VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ZABEZPEČENÍ SENZORŮ-JAK ZABRÁNIT FALZIFIKACI

SECURING SENSORS - HOW TO PREVENT FALSIFICATION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **HYNEK BERNARD**
AUTHOR

VEDOUCÍ PRÁCE **PAVEL ZEMČÍK, prof. Dr. Ing.**
SUPERVISOR

BRNO 2020

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav počítačové grafiky a multimédií (UPGM) Akademický rok 2019/2020

Zadání bakalářské práce

Student: **Bernard Hynek**
Program: Informační technologie
Název: **Zabezpečení senzorů - jak zabránit falzifikaci**
Securing Sensors - How to Provent Falsification
Kategorie: Vestavěné systémy
Zadání:

1. Prostudujte literaturu a různá řešení zabezpečení senzorů/dat z nich a embedded systémů, zaměřte se na podepisování, časová razítka, zábranu proti kopírování systémů, ale i na fyzické vstupy senzorů. Zajímavé jsou například kamery, ale i jiné senzory jsou hodny zřetele (radar, zvukové senzory...)
2. Vyberte některý z aspektů zabezpečení senzorů, případně jejich soubor, a navrhněte způsob implementace zabezpečení (toto může být například rozpoznání záměny kamerového senzoru za záznam, zamezení "podstrčení" dat počítačovou sítí, ochrana proti záměně místa a času snímání apod.)
3. Navrhněte způsob implementace zabezpečení senzoru a diskutujte vlastnosti řešení.
4. Implementujte ochranu a funkčnost demonstrujte na vhodném příkladu.
5. Diskutujte dosažené výsledky a možnosti pokračování práce.

Literatura:

- Dle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3 zadání

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Zemčík Pavel, prof. Dr. Ing.**
Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2019
Datum odevzdání: 31. července 2020
Datum schválení: 1. listopadu 2019

Abstrakt

Tato práce se zabývá zabezpečením digitálních obrazových senzorů ve fotoaparátech a kamerách. Hlavním zaměřením práce je studie aktuálně používaných způsobů zabezpečení a identifikace kamerového senzoru podle pořízených obrazů, konkrétně extrakcí a porovnání přibližného vzorového šumu senzoru se zbytkovým šumem fotografie. Součástí práce je experimentování s reálnými senzory a vyhodnocování nejefektivnějšího postupu s vizí dalšího výzkumu použitých metod.

Abstract

This thesis is focused on securing digital imaging sensors in cameras and videocameras. Main direction of thesis is study of currently used security procedures and identification of camera sensor based on captured images, tangibly extraction and comparing aproxximated sensor pattern noise with noise residue of given photo. Experiments with real sensors and evaluation of most efficient approach with visions of next research of used methods are part of this thesis.

Klíčová slova

zabezpečení senzoru, PRNU, vzorový šum, obrazový filtr, identifikace senzoru

Keywords

sensor security, PRNU, pattern noise, image filter, sensor identification

Citace

BERNARD, Hynek. *Zabezpečení senzorů-jak zabránit falzifikaci*. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Pavel Zemčík, prof. Dr. Ing.

Zabezpečení senzorů-jak zabránit falzifikaci

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Prof. Dr. Ing. Pavla Zemčíka. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Hynek Bernard
31. července 2020

Poděkování

Rád bych poděkoval vedoucímu práce panu Prof. Dr. Ing. Pavlu Zemčíkovi za velikou ochotu a vstřícný přístup při konzultacích.

Obsah

Kapitola 1

Úvod

Zabezpečení je v dnešním světě aktuální téma. Při dnešní dostupnosti prostředků informačních technologií je možné velice detailně podvrhnout jakýkoliv záznam. Bez ochrany by bylo možné vytvářet falešné důkazy, či falzifikovat důkazy reálné. Pokud by se takové výtvory podařilo prosadit například u soudu, bylo by možné usvědčit nevinného člověka. Z toho vyplývá, že je zapotřebí vyvinout maximum při ověřování původu záznamů a jejich pravosti.

Kamerový systém je neocenitelný pomocník při monitorování veřejného prostranství a soukromých objektů. Stát využívá kamery například na mýtných branách a rychlostních radarech na silnici, pro zachycení důkazů při přestupcích, nebo v centrech měst, kde slouží k ochraně veřejného majetku. Když má město kamerový systém neustále pod dozorem, může v rychlosti vysílat policejní složky na místa, kde se schyluje k protiprávnímu jednání. Při pořizování a přenosu záznamu je použito několik prvků systému (senzor, procesor, komunikace...), z nichž je každý v základu zneužitelný.

V této práci analyzuji jednotlivé způsoby které již pro ochranu existují.

