra, avšak každá rozpoznávaná třída se od ostatních odlišuje jen v pár úzkých spektrálních částech. Ty lze identifikovat porovnáním zástupců jednotlivých tříd např.: metrikou Euklidovské vzdálenosti a vybrat pouze nejvýznamnější části spektra. Při významném zmenšení spektrální charakteristiky dojde ke ztrátě přesnosti vyhodnocení a také k větší náchylnosti na šum.

Lepší redukce je pomocí metody analýzy hlavních komponent (Principal Component Analysis, PCA), která využívá již zmíněnou „kovarianční matici“ (3.3). Její rozklad obsahuje diagonální matici vlastních čísel  $\Lambda$ , kterou lze oříznout a ponechat pouze část s  $N$  největšími čísly. Tím vytvoříme menší transformační matici  $C_Y$ , jejímž vynásobením se vstupní spektrální charakteristikou vznikne vektor příznaků o zvolené velikosti  $N$  vhodný pro následné vyhodnocení. Příznaky se nepodobají původnímu průběhu charakteristiky, ale zpětným vynásobením inverzní maticí  $C_Y^{-1}$ , lze získat jejich obdobný degradovaný průběh.

### 3.3 Neuronové sítě

Intenzivně zkoumanou alternativou k základním vyhodnocovacím algoritmům jsou neuronové sítě [22] [23]. Ty vytváří naučený model mezi vstupy a výstupy systému na předem připravených vyhodnocených a ověřených množinách vzorků. Pomocí kaskády vrstev vzájemně provázaných neuronů se přiřadí daný vstupní vektor do některé z definovaných výstupních tříd. Každý neuron sčítá s určitou natrénovanou vahou výstupy neuronů z předchozí vrstvy, které jsou přepočteny nelineární aktivační funkcí. Její výsledek je následně vstupem do příslušných neuronů v následující vrstvě. Trénování vah sítě je zdlouhavý proces, který pro dosažení vysoké přesnosti vyhodnocení vyžaduje ohromné množství vzorků a využití pokročilých minimalizačních metod a je počítaný zejména na GPU nebo výpočetních clusterech. Následný opakovaný výpočet výstupu pro nové vstupní vzorky „inference“ je výrazně rychlejší než učící fáze, a lze ji optimalizovat pro jednotlivé výpočetní platformy (CPU,

GPU, FPGA) [24]. Ukázka malé konvoluční neuronové sítě se dvěma konvolučními vrstvami je viz Obrázek 33.



Obrázek 33 Ukázka konvoluční neuronové sítě

### 3.3.1 Redukce velikosti neuronové sítě

Velikost neuronové sítě je daná počtem jejích vah a jejich datovým formátem (Float32). Ten poskytuje velký rozsah hodnot a umožňuje vysokou přesnost jednotlivých výpočtů. Znamená to ale i, že každá váha zabere 4 B v paměti. Při zvyšujícím se výkonu (operací za sekundu) výpočetních platformech tak narůstají nároky na přenosové rychlosti propojovacích sběrnic. Tyto nároky lze snížit redukcí počtu vah „Pruning“ a převedením vah do datového formátu s menším rozsahem hodnot „Quantization“ [25] [26].

#### Pruning

Jedná se o iterační proces, který nuluje nejmenší váhy neuronové sítě a následně opakuje proces učení sítě. Datový typ vah sítě zůstává Float32 nebo Float16 pro zachování potřebných přesností. Dělí se na dva základní typy:

- **Redukce počtu vah sítě** – Nahrazuje nízké váhy nulou a vytváří tak řídké matice vah, pro jejichž efektivnější výpočet je potřeba využít speciální funkce, které nejsou na některých platformách podporovány.
- **Redukce počtu neuronů** – Vychází z redukce počtu vah sítě a analyzuje podobnost neuronových propojení a odstraňuje jejich duplicitu. Vypočítává bázi řídké matice vah neuronů dané vrstvy a odstraní tak její lineárně závislé členy. Díky tomu lze upravit architekturu sítě zmenšením dané vrstvy. Výhodou je, že výsledná síť nemá příliš řídké matice vah, a nepotřebuje tak speciální výpočetní funkce.

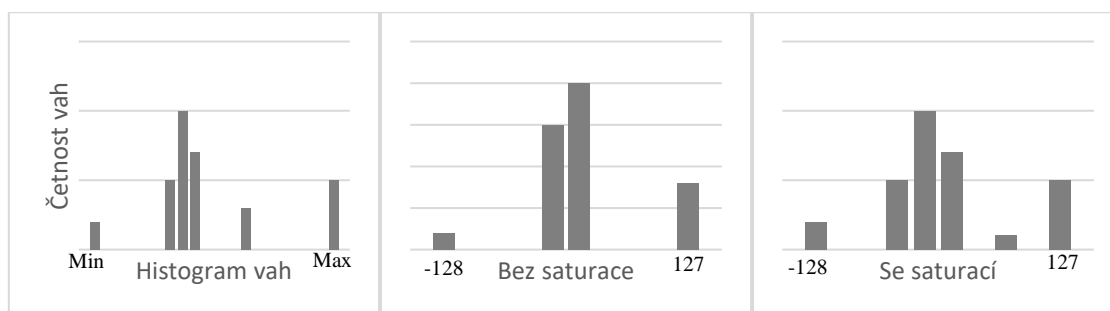
#### Kvantizace

Převádí datový formát vah sítě na formát s menším rozsahem hodnot, přičemž výpočetní formát dané platformy je potřeba adekvátně přenastavit. Paměťová náročnost sítě klesne v případě převedení na formát Float16 (dvakrát menší) nebo na Int8 (čtyřikrát menší) v extrémním případě lze přejít na 1 bit (binární neuronové sítě). Při konverzi vah do menšího rozsahu hodnot,



může být pokles přesnosti sítě velmi významný, zejména pokud má některá z vrstev velký rozptyl hodnot.

Používají se dva přístupy ke konverzi formátů, viz Obrázek 34. První přístup je „bez saturace“, který převádí krajní hodnoty starého rozsahu na krajní hodnoty rozsahu nového, přičemž dojde ke zkreslení rozdílu středových hodnot. Druhý přístup „se saturací“ zohledňuje rozdíly hodnot uprostřed rozsahu a krajní hodnoty jsou zdánlivě v saturaci, tj. nemají od středových hodnot takový odstup.



Obrázek 34 Změna histogramu po konverzi do menšího rozsahu hodnot

### 3.4 Porovnání dosažených přesností

Základní metrikou úspěšnosti vyhodnocení je celková přesnost (Overall accuracy, OA), která představuje poměr mezi správně rozpoznányými vzorky ku celkovému počtu vzorků v testovací množině. Pro nerovnoměrnou testovací množinu více tříd se OA počítá pro každou třídu zvlášť a výsledné přesnosti se průměrují. Další používané metriky upřednostňují některou z vyhodnocovaných tříd a počítají poměr pozitivních výsledků dané třídy ku počtu všech pozitivních výsledků (Přesnost, Precision, P), nebo poměr pozitivních výsledků dané třídy ku počtu skutečně správných výsledků (Úplnost, Recall, R), popřípadě jejich kombinace (F1-score, F1).

#### 3.4.1 Vytvořené testovací množiny

Každý vzorek ze zmíněných dostupných hyperspektrálních kostek byl náhodně rozdělen do jedné ze tří množin: trénovací (20 %), verifikační (10 %) a testovací (70 %), viz Tabulka 2. Pro jednotlivé množiny byly vygenerované i varianty s minimalizovanou spektrální charakteristikou a se zahrnutým okolím (od 3×3 do 7×7).

Tabulka 2 Testovací množiny

Hyperspektrální kostka	Velikost spekter	Počet vzorků	Počet tříd
IndianPines	220	10 000	16
PaviaU	103	40 000	9
Filtry	440	6 000	4
FiltryNC	440	6 000	4

Výsledky vyhodnocení testovaných vzorků základními metodami jsou sepsané viz Tabulka 3. Za dostatečně přesné je považováno vyhodnocení s přesností OA nad 95 %.

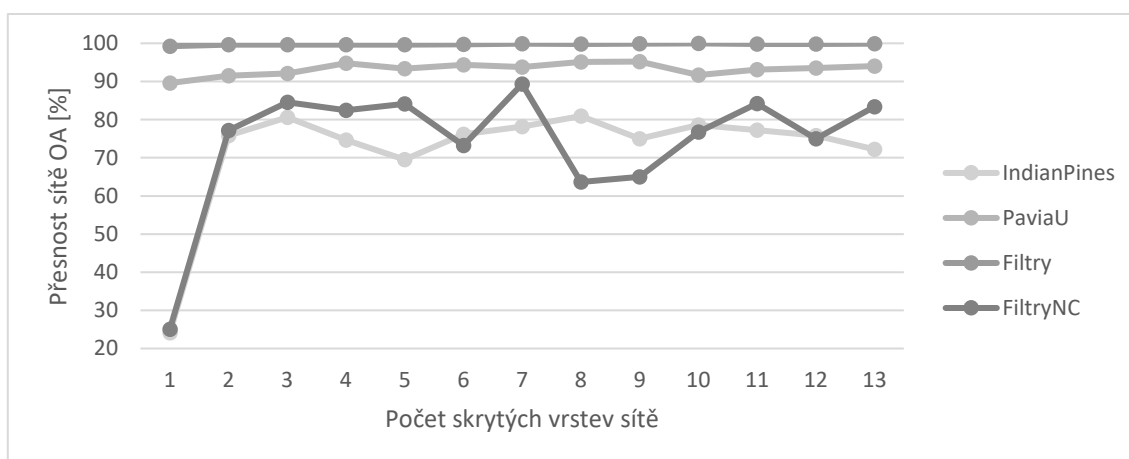
Tabulka 3 Dosažené přesnosti vyhodnocení testovacích vzorků

Metoda vyhodnocení	IndianPines OA [%]	PaviaU OA [%]	Filtry OA [%]	FiltryNC OA [%]
Euklidovská vzdálenost	53	68	57	33
Spektrální úhel	60	70	58	33
Mahalanobisova vzdálenost	52	68	57	33
K-mean	46	67	47	25
NN	83	97	99	88
CNN	81	94	99	87

Neuronové sítě dosáhly výrazně lepších přesností než původní algoritmy pro všechny testované vzorky. Zejména pak umožňují vyhodnocení snímků z hyperspektrální kamery bez aplikace korekcí, jak je vidět na výsledcích vzorku „FiltryNC“.

### 3.4.2 Vliv hloubky neuronové sítě

Větší počet vnitřních vrstev navyšuje velikost sítě a dobu jejího učení. Dosažená přesnost se nemusí zlepšovat s každou přidanou vrstvou, protože byly použity stejné učící parametry, jejichž optimalizace pro trénování hlubokých sítí není předmětem této práce. Dosažené přesnosti pro NN jsou viz Obrázek 35.

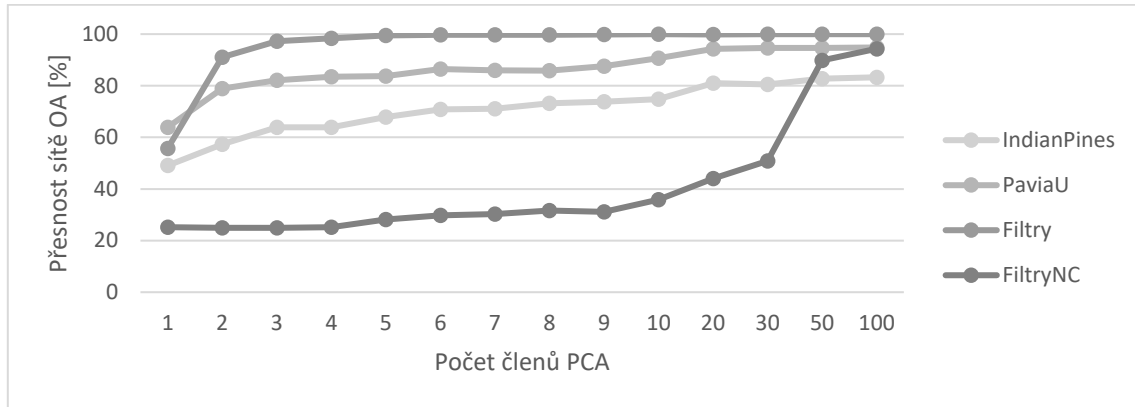


Obrázek 35 Vliv počtu vrstev sítě na její přesnost

Zvýšení počtu skrytých vrstev neuronových sítí mírně zvýšilo dosažené přesnosti a dosáhlo saturace přesnosti při struktuře sítě se čtyřmi vnitřními vrstvami. Pro následné experimenty byly testovány struktury sítě s až šesti vnitřními vrstvami (pro větší okolí pixelů až 10 vrstev).

### 3.4.3 Vliv redukce spektrální charakteristiky

Oříznutí spektrální charakteristiky významně snižuje velikost první vrstvy neuronové sítě, což je nejznatelnější u sítí s malým počtem vrstev. Zároveň s tím je snižována dosažená přesnost sítě, zejména pro redukce charakteristiky pod počet rozlišovaných tříd. Změna dosažené přesnosti sítí při redukci metodou PCA je viz Obrázek 36.



Obrázek 36 Dosažené přesnosti sítí při redukci metodou PCA

Porovnání dosažení přesností sítí před a po redukci spektrální charakteristiky metodou PCA s 50 ti členy a s N členy (počet rozpoznávaných tříd) je viz Tabulka 4.