Téma práce jsem si vybral, jelikož mám již od útlého věku tendenci rozkládat složité systémy na nejmenší možné součásti pro jejich pochopení a hledat v nich slabiny které by se daly zneužít. Již jsem v minulosti několik takových slabin objevil a při jejich znalosti si lze ušetřit spoustu práce a získat výsledky, kterých by se při zamýšlené funkcionalitě nedostalo. Na druhou stranu je výhodné znát slabiny mnou provozovaných systémů pro zajištění jejich zabezpečení a robustnosti.

V práci se zaměřuji nejvíce na identifikaci senzoru podle získaného záznamu, poté ji implementuji a podle analýzy výsledků vybírám nejefektivnější postupy. Následující kapitola 2 se zabývá podrobnějším popisem senzorů a digitálních fotografií, v kapitole 3 jsou popsány klasicky používané metody zabezpečení. Další kapitola 4 se již zabývá problematikou identifikace senzoru na základě vzorového šumu a popisem této metody. Kapitola 5 je věnována analýze získaných dat a návrhu řešení. Kapitolou 6 popisuji implementaci vybraného způsobu identifikace. V kapitole 7 provádím experimenty s hlavní úlohou určení, jaký z postupů uvedených v kapitole 6 přináší největší efektivitu. V závěru diskutuji naměřené hodnoty a možnosti pokračování práce.

Kapitola 2

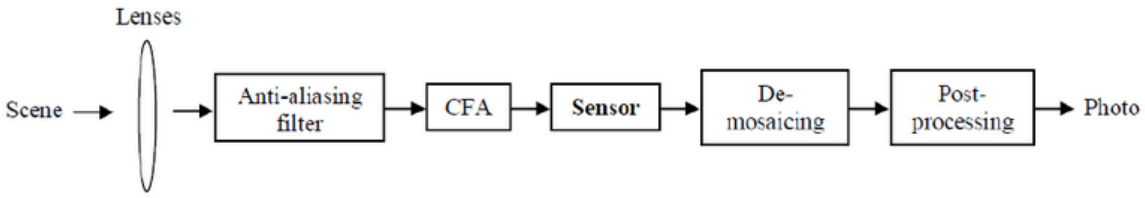
Způsob pořízení fotografie

V této kapitole jsou popsány způsoby pořizování dat kamerovými senzory od obrazu scény až po výslednou fotografii. V uvedeném rozsahu práce není prostor pro vyčerpávající encyklopedický výčet všech relevantních informací, proto jsou popsány pouze informace úzce související s hlavním tématem práce.

2.1 Pořízení a reprezentace obrazových dat

Pro pochopení zabezpečení, je nutné znát původ obrazových dat, jak se přenášejí a ukládají. Matematická reprezentace obrazových dat a teoretický přehled je dobře popsán například v [3].

Typický proces získání barevné fotografie digitálním fotoaparátem může být zjednodušeně ilustrován jako na obrázku 2.1.



Obrázek 2.1: Proces pořízení fotografie digitální kamerou [1]

Světlo ze scény prochází skrz čočku a většinou i anti-aliasingový filtr. Dále propadá skrz color filter array přímo na senzor který přijímané světlo převede na signál a odešle jej do procesoru, který signál dále demozaikuje a zpracuje.

2.2 Anti-Aliasingový filtr

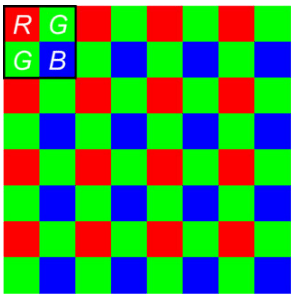
V nejmodernějších zařízeních se můžeme setkat s řešením bez fyzického filtru při použití velmi detailního senzoru, nebo za aplikace většího počtu senzorů, typicky pro každou barevnou složku. Optický Anti-Aliasingový filtr se chová jako dolní propusť, která roztřišťuje přicházející světlo a vytváří efekt rozmazání pro zamezení aliasingového efektu. Aliasingový efekt je jev který se projevuje při pokusu o zachycení frekvence jemnější, než jakou

[1] Obrázek byl převzat z [5]

podporuje rozmístění pixelů v senzoru[1]. Efekt je nejvíce zřejmý a nežádoucí při získávání jemných periodických vzorů na objektech při velkých optických rozlišeních[23]. Aliasingový efekt digitálního obrazu je nejčastěji pozorován v barevných systémech pokud mají pouze jeden obrazový senzor[15].

2.3 Color Filter Array a demozaikování

CFA je mozaikou barevných filtrů překrývající pixely senzoru. Barevné filtry omezují intenzitu světla dopadající na pixel v závislosti na barvě[1]. Nejčastěji používanou mozaiku barevných složek nazývanou bayerův filtr můžeme vidět na obrázku 2.2



Obrázek 2.2: Bayerův filtr [2]

Po průchodu senzorem vznikne prokládaný vzor signálových částí které reprezentují různé části barevného spektra. Z těchto dat rekonstruuje demozaikující algoritmus třídimenzionální plnobarevný signál[24].