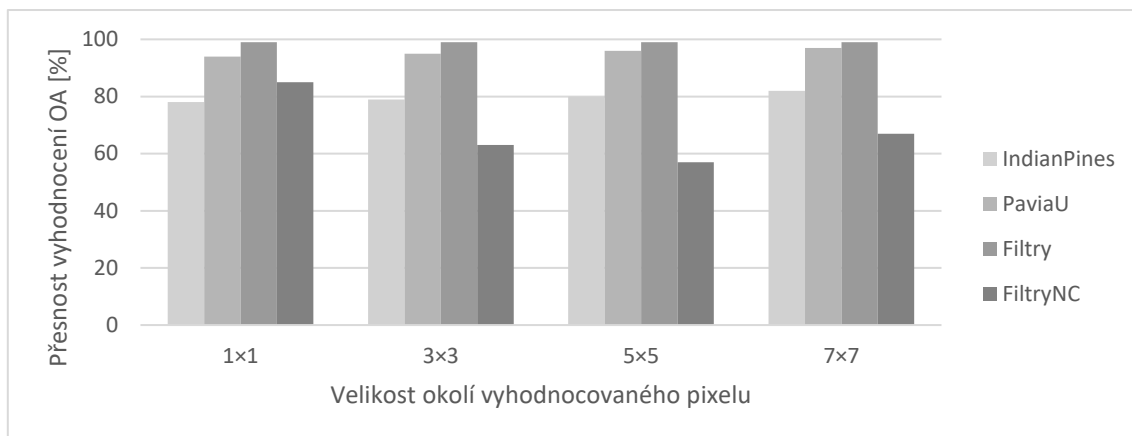
Tabulka 4 Dosažené přesnosti vyhodnocení redukovaných testovacích vzorků

Metoda redukce	IndianPines OA [%]	PaviaU OA [%]	Filtry OA [%]	FiltryNC OA [%]
Původní NN	83	97	99	88
PCA 50	82	94	99	89
PCA N	81	90	98	31
Max N	74	84	87	53

Pro testované hyperspektrální vzorky byla ztráta přesnosti s PCA metodou (~2 %) a s výběrem významných pásem (Max) byla ztráta o něco větší (~10 %).

### 3.4.4 Vliv zahrnutí okolí pixelu

Pro zpřesnění výsledků byly vstupní spektrální charakteristiky k vyhodnocení naopak rozšířeny o jejich nejbližší okolí 3×3 až 7×7 a jejich dosažené nejlepší výsledky jsou viz Obrázek 37.

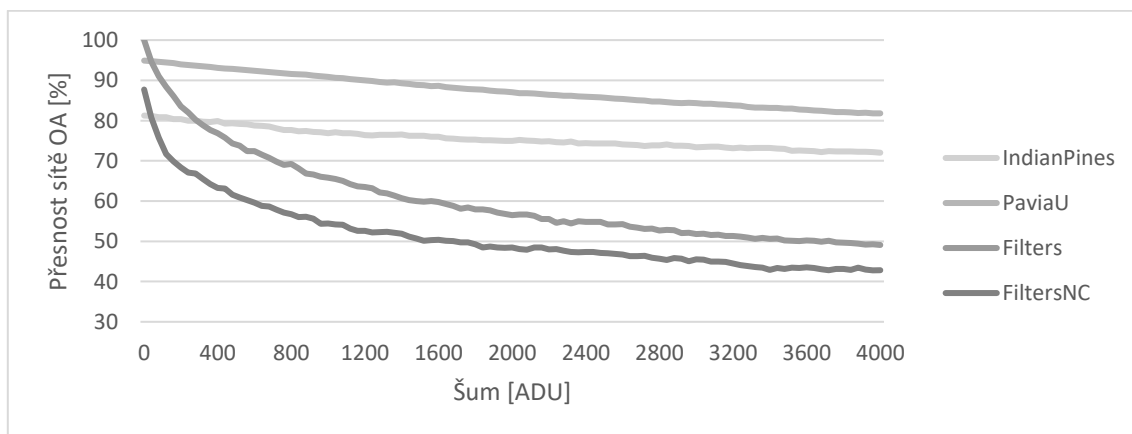


Obrázek 37 Vliv zahrnutí okolí na přesnost sítě

Díky zahrnutí okolí bylo dosaženo zlepšení (IndianPines) v řádu několika procent, což není příliš dramatická změna. Nicméně čím bližší je úspěšnost vyhodnocení ke 100 %, tím cennější je každé zlepšení, byť jen o pár procent.

### 3.4.5 Citlivost vyhodnocení na šum vstupních dat

Vlastnosti kamery jsou závislé na okolním prostředí, čímž se zanáší šum do měřených snímků v hodnotách stovek až tisíců ADU. Částečně je tento šum eliminován průměrováním snímků, ale tím se významně prodlužuje doba záznamu scény. Natrénované neuronové sítě byly otestovány množinou zatíženou bílým šumem až 4000 ADU, což odpovídá poloviční úrovni vyhodnocovaného signálu. Průběh ztráty přesnosti pro jednotlivé vzorky je viz Obrázek 38.



Obrázek 38 Vliv zatížení šumem na přesnost sítě

Ztráta přesnosti byla velmi významná pro laboratorní vzorky filtrů oproti zbylým významně větším vzorkům, u kterých byla ztráta přesnosti v jednotkách procent. Způsobeno to je zejména tím, že vybrané trénovací vzorky sítě jsou příliš hladké, přičemž více zašuměné okraje filtrů byly z použitých (ohodnocených) množin vyňaty.

## 4 Porovnání výpočetních platforem

Výkon platforem se udává v počtu operací za sekundu (OPS) a pokud se jedná o operace ve float formátu (FLOPS). Dnešní dostupné platformy mají výkon v řádu GOPS až TOPS.

Výpočet neuronových sítí a ostatních testovaných algoritmů vyžaduje pro svůj výpočet zejména násobení dvou matic ( $N \times N$ ), které vede na výslednou matici o stejné velikosti ( $N \times N$ ), a pro výpočet vyžaduje  $N^3$  operací násobení a sčítání. Přitom je potřeba načíst z paměti  $2 \cdot N^3$  hodnot, při možnosti jejich uložení do mezipaměti jen  $2 \cdot N^2$  hodnot. Tím se vytváří vysoké výpočetní nároky při počítání velkých matic a vysoké paměťové nároky při počítání malých matic. Optimalizované metody snižují výpočetní náročnost až na  $N^{2.37}$  [27]. Rychlost výpočtů platformy a bitová šířka operandů pak vytváří nároky na propustnost propojovacích sběrnic, které v případě výpočtu menších matic snadno dosáhnou svých limitů.