Existuje více návrhů CFA a další se stále vytvářejí[22].

2.4 Obrazový senzor

V každém digitálním fotoaparátu nebo kameře nalezneme obrazový senzor, který pořizuje záznam. V dnešních kamerách lze narazit na 2 typy senzorů - CCD a CMOS. Oba typy jsou si velice podobné, mají společnou technologii a princip: jsou složeny z MOSFET tranzistorů a PN přechodů a převádějí světelnou energii(fotony) na elektrickou(elektrony) za užití fotoelektrického jevu. Nejmenší adresovatelná jednotka takového senzoru se nazývá pixel. Pixely jsou většinou čtvercové, s velikostí v řádech mikronů. Počet elektronů generovaných světlem v pixelu závisí na rozměrech pixelu a homogenitě výrobního materiálu.[9]. Když je expozice hotová, CCD přenáší každý náboj pixelu sekvenčně na společný výstup a dále konvertuje náboje na napětí. V CMOS senzorech se provádí konverze přímo v každém pixelu souběžně.[8]

2.5 Post-Processing

V této části lze již uložit signál jako obraz ve formátu RAW, ale takový obraz ještě potřebuje demozaikovat a provést barevné a gamma korekce. Některé kamery implementují i filtrování, jako například odšumění a doostření. Na konci tohoto řetězce je fotografie uložena v JPEG nebo jiném formátu který může zahrnovat kvantování[9].

[2]Obrázek byl převzat z [27]

Kapitola 3

Existující typy zabezpečení

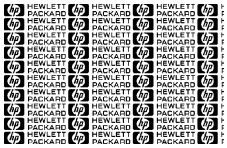
V této kapitole se zaměřuji na existující způsoby zabezpečení a alternativy k hlavnímu tématu práce. Pro útočníka by bylo velmi náročné podvrhnout data vycházející přímo ze senzoru, jelikož by potřeboval fyzický přístup k zařízení a senzor je připájený na desku s plošnými spoji, což znamená, že je neoddělitelnou součástí celé kamery. Informace nejsou popsány vyčerpávajícím způsobem, byly vybrány pouze informace úzce souvisejícím s hlavním tématem práce. Dnešní metody zabezpečení se tedy nezaměřují na senzor jako takový, ale spíše na prokázání původu snímku a toho, že nebyl snímek alterovaný. Většina těchto metod se dá mezi sebou kombinovat.

3.1 Identifikace podle souboru

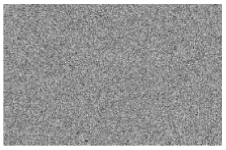
Původ fotografie se dá snadno identifikovat například pomocí EXIF hlavičky v souboru. V takové hlavičce nalezneme mimo jiné i informace o datu a času pořízení, času expozice, výrobci a modelu fotoaparátu/kamery[2]. Další informace lze vytáhnout i z kvantizačního JPEG headeru (některé kamery používají vlastní kvantizační matice)[17]. Všechny tyto informace jsou při identifikaci velmi užitečné, ale pro účely zabezpečení absolutně nedostačující. Hodnoty jsou kýmkoliv libovolně upravitelné, proto nelze absolutně určit jejich pravost.

3.2 Vodoznak

Obrázek 3.1: Obraz s vodoznakem

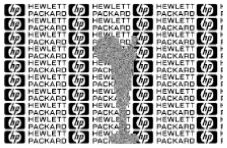


Obrázek 3.2: Extrahovaný vodoznak



Obrázek 3.3: Pokus o extrakci bez klíče

Obrázek 3.4: Upravený obraz s vodoznakem



Obrázek 3.5: Porušený vodoznak

[1]

Vodoznak lze chápat jako přidanou vrchní vrstvu na fotografii. Je více druhů vodoznaků, které můžeme zakomponovat do fotografie. Rozdělují se na křehké a robustní. Robustní vodoznaky jsou většinou viditelné, často se používají jako podpis fotografa, který zveřejňuje

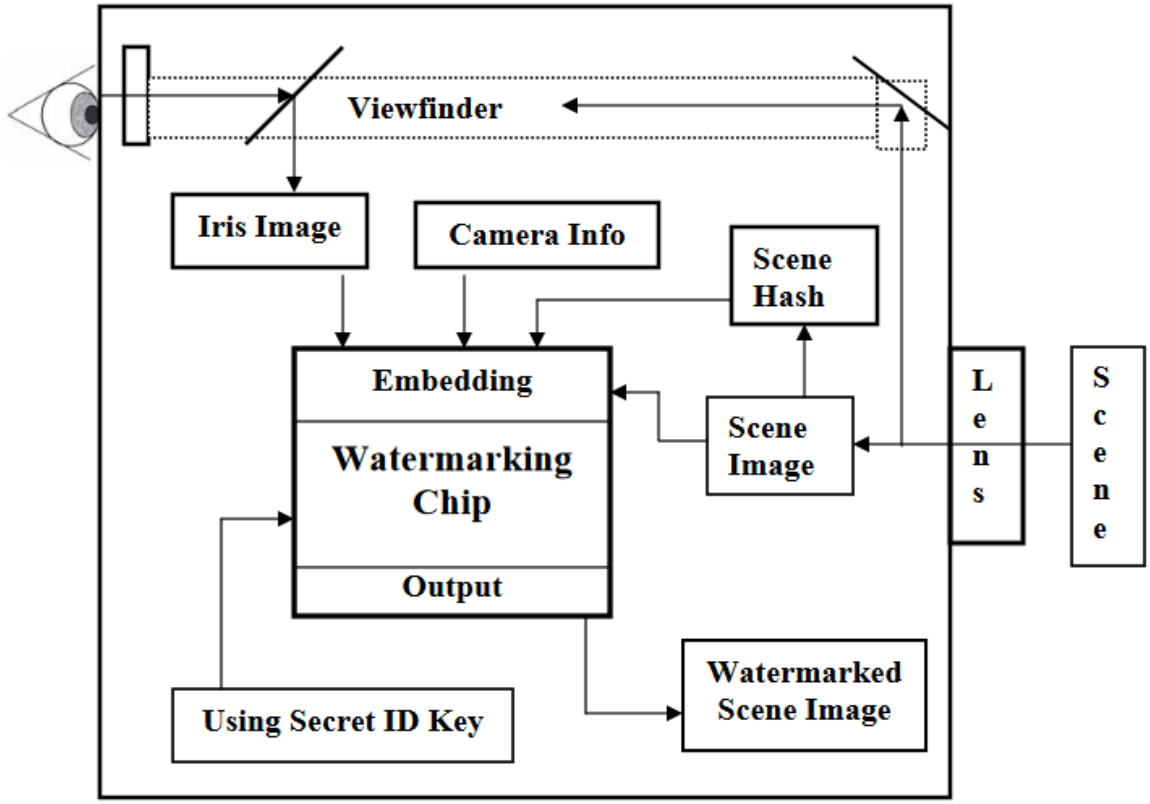
[1] Obrázky 3.1,3.2,3.3,3.4,3.5 byly převzaty z [28]

své fotografie. Takový vodoznak zamezí falešnému vydávání obrazu jako cizího vlastnictví. Pro nejefektivnější funkcionalitu by měl být stále viditelný i po ořezání a kompresi[28]. Většina dnešních profesionálních fotoaparátů má možnost automatického přiložení vodoznaku do každé fotografie.

Jako křehký vodoznak je označován takový, který zajišťuje snímkovou integritu, při úpravě snímku se poruší a vodoznak není nadále čitelný, to znamená že byl obraz upraven. Většinou není lidským okem viditelný. Integrita i vlastnictví snímku může být například ověřena uživatelem s příslušným privátním klíčem[28]. Na obrázku 3.3 lze efektivně vidět jak vypadá neoprávněný pokus o získání vodoznaku. Na obrázku 3.5 je extrahovaný vodoznak na snímku s porušenou integritou, lze vidět zřejmou oblast která byla upravena pro získání falzifikovaného snímku. Například Epson PhotoPC 700 má možnost nahrání softwaru pro podepisování obrazu neviditelným křehkým vodoznakem.

3.3 Biometrická stopa fotografa

Biometrickou stopou se rozumí fyzický nezaměnitelný identifikátor fotografa, například obraz jeho duhovky - žádné 2 duhovky nejsou totožné. Fotoaparát automaticky snímá obraz lidské duhovky skrz hledáček při každém pořízení fotografie. Tento obraz je poté komprimovaný a kombinovaný se skrytým identifikačním klíčem fotoaparátu, hashem originálně pořizované scény a přidanými informacemi o vlastnostech kamery. Výsledkem je bioforenzní ověřovací podpis který je vestavěn do výsledné fotografie jako vodoznak[4]. Na obrázku 3.6 můžeme vidět diagram součástí takto zabezpečené kamery.



Obrázek 3.6: Biometrická kamera [2]

Duhovka není jediná biometrika aplikovatelná tímto způsobem, lze například použít i otisky prstů.

3.4 Analýza pixelových defektů

V [10] byl navrhnut způsob identifikace senzoru podle vadných pixelů. Úspěšnost určení senzoru se pohybovala mezi 75-95%. Tato metoda se však pouze zabývá CCD senzory a je založena na předpokladu, že při výrobě, nebo stářím senzoru, nastal některý z následujících defektů[10]:

- bodový defekt: pokud je CCD nasvícen na 70% saturace, pixel má odchylku více jak 6 procent,
- vypálený bodový defekt: pixely s velmi vysokým výstupním napětím,
- mrtvé pixely: pixely se špatnou responzivitou,
- pixelové pasti: problém s přenosem náboje, který má za výsledek částečný nebo kompletní špatný sloupec,
- shluky defektů: oblasti složené z bodových defektů.