Pro jednotlivé výpočty využívám prostředí Python s modulem Tensorflow, které disponuje podporou pro všechny testované platformy. Pro FPGA implementaci navíc vytvářím celý design, který umožňuje výpočet neuronových sítí příslušnými nástroji (Deep Neural Network Development Kit, DNNDK), nově známé jako Vitis AI. Postupně se ale podpora nástrojů rozšiřuje i na prostředí Matlab a Pytorch.

### Testované výpočetní platformy:

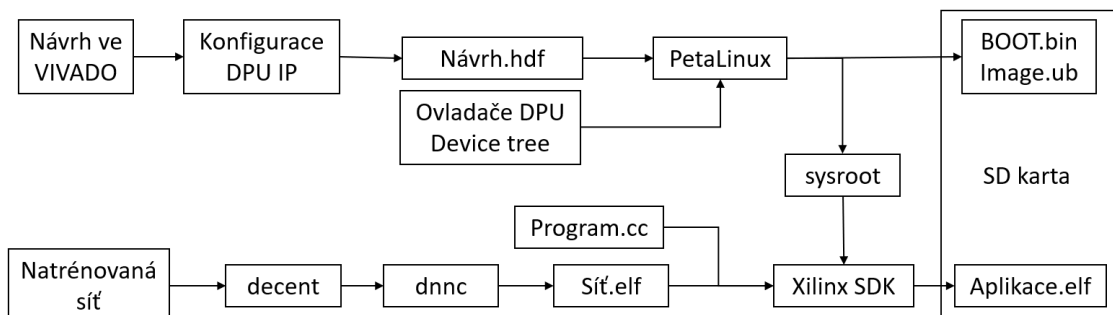
- **Platforma CPU** – Procesor řady X-series s označením Intel Core i7 7820X s 8 jádry a 16 vláken s maximální frekvencí až 4.3 GHz a spotřebou až 140 W. Inference se provádí pomocí nástrojů Tensorflow pro CPU s modulem Keras.
- **Platforma GPU** – Grafická karta Nvidia Geforce RTX 2070 s 8 GB RAM a maximálním výkonem 42 RTX-OPS a spotřebou až 185 W. Inference se provádí pomocí nástrojů Tensorflow pro GPU s modulem Keras a knihoven CUDA 10.2.
- **Platforma FPGA** – Vývojový kyt ZCU106 s FPGA Xilinx Zynq UltraScale+ XCZU7EV MPSoC, který má dostupných 504k LUT, 1728 DSP bloků a spotřebu až 20 W. Inference se provádí pomocí nástrojů Xilinx Vitis AI a implementovaným výpočetním jádrem DPU [9].

### 4.1 Příprava FPGA

Zatímco pro provedení a změření rychlosti inference neuronových sítí na platformách CPU a GPU postačuje úprava programů vytvořených už při jejich trénování, pro platformu FPGA je potřeba nejprve vytvořit design s nakonfigurovaným akcelerátorem DPU a k němu vygenerovat nastavený obslužný operační systém PetaLinux [10]. Teprve potom je možné převést

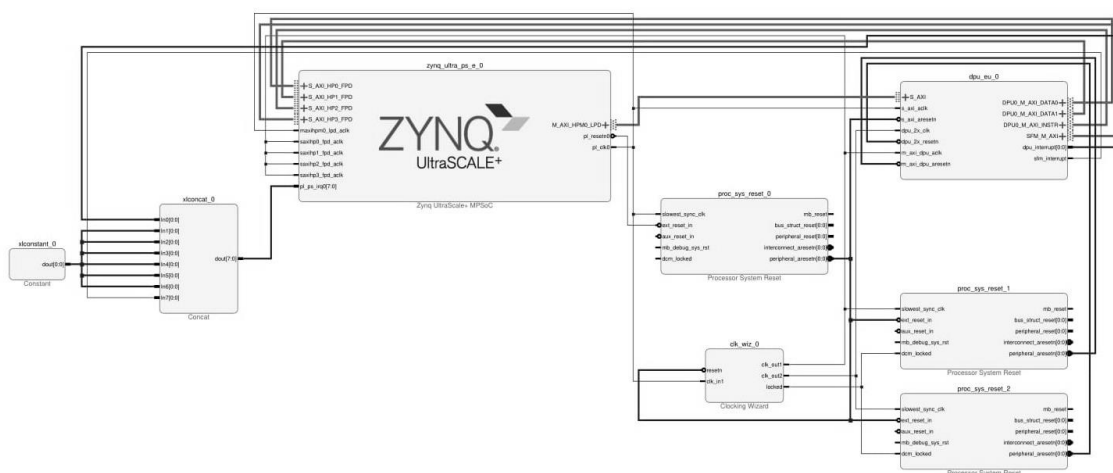
natrénovanou neuronovou sít' do správného inferenčního formátu a otestovat ji pomocí C/C++ testovacího programu.

Návrh FPGA se provádí ve vývojovém studiu Vivado, ve kterém se zapojí všechny potřebné bloky (Intellectual property, IP), zejména blok DPU. Následně se vygeneruje soubor „Návrh.hdf“ obsahující vytvořenou konfiguraci pro cílové FPGA. Ten je společně s dalším nastavením potřebný pro kompilaci OS PetaLinux a vytvoření spouštěcích souborů „boot.bin“, „image.ub“ a kompilačních souborů „sysroot“. Ty jsou později potřebné k tvorbě spustitelných testovacích programů na FPGA. Testovanou neuronovou sít' je potřeba nejprve kvantizovat nástrojem „decent“ a poté ji lze převést do inferenčního formátu „Síť.elf“ nástrojem „dnnc“ pro zvolenou konfiguraci DPU. Testovací aplikace „Main.cc“ vyvíjená v programovacím studiu „Xilinx SDK“ s příloženými soubory vytvoří spustitelný testovací program „Aplikace.elf“ pro příslušné FPGA (OS). Všechny konečné vytvořené soubory se nahrají na SD kartu, ze které se celý systém pustí. Ten je pak možné ovládat přes UART nebo Ethernet (SSH) rozraní. Celý proces je pro názornost vyobrazen viz Obrázek 39 a jeho časová náročnost sestavení systému je v řádu hodin.



Obrázek 39 Postup přípravy FPGA a převedení NN sítě

Vytvořený návrh designu ve studiu Vivado pro vývojovou desku ZCU106 je viz Obrázek 40. IP blok DPU je do procesního systému FPGA připojen jednou „master AXI“ sběrnici a dvěma „slave AXI“. Celý systém běží na základní frekvenci 300 MHz a do DPU jsou navíc přivedeny ještě jedny dvojnásobné hodiny, tj. 600 MHz.



Obrázek 40 Návrh zapojení FPGA s DPU IP

Úspěšně byly vygenerované konfigurace s jedním DPU jádrem (pojmenované podle jejich velikosti), jejichž detaily jsou uvedené viz Tabulka 5. Při generování bylo často výsledné časování systému nevyhovující, což vedlo k nevalidním výsledkům inference neuronové sítě, a proto bylo potřeba generování zkoušet vícekrát.