[2]Obrázek byl převzat z [4]

Kapitola 4

Identifikace senzoru na základě šumu senzoru

V této kapitole popisuji relevantní informace k hlavnímu tématu práce, výčet není encyklopedický a jsou vybrány pouze nejdůležitější informace. Popsanými metodami jsou získávány všechny výsledky v experimentech.

4.1 Vzorový šum senzoru

Je hodně zdrojů nedokonalostí a šumů, které se projevují v jednotlivých fázích pořizování fotografie. I když senzor pořídí obrázek absolutně rovnoměrně osvětlené scény, výsledný digitální obraz stále projeví malé změny v intenzitě mezi jednotlivými pixely. Toto je částečně způsobeno jevem, jenž je nazýván jako shot noise[1], který je náhodnou složkou, a částečně vzorovým šumem - deterministický komponent který zůstává přibližně stejný i pokud se pořídí více snímků úplně stejné scény. Díky této vlastnosti je vzorový šum přítomný v každém obrazu, který senzor pořídí a tím pádem může být použitý pro identifikaci kamery[17]. Předpokládá se, že průměrování více snímků redukuje náhodné veličiny a vytahuje vzorový šum.

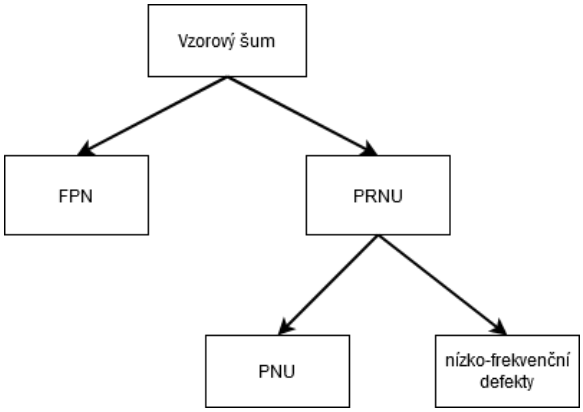
Dvě hlavní složky vzorového šumu, nazývaného jako sensor pattern noise(SPN), jsou fixed pattern noise (FPN) a photo-response nonuniformity noise (PRNU) (viz obrázek 4.1). FPN je způsobený "temným proudem"[17][2]. FPN primárně odpovídá rozdílům mezi pixely, když není senzor vystaven světlu[17]. Protože je aditivním konstantním šumem, některé kamery a fotoaparáty tento šum automaticky potlačují odpočtem dark-framu z každého obrazu, který vyfotí[7].

V přirozených fotografiích je dominantní částí vzorového šumu PRNU. Ta je způsobena primárně pixelovou nejednotností (PNU - pixel nonuniformity), která je definována jako různá citlivost pixelů na světlo způsobená nejednolitostí silikonových waferů a nepřesností v průběhu výroby senzoru. Vlastnosti a původ PNU šumu nasvědčují tomu, že i senzory vyrobené ze stejného waferu, by naznačovaly korelované PNU vzory. PNU šum na rozdíl od FPN není ovlivněn okolní teplotou a vlhkostí[17]. PNU obsahuje také všechny systematické defekty senzoru, včetně mrtvých a vypálených pixelů.

Vzorový šum má velmi užitečné vlastnosti vhodné pro forenzní účely, dá se chápat jako otisk senzoru s následujícími vlastnostmi[9]:

[1] shot noise = fluktuace počtu detekovaných fotonů, způsobená na jejich projevu nezávisle na sobě

[2] dark current = nízký proud proudící světlocitlivými součástkami i když na ně nedopadají žádné fotony



Obrázek 4.1: Rozdělení vzorového šumu senzoru

- Dimenzionalita - Otisk je přirozeně stochastický a má hodně informací, které zapříčiňují originalitu u každého senzoru
- Univerzálnost - Všechny senzory projevují PRNU
- Generalita - zkoumaný šum je přítomný na každém snímku nezávisle na čočkách, nastavení fotoaparátu nebo scéně, vyjímkou mohou být absolutně tmavé scény
- Stabilita - šum je stabilní v čase i při změně okolních podmínek (teplota, vlhkost)
- Robustnost - přežije ztrátovou kompresi a běžné zpracování[18]

Otisk může být použit nejen pro identifikaci senzoru, ale i pro další forenzní úlohy[9]:

- Testováním přítomnosti určitého otisku v obrázku lze identifikovat zařízení nebo prokázat skutečnost, že byly 2 fotografie pořízeny stejným zařízením. Přítomnost otisku v obrazu je také indikátor toho, že zkoumaný obraz je vyfocený a nejedná se o počítačové vykreslení.
- Zjištěním absence otisku v jednotlivých obrazových částech je možné odhalit zaměněné nebo upravené části obrazu - tato úloha podporuje ověření integrity zkoumaného obrazu.
- Zjištěním síly a formy otisku je možné rekonstruovat části historie zpracování obrazu, například oříznutí, rotaci nebo zvětšení.