Tabulka 5 Vytvořené designy s DPU IP

Design	LUT [k]	FF [k]	BRAM	DSP	Spotřeba [W]
DPU_512	38	42	93	124	5.7
DPU_800	41	49	113	171	6.3
DPU_1024	45	56	140	232	6.6
DPU_1152	43	55	149	226	6.5
DPU_1600	59	67	167	326	7.8
DPU_2304	53	77	215	436	8.7
DPU_3136	57	89	266	562	9.4
DPU_4096	72	109	310	704	13.0

Pro každý vygenerovaný design FPGA je potřeba vytvořit spustitelný obraz PetaLinuxu, vygenerovat inferenční formát pro každou z testovaných sítí a přeložit jí s testovací aplikací, nahrát na SD kartu a otestovat.

## 4.2 Optimalizace neuronových sítí

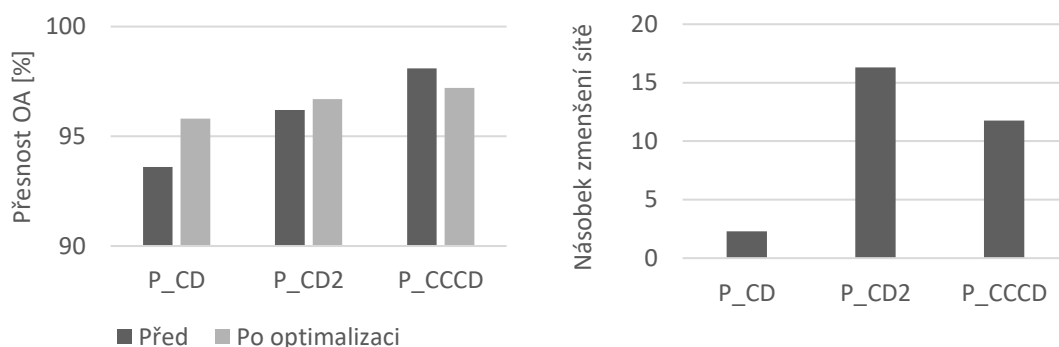
Využití neuronových sítí pro vyhodnocení hyperspektrálních snímků prokázalo svoji kvalitu oproti jiným metodám. Pro ověřování výkonosti platform byly vybrány natrénované konvoluční neuronové sítě pro vzorek „PaviaU“, které byly dále zmenšeny pruningem a kvantizovány do INT8 formátu, jejich vlastnosti a dosažené přesnosti jsou rozepsány viz Tabulka 6. Výsledky sítí s čistě plně propojenými vrstvami i po redukci velikosti byly významně větší (~10×), proto byly z výsledků a následných testů vynechány.

Vybrané sítě mají strukturu: jedna konvoluční a plně propojená vrstva „P\_CD“, kaskáda tří konvolučních a jedné plně propojené vrstvy „P\_CCCD“ a poslední síť „P\_CD2“, která má větší posun mezi konvolucemi než „P\_CD“. Všechny sítě mají jednotnou velikost konvolučního jádra 3×3.

Tabulka 6 Porovnání optimalizovaných CNN

Sít'	Operací [M]	Váhy [M]	OA [%]	OA po kvantizaci [%]
P_CD	4.4	2.2	96.3	93.6
P_CD_p	1.9	0.9	97.5	95.8
P_CD2	3.6	1.8	97.8	96.2
P_CD2_p	0.2	0.1	97.6	96.7
P_CCCD	6.8	3.4	98.7	98.1
P_CCCD_p	0.6	0.3	97.0	97.2

Testované sítě byly zmenšeny 2× až 16×, přičemž se i lépe natrénovaly a dosahují přesnosti srovnatelné či lepší než původní neoptimalizované sítě, viz Obrázek 41. Samotná kvantizace sítí ubrala až 2 % jejich přesnosti.



Obrázek 41 Porovnání vlastností optimalizovaných sítí

### 4.3 Výkonnost platform

Na jednotlivých platformách jsem realizoval měření inference připravených neuronových sítí, viz Tabulka 7. Měřený byl zvláště první provedení cyklus výpočtu testovací množiny (tzv. Cold run, 30k vzorků na cyklus) a poté stejných 20 cyklů výpočtů pro věrohodnější průměrovaný čas výpočtu (tzv. Hot run). Při prvním cyklu se výrazně projevují inicializační procesy platformy, jako je například nahrávání potřebných parametrů do patřičných pamětí a plnění výpočetních front (pipeline).

Tabulka 7 Rychlost inference neuronových sítí na různých platformách

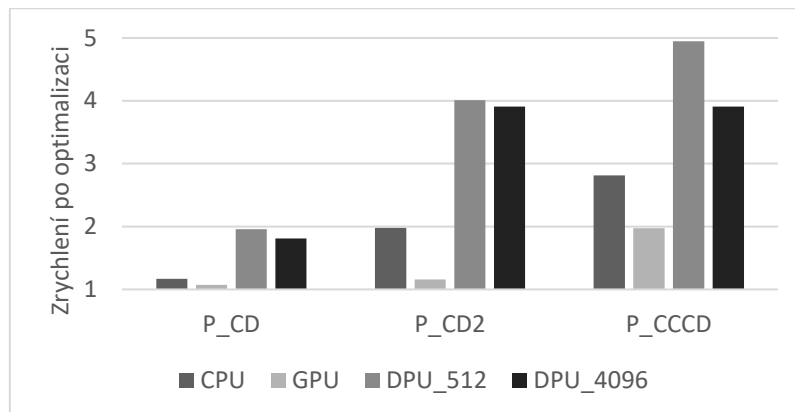
Sít'	CPU [ $\mu$ s]	GPU [ $\mu$ s]	DPU_512 [ $\mu$ s]	DPU_4096 [ $\mu$ s]
P_CD	119 (138)	15.5 (165)	539 (580)	350 (420)
P_CD_p	105 (118)	14.1 (154)	228 (296)	162 (232)



<b>P_CCCD</b>	501 (509)	25.2 (373)	883 (965)	526 (610)
<b>P_CCCD_p</b>	174 (181)	15.0 (189)	123 (195)	84 (156)
<b>P_CD2</b>	240 (253)	16.0 (166)	420 (493)	284 (354)
<b>P_CD2_p</b>	121 (128)	13.9 (143)	55 (123)	46 (114)

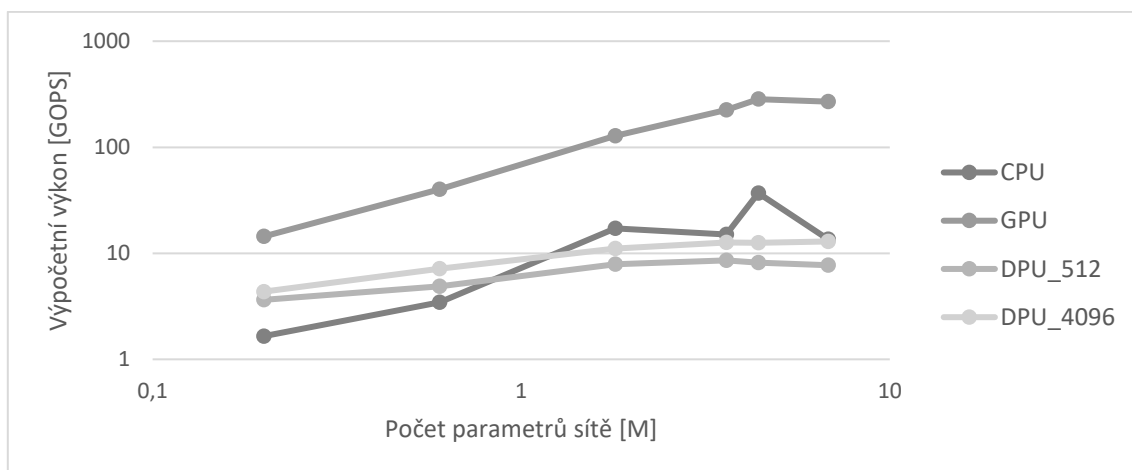
Uvedený čas „Hot run“ a v závorce „Cold run“ pro vyhodnocení jedné spektrální charakteristiky

Porovnání zrychlení inference sítě před a po optimalizování sítí na jednotlivých platformách je viz Obrázek 42. Na něm je patrné nejvýraznější zrychlení u FPGA platformem a pro největší síť „P\_CCCD“ i u ostatních platformem.



Obrázek 42 Porovnání zrychlení inference optimalizovaných neuronových sítí

Díky většímu množství testovaných sítí na jednotlivých platformách, které díky redukcí jejich velikosti pokrývají rozsah počtu parametrů sítě od stovek k desetitisícům, lze vyobrazit výkonnostní křivku platformy, viz Obrázek 43. Na ní je patrný postupný růst výkonu pro větší síť způsobený omezením na straně paměti platformy, až po saturaci výpočetního výkonu platformy.



Obrázek 43 Porovnání rychlosti výpočtu jednotlivých platformem

Nejmenší pokles výkonu pro menší síť mají realizované FPGA platformy, čímž potvrzují svoji variabilitu pro výpočet malých i velkých sítí. Prezentované výsledky byly rovněž publikované v [E].

#### 4.4 Nároky systému RODES

Pro vyhodnocování v reálném čase je potřeba vyhodnotit 300 spektrálních charakteristik na snímek, kterých je zaznamenáno 50 za sekundu. Požadovaný výkon platformy je tedy minimálně vyhodnocení 15000 vzorků za sekundu. Dosažený výkon platformem pro testované sítě je viz Tabulka 8.

Tabulka 8 Počet vyhodnocených vzorků za sekundu na jednotlivých platformách

Sít'	CPU	GPU	DPU_512	DPU_4096
P_CD_p	9500	71000	4400	6200
P_CCCD_p	5800	67000	8200	12000
P_CD2_p	8300	72000	18000	22000

Dosažené výpočetní rychlosti na CPU nejsou pro zpracování v reálném čase dostačující, oproti tomu výkon GPU je více než 4× vyšší, než by bylo potřeba. Výkon FPGA se pohybuje velmi blízko kolem potřebné hodnoty a pro testovanou síť „P\_CD2\_p“ o velikosti 0,2M parametrů, bylo dosaženo cíleného výkonu pro největší i nejmenší implementovanou konfiguraci DPU.

## Závěr

Úspěšně byly natrénovány neuronové sítě řešící úlohu vyhodnocování hyperspektrálních snímků z testovaného hyperspektrálního systému RODES, které byly dále optimalizovány a implementovány pro výpočet na FPGA platformě. Celý testovaný systém byl řádně identifikován a zkalibrován tak, aby poskytoval snímky srovnatelné s ostatními hyperspektrálními systémy v podobě hyperspektrálních kostek. Výsledkem zkalibrovaného systému je hyperspektrální záznam scény ve spektrálním rozsahu LWIR (7–14  $\mu\text{m}$ ) rozloženým přes 400 pixelů pro 300 prostorových pixelů. Tímto je postupně oskenovaná celá scéna, tj. po proužkách, rychlostí až 50 snímků za sekundu.

Jádrem hyperspektrální kamery je nechlazený infračervený bolometrický senzor a procesní modul s platformou FPGA Zynq od Xilinx. Tím jsou dány malé rozměry a poměrně vysoký šum systému v porovnání s ostatními systémy, zejména těmi založenými na chlazených senzorech. Výhodou použitého systému je přímé ovládání senzoru a výčet nezpracovaných „surových“ snímků, převoditelných na radiometrickou informaci. Tím je umožněno navázání vyhodnocování měřených hyperspektrálních snímků přímo na jejich vyčítání ze senzoru a zpracovávat je v řídicí FPGA platformě systému.

Práce kladla důraz na dílčí výstupy záznamu hyperspektrálních snímků, jejich kalibraci a interpretaci, tak aby bylo možné kontrolovat jejich správnost a určit jejich vlastnosti. Díky tomu bylo možné odhalit chyby nastavení systému a jeho limity. Zároveň byly testovány obecně používané vyhodnocovací metody a vyhodnocení pomocí neuronových sítí, pro které bylo potřeba nejprve získat množiny anotovaných průběhů pro potřeby jejich učení a testování. Natrénované neuronové sítě byly poté optimalizovány a převedeny na FPGA platformu, kde byla ověřena jejich funkčnost a změřen jejich výkon v porovnání s ostatními dostupnými platformami.

Největší obtížící úlohy vyhodnocení byla nízká citlivost použitého systému RODES, která umožňovala vyhodnocování pouze laboratorních vzorků zaznamenaných proti vysoce teplému pozadí (černé těleso o teplotě 300 °C). Nejnovější upravený hyperspektrální systém má výrazně větší citlivost, což umožňuje jeho využití i za běžných podmínek, pro který byly výstupy práce aktualizované.

Další komplikací byl velmi rychle se vyvíjející nástroje pro zpracování hyperspektrálních snímků a trénování neuronových sítí (Matlab, Lua Torch, Python Tensorflow) a nové nástroje pro akceleraci výpočtu (Vitis AI) na FPGA. Pro který existují předpřipravené designy, bohužel pro jiné vývojové desky, než které byly k dispozici. Proto bylo

potřeba vytvářet design od samého začátku, přičemž nové verze používaných nástrojů, opravujících chyby verzí předchozích, zároveň měnili i celý systém generování designu. Byly tak nalezeny realizovatelné konfigurace výpočetního akcelérátoru pro dostupnou vývojovou desku, a o to snadněji i pro FPGA modul použitý v systému RODES.