4.2 Detekce na základě vzorového šumu

Metoda A

Pro detekci je důležité mít otisk senzoru $\mathbf{K}$, jako jeden způsob se nabízí průměrování několika snímků $\mathbf{I}^{(i)}, i = 1, ..., N$ ze senzoru, ovšem podle [17] je mnohem efektivnější průměrování zbytkových šumů $\mathbf{W}^{(i)}$ z obrazů $p^{(i)}$ s potlačením scény filtrem F.

$$\mathbf{W}^{(i)} = \mathbf{I}^{(i)} - F(\mathbf{I}^{(i)}) \tag{4.1}$$

$$\mathbf{K} = \sum_{i=1}^{N_p} \mathbf{W}^{(i)} \tag{4.2}$$

Čím více je při výpočtu použito obrázků N, tím více se potlačí náhodné šumy a dopad scény na výsledek, je doporučováno použít $N > 50$[17].

Pro zjištění skutečnosti, zda byl obraz $\mathbf{I}$ pořízen kamerou C, počítá se korelace p_C mezi zbytkovým šumem $\mathbf{n} = \mathbf{p} - F(\mathbf{p})$ a otiskem kamery $\mathbf{P}_C$

$$p_C(\mathbf{p}) = corr(\mathbf{n}, \mathbf{P}_C) = \frac{(\mathbf{n} - \overline{\mathbf{n}}) * (\mathbf{P_C} - \overline{\mathbf{P}}_\mathbf{C})}{||\mathbf{n} - \overline{\mathbf{n}}||||\mathbf{P_C} - \overline{\mathbf{P}}_\mathbf{C}||} \tag{4.3}$$

Čím větší hodnota p_C, tím více se šum zkoumaného obrázku podobá aproximovanému otisku, ovšem pro přesnost je nutné zvolit statisticky práh, který bude definitivně určovat zda je snímek pořízený touto kamerou.

Metoda B

Další, složitější způsob pro aproximaci, použitý v [11], pracuje také s vyfiltrovaným šumem $\mathbf{W}^{(i)}$, ale používá jiný přístup k průměrování:

$$\mathbf{K} = \frac{\sum_{i=1}^{N} \mathbf{W}^{(i)} \mathbf{I}^{(i)}}{\sum_{i=1}^{N} (\mathbf{I}^{(i)})^2} \tag{4.4}$$

Jako aproximaci testu generalizovaného pravděpodobnostního poměru udává maximum normalizované korelace p[11]. Metoda předpokládá maximálně ořezání jako jedinou geometrickou úpravu.

$$\max_{s_1, s_2} p(s_1, s_2; \mathbf{X}, \mathbf{Y}), \tag{4.5}$$

kde

$$p(s_1, s_2; \mathbf{X}, \mathbf{Y}) = \frac{\sum_{k=1}^{m} \sum_{l=1}^{n} (\mathbf{X}[k, l] - \overline{\mathbf{X}})(\mathbf{Y}[k + s_1, l + s_2] - \overline{\mathbf{Y}})}{||\mathbf{X} - \overline{\mathbf{X}}||||\mathbf{Y} - \overline{\mathbf{Y}}||}, , \tag{4.6}$$

$||.||$ je norma L_2, a

$$\mathbf{X} = \mathbf{IK}, \mathbf{Y} = \mathbf{W}, \tag{4.7}$$

Maximum v 4.5 je vybráno ze všech k přípustných posunů mezi možně ořezanými obrázky a otiskem kamery. Počet přípustných posunů je $k = (m_K - m + 1)(n_K - n + 1)$

Před vyhodnocením 4.6, je obraz odsazen nulami aby odpovídal velikostem $\mathbf{X}$ a $\mathbf{Y}$. Na posuny $k + s_1$ a $l + s_2$ je aplikováno modulo m a n.

Při označení souřadnic vrcholu kde se projevuje maximum 4.5 jako $s_{peak} = [s_1, s_2]$. Lze zapsat Peak to Correlation Energy ratio (PCE) používané jako míra výšky vrcholu takto:

$$PCE_k = \frac{p(\mathbf{s_{peak}}; \mathbf{X}, \mathbf{Y})^2}{\frac{1}{mn-|N|}\sum_{s,s\notin N} p(\mathbf{s}; \mathbf{X}, \mathbf{Y})^2} = \frac{(\mathbf{X} * \mathbf{Y}(\mathbf{s_{peak}}))^2}{\frac{1}{mn-|N|}\sum_{s,s\notin N}(\mathbf{X} * \mathbf{Y}(\mathbf{s}))^2}, \tag{4.8}$$

Kde $\mathbf{X} * \mathbf{Y}(\mathbf{s})$ je skalární součin mezi $\mathbf{X} - \overline{\mathbf{X}}$ a $\mathbf{Y}(\mathbf{s}) - \overline{\mathbf{Y}}$ kruhově posunutý vektorem s_2 a N je lokální okolí vrcholu. Pokud nebyl obraz oříznutý, nehledá se vrchol a $k = 1$ v 4.8[11].