Byly vyvinuté a ověřené metody identifikace šumu a kalibrace hyperspektrální kamery, které pomáhají vybrat aktivní region senzoru kamery. Tím je efektivně zmenšena délka vyhodnocované spektrální charakteristiky z maximální možné délky 640 px na 400 px. Byly pořízeny spektrální záznamy testovacích filtrů a některých chemických látek pro běžné teploty 0-100 °C v laboratorních podmínkách. Naměření většího množství vzorků chemikálií se nezdařilo, z důvodů jejich nízkého odparu za běžných podmínek a tím pádem i jejich nízké koncentrace, dostupným systémem neměřitelné.

Vyhodnocovací metody založené na principu porovnávání vzorků vybranou metrikou a pomocí neuronových sítí byly testovány na obecných hyperspektrálních vzorkách „IndianPines“ a „PaivaU“ společně s naměřenými snímky ze systému RODES. Na nich byla demonstrována efektivita použitých metod pro reálné pracovní podmínky systému. Zároveň byly zkoumány vlastnosti jednotlivých modifikací metod, jako je zahrnutí okolí vyhodnocovaného bodu ve scéně, redukce vstupní spektrální charakteristiky metodou PCA, změna struktury neuronových sítí, počet jejich vrstev a využití konvolučních neuronových sítí.

Jednotlivé metody vyhodnotili testované vzorky s přesností kolem 70 % a pomocí natrénovaných neuronových sítí bylo dosaženo přesností 80 % až 95 %. Srovnatelné výsledky byly i pro konvolučních neuronové sítě, které byly následně použity pro testování jednotlivých platforem. Zlepšení výsledků bylo dosaženo při zahrnutím okolí vyhodnocované charakteristiky (3×3) a s počtem vnitřních vrstev 3 až 5.

Natrénované neuronové sítě byly optimalizovány systematickou redukcí počtu jejich vah „pruning“ metodou, čímž byl významně snížen počet operací potřebných pro výpočet sítě (až 15×). Pro potřeby výpočtu na FPGA byly váhy sítě převedeny na datový typ Int8, čímž se snížila i paměťová náročnost sítě. Redukované sítě dosáhly lepších přesností vyhodnocení než sítě s použitou redukcí vstupů metodou PCA. Nejmenší dostatečně přesná síť byla konvoluční a obsahovala pouhých 200k vah.

Optimalizované sítě byly testované na dostupných CPU, GPU a na několika konfiguracích FPGA platformy. Nejmenší testovaná konfigurace pro FPGA je použitelná i pro systém RODES, ostatní konfigurace vyžadují větší FPGA, spadající do kategorie střední výkonnostní třídy.

Dosažené zrychlení výpočtu sítě po minimalizaci bylo pro FPGA (až 5×), pro ostatní platformy výrazně méně (~2×), nicméně v absolutních číslech byl výpočet na GPU stále 3× rychlejší než na FPGA. Díky většímu počtu vypočtených sítí různých velikostí a obdobné architektury, bylo možné porovnat výkon jednotlivých platform. To vyšlo velice příznivě pro FPGA jejichž výkon klesal pro menší sítě výrazně pomaleji než u ostatních platform. Pokud by se podařilo ještě více minimalizovat velikost sítí, mohlo by se dosáhnout srovnatelných výkonů jako s využitím moderních GPU. Tomuto výsledku napomáhá zejména systém vyhodnocování prováděný po jednotlivých spektrálních charakteristikách tak, jak jsou čteny ze senzoru, a nikoliv hromadné zpracování již naměřených snímků.

Komericializace výstupů práce je závislá na využití FPGA platformy s operačním systémem PetaLinux (nebo Ubuntu). Pro integraci do systému RODES je potřeba převést stávající FW na ovladače využívaných IP bloků pro daný OS a určit způsob integrace a práce s DPU IP. Výhodou pro systém RODES je pak i lepší podpora komunikačních protokolů (USB, Gen-i-Cam), možnost využití více knihoven a programovacích jazyků (Python), širší podpora periférií (display, vzdálená zprava systému, ...).

Díky rapidně se vyvíjejícím technologiím Vitis AI, se rozrůstá i podpora vývojových nástrojů (Dark Net, Gstreamer), nových typů vrstev sítí (3D konvoluce) a probíhá i optimalizace a integrace DPU pro Versal ACAP (série AI Edge a AI Core). Obdobou používané výpočetní platformy v systému RODES je platforma Kria [28], pro kterou je vytvořen i online obchod licencovaných aplikací s jednotnou mechanikou využití DPU a dalších akceleratorů pro nejrůznější aplikace (rozpoznání obličeje, detekce chodců a další vyhodnocování objektů), do kterého můžou uživatelé přispívat i svým řešením.

## Citovaná literatura

- [1] FLIR, „On-Camera Deep Learning Firefly DL,“ [Online]. Available: <https://www.flir.com/products/firefly-dl/>. [Přístup získán 2021].
- [2] COGNEX, „VisionPro Deep Learning SW,“ [Online]. Available: <https://www.cognex.com/products/deep-learning/visionpro-deep-learning>. [Přístup získán 2021].
- [3] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi a J. Chanussot, „Hyperspectral Remote Sensing Data Analysis and Future Challenges,“ *IEEE Geoscience and Remote Sensing Magazine*, sv. 1, č. 2, pp. 6 - 36, 2013.
- [4] Ing. Vladimír Hampl, „Hyperspektrální snímání,“ APPLIC s.r.o, [Online]. Available: <http://www.applic.cz/infratechnika/hyperspektralni-snimani/>. [Přístup získán 2022].
- [5] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Springer Science & Business Media, 2003.
- [6] K. Huang, S. Li, X. Kang a L. Fang, „Spectral–Spatial Hyperspectral Image Classification Based on KNN,“ *Sensing and Imaging*, sv. 17, č. 1, 2016.
- [7] C. González, A. P. S. Sánchez, J. Resano, D. Mozos a A. Plaza, „Use of FPGA or GPU-based architectures for remotely sensed hyperspectral image processing,“ *Integration*, sv. 46, č. 2, pp. 89-103, 2013.
- [8] N. W. Chemie, „<https://webbook.nist.gov/chemistry/>,“ National Institute of Standards and Technology (NIST). [Online]. [Přístup získán 2021].
- [9] Xilinx, „DPUv3E for Alveo Accelerator Card with HBM,“ [Online]. Available: <https://www.xilinx.com/developer/articles/dpuv3e-for-alveo-accelerator-card-with-hbm.html>. [Přístup získán 2021].
- [10] Xilinx, Inc., „PetaLinux Tools,“ [Online]. Available: <http://www.xilinx.com/tools/petalinux-sdk.htm>. [Přístup získán 2015].