4.3 Externí vlivy na vzorový šum

Vliv zoomu při pořízení fotografie

V [21] provedl autor zkoumání vlivu přiblížení při pořízení fotografie na finální aproximovaný vzorový šum. Experimentální výsledky prokázaly, že použití přiblížení má značný vliv na výslednou aproximaci. Autoři došli k závěru že velká část vlivu je způsobená post-processingem po digitálním přiblížení[21].

Vliv velikosti a pozice zkoumané oblasti

Může se zdát, že čím větší je zkoumaná plocha snímku, tím přesnější otisk snímku lze získat, protože na větší ploše je více informací. Toto tvrzení není vždy pravdivé. V [5] byly provedeny testy na různých velikostech výřezů z různých částí obrazu. Práce je založena na projevu anomálie v tabulce 4.1 při testování různých velikostí.

Rozlišení	128 x128	128 x256	256 x512	256 x512	512 x512	512 x1024	1024 x1024	1024 x2048	1536 x2048
FPR %	41,68	38,68	32,60	25,71	16,28	6,75	1,9	2,4	12,03

Tabulka 4.1: False positive rate[3](%) za použití bloků různých velikostí

Experimenty, které byly provedeny v [5], zkoumající různé oblasti v obraze, jsou shrnuty v tabulce 4.2. Velikost každého bloku je 512x512 pixelů a odsazení se počítá od nejbližšího okraje a vrchu obrazu.

Pozice	**vpravo-nahoře**	**vpravo-nahoře**	**vpravo-nahoře**	**uprostřed**	**vlevo-nahoře**	**vlevo-nahoře**	**vlevo-nahoře**
Odsazení	**0px**	**16px**	**48px**	**-**	**0px**	**16px**	**48px**
FPR (%)	25,48	9,33	10,95	7,19	64,67	41,43	22,76

Tabulka 4.2: Vliv různých oblastí snímků na kvalitu rozpoznání původní kamery

Neintuitivní situace a výsledky jsou dávány za vinu vinětaci[4] fotoaparátů a kamer. Finální doporučení pro zkoumané oblasti jsou[5]:

[3]FPR - výsledek ukazuje že obraz je pořízen senzorem i když ve skutečnosti není -> čím menší číslo, tím efektivnější metoda

[4]Vinětace = vada optických soustav, projevující se nižším jasem na okrajích zobrazovaného obrazu

- Vynechání periferních pixelů z obrazů před extrakcí otisku
- Výběr bloků z prostřední části obrazu, pokud jsou při aplikaci zapotřebí jen menší bloky
- Obezřetnost při používání bloků zahrnujících okraje obrazu, pokud se používá ověření na základě bloků

Vliv komprese

Pro vliv komprese JPEG byly proveden experiment v [17]. I když ztrátová komprese JPEG snižuje průměrnou hodnotu korelací mezi zbytkovým šumem a správným vzorovým šumem, děje se tak postupně se snižujícím se parametrem kvality. Bylo pozorováno, že parametr kvality pro JPEG kompresi 90 a méně potlačuje nechtěné minimální pozitivní korelace mezi vypočítanými otisky [17].

Vliv použitého filtru na vzorový šum

Ideální filtr používaný pro identifikaci senzoru by měl pouze extrahovat šum senzoru bez projevů scény do výsledku. V [17][11][6] se používá filtr založený na vlnkové doméně, ale později se objevily výzkumy zkoumající vliv jiných filtrů[25] na výsledky a rychlost výpočtu[29]. Žádný filtr ovšem není ideální a všechny mají své klady a zápory.

4.4 Wienerův filtr

Wienerův filtr se dá při filtrování dvou-dimenzionálních obrazů použít jak v frekvenční, tak v prostorové doméně.