- [11 „DaVinci™ Technology,“ Texas Instruments, 2008. [Online]. Available:  
] <http://www.thedavincieffect.com/>. [Přístup získán 2021].
- [12 Trenz, „Trenz Electronic SoC Modules with Xilinx Zynq-7000,“ [Online]. Available:  
] <https://shop.trenz-electronic.de/en/Products/Trenz-Electronic/TE07XX-Zynq-SoC/>.  
[Přístup získán 2021].
- [13 perClass, „perClass Mira,“ 2022. [Online]. Available:  
] <https://www.perclass.com/apps/perclass-mira>.
- [14 Ústav fyziky plazmatu AV ČR, v.v.i., „TOPTEC,“ Výzkumné centrum speciální optiky a  
] optoelektronických systémů, [Online]. Available: <http://www.toptec.eu/cs/>.
- [15 „Pico640Gen2,“ Lynred, [Online]. Available:  
] <http://www.lynred.com/products/pico640gen2>.
- [16 Ing. Vladimír Hampl, „LWIR kamery IRCA,“ APPLIC s.r.o., 2022. [Online]. Available:  
] <https://www.apPLIC.cz/infratechnika/lwir-kamery-irca/>.
- [17 Y. Cao a Y. Li, „Strip non-uniformity correction in uncooled long-wave infrared focal  
] plane array based on noise source characterization,“ *Optics Communications*, sv. 339, pp.  
236-242, 2015.
- [18 Quantum Design Inc., „LWIR spectral camera,“ [Online]. Available: [https://qd-](https://qd-europe.com/fileadmin/Mediapool/products/hyperspectral/pdf/Spectral_cameras_LWIR.pdf)  
] [europe.com/fileadmin/Mediapool/products/hyperspectral/pdf/Spectral\\_cameras\\_LWIR.p](https://qd-europe.com/fileadmin/Mediapool/products/hyperspectral/pdf/Spectral_cameras_LWIR.pdf)  
df. [Přístup získán 2022].
- [19 M. V. B. A. M Graña,  
] „[http://www.ehu.eus/ccwintco/index.php/Hyperspectral\\_Remote\\_Sensing\\_Scenes](http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes),“ Hyp  
erspectral Remote Sensing Scenes. [Online]. [Přístup získán 2021].
- [20 G. B. M. Pedersen, „<https://crs.hi.is>,“ Center for Remote Sensing (CRS). [Online].  
] [Přístup získán 2021].
- [21 AVIRIS, „<https://aviris.jpl.nasa.gov>,“ NASA. [Online]. [Přístup získán 2021].  
]

- [22 Y. Chen, H. Jiang, C. Li, X. Jia a P. Ghamisi, „Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks,“ *IEEE Transactions on Geoscience and Remote Sensing*, sv. 54, č. 10, pp. 6232-6251, 2016.
- [23 X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang a X. Huang, „Hyperspectral Image Classification With Deep Learning Models,“ *IEEE Transactions on Geoscience and Remote Sensing*, sv. 56, č. 9, pp. 5408-5423, 2018.
- [24 C. González, S. Sánchez, A. Paz, J. Resano, D. Mozos a A. Plaza, „Use of FPGA or GPU-based architectures for remotely sensed hyperspectral image processing,“ *Integration*, sv. 46, č. 2, pp. 89-103, 2013.
- [25 Y. Jiwei, X. Shen, J. Xing, X. Tian, H. Li, B. Deng, J. Huang a X.-S. Hua., „Quantization Networks,“ *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7300-7308, 2019.
- [26 T. Liang, J. Glossner, L. Wang, S. Shi a X. Zhang, „Pruning and Quantization for Deep Neural Network Acceleration: A Survey,“ 2021. [Online]. Available: <https://arxiv.org/abs/2101.09671>.
- [27 F. L. Gall, „Faster Algorithms for Rectangular Matrix Multiplication,“ *Foundations of Computer Science*, sv. arXiv, č. 1204.1111, pp. 514-523, 2012.
- [28 Xilinx, „Kria K26 System-on-Module,“ [Online]. Available: <https://www.xilinx.com/products/som/kria/k26i-industrial.html>. [Přístup získán 2021].



## Seznam publikací

### 2016

- A. M. Rozkovec and J. Čech, "Polynomial Based NUC Implemented on FPGA," 2016 Euromicro Conference on Digital System Design (DSD), 2016, pp. 670-673, doi: 10.1109/DSD.2016.31

### 2017

- B. J. Čech and M. Rozkovec, "Detecting LWIR filters using hyperspectral camera and neural networks," 2017 International Conference on Applied Electronics (AE), 2017, pp. 1-4, doi: 10.23919/AE.2017.8053578.

### 2018

- C. J. Čech and M. Rozkovec, "Selection of Ideal Operating Point for Infrared Camera System," 2018 International Conference on Applied Electronics (AE), 2018, pp. 1-4, doi: 10.23919/AE.2018.8501456.

### 2021

- D. J. Čech, J. Václavík, O. Mach, R. Doleček and J. Pleštil, "Robust System for Hyperspectral Imaging in the LWIR Spectral Region with Uncooled Thermal Detector," 2021 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2021, pp. 1-4, doi: 10.1109/WHISPERS52202.2021.9483954.
- E. J. Čech and M. Rozkovec, "Comparison of Performance of Optimized HSI CNN models on Desktop and Embedded Platforms," 2021 International Conference on Applied Electronics (AE), 2021, pp. 1-4, doi: 10.23919/AE51540.2021.9542900.

### 2022

- F. A. Noé, P. Vandame, T. Porchez, L. Apostol, J. Čech, K. McNamara, F. Martin, T. Sequies, T. McNamara, M. Barthélémy, J. Chanussot, „Assessing the potential of Qormino processor for embedded AI on board a CubeSat“, IEEE Journal on Miniaturization for Air and Space Systems, [podáno]

## **Životopis**

### **Osobní údaje**

Ing. Jiří Čech

Pohlaví: Muž

Datum narození: 10/08/1991

Bydliště: Nový Bor, 47301

Státní příslušnost: Česká republika

### **Pracovní zkušenosti**

2014–do současnosti, Programátor v APPLIC, s.r.o., Liberec

### **Vzdělání**

2015–do současnosti, Doktorské studium, obor Technická kybernetika, Technická Univerzita v Liberci

2013–2015, Magisterské studium, obor Automatické řízení a inženýrská informatika, Technická Univerzita v Liberci

2010–2013, Bakalářské studium, obor Elektronické Informační a Řídicí Systémy, Technická Univerzita v Liberci

2006–2010, Maturitní vysvědčení z oboru Informatiky Střední škola aplikované kybernetiky, Hradec Králové

### **Osobní dovednosti**

Jazyky: čeština, angličtina

Programovací jazyky: C#, Lua, Python, Matlab, assembler, C, C++, LabView, HTML, Unity, VHDL, PLC, PHP, SQL