Pro prostorovou doménu by se dal zapsat jako

$$y(i,j) = \sum_{m=-N}^{N} \sum_{n=-N}^{N} w(m,n)x(i+m, j+n) \tag{4.9}$$

ke $y(i,j)$ je výstup filtru, $x(i,j)$ zašuměný obraz a $d(i,j)$ originální scéna bez šumu. Váhy $w(m,n)$ mohou být nalezeny minimalizací.

$$j = E(\{d(i,j) - y(i,j)\}^2) \tag{4.10}$$

E značí předpoklad. Výsledek pro $w(m,n)$ je získán ve vektorovém tvaru jako

$$w = R^{-1}p \tag{4.11}$$

kde

$$\begin{aligned} w &= [w(-N,-N)...w(-N,N), w(-N+1,-N)...w(-N+1,N)...w(0,0)...w(N,N)]^T \\ p &= [p(-N,-N)...p(-N,N), p(-N+1,-N)...p(-N+1,N)...p(0,0)...p(N,N)]^T \end{aligned} \tag{4.12}$$

a

$$R = \begin{pmatrix} R(0,0) & ... & R(0,2N) & ... & R(1,0) & ... & R(2N,2N) \\ R(0,-2N) & ... & R(0,0) & ... & R(1,-2N) & ... & R(2N,2N) \\ R(-1,0) & ... & R(-1,2N) & ... & R(0,0) & ... & R(2N-1,2N) \\ R(-2N,-2N) & ... & R(-2N,0) & ... & R(-2N+1,-2N) & ... & R(0,0) \end{pmatrix} \tag{4.13}$$

Tyto rovnice jsou známé jako Wiener-Hopfovy rovnice. Matice R objevující se v rovnici je symetrická Toeplitzova matice.

$R(m,n)$ a $p(m,n)$ odpovídají autokorelační funkci $x(i,j)$ a křížově-korelační funkci $d(i,j)$ a $x(i,j)$, které jsou zadány[20]:

$$\begin{aligned} R(m,n) &= E[x(i,j)x(i-m,j-n)] \\ p(m,n) &= E[d(i,j)x(i-m,j-n)] \end{aligned} \tag{4.14}$$

V MATLABu je filtr implementován jako funkce `wiener2` následovně:

Prvně `wiener2` *odhadne lokální průměr a rozptyl okolo každého pixelu*

$$\mu = \frac{1}{NM} \sum_{n_1,n_2 \in \eta} a(n_1,n_2) \tag{4.15}$$

a

$$\sigma^2 = \frac{1}{NM} \sum_{n_1,n_2 \in \eta} a^2(n_1,n_2) - \mu^2 \tag{4.16}$$

kde η je lokální okolí v obrázku A o rozměrech N krát M. Funkce potom vytvoří Wienerův filtr za použití těchto odhadů,

$$b(n_1,n_2) = \mu + \frac{\sigma^2 - v^2}{\sigma^2}(a(n_1,n_2) - \mu), \tag{4.17}$$

kde v^2 je šumový rozptyl. Pokud není zadán šumový rozptyl, `wiener2` *použije průměr všech lokálních odhadnutých rozptylů*[19].(překlad autora)

4.5 Vlnkový filtr

Filtr používaný v [17][11][6] by se dal nazvat jako vlnkový. Počítá s maximální velikostí bloku 512 x 512 (obraz se může rozdělit do bloků) a barevné složky jsou filtrovány odděleně. *Vysokofrekvenční vlnové koeficienty zašuměného obrazu jsou modelovány jako přidaná složka lokálního stacionárního nezávislého a rovnoměrně rozloženého signálu s průměrem 0 a stacionární bílý Gaussovský šum $N(0,\sigma_0^2)$(šumová složka). Odšumovací filtr je postaven ve 2 částech. V první části se odhaduje lokální obrazový rozptyl a v druhé části se používá lokální Wienerův filtr pro odhad odšumeného obrazu ve vlnkové doméně. Jednotlivé kroky vypadají takto:*

1. *Vypočet čtvrté úrovně vlnkové dekompozice zašuměného obrazu s 8-krokovými Daubechiesovými kvadratickými zrcadlovými filtry. Popisujeme postup pro jednu fixní úroveň(je provedena ve vysokofrekvenčních pásmech pro všechny čtyři úrovně). Označíme vertikální, horizontální a diagonální subpásma jako $\mathbf{h}[i,j], \mathbf{v}[i,j], \mathbf{d}[i,j]$ kde (i,j) probíhá indexním souborem $\mathcal{J}$ který závisí na úrovni dekompozice.*

2. *Pro každé subpásmo, odhadnout lokální rozptyl originálního obrazu bez šumu pro každý vlnkový koeficient za použití MAP estimace pro 4 velikosti čtverce $W \times W$ okolí $\mathcal{N}$, pro $W \in 3,5,7,9$.*

$$\hat{\sigma}_W^2[i,j] = \max(0, \frac{1}{W^2} \sum_{(i,j)\in\mathcal{N}} \mathbf{h}^2[i,j] - \sigma_0^2), (i,j) \in \mathcal{J}. \tag{4.18}$$

Jako finální odhad 4 rozptylů se bere minimum,

$$\hat{\sigma}^2(i,j) = \min(\sigma_3^2[i,j], \sigma_5^2[i,j], \sigma_7^2[i,j], \sigma_9^2[i,j]), (i,j) \in \mathcal{J}. \tag{4.19}$$

3. *Odšuměné vlnkové koeficienty jsou získány Wienerovým filtrem*

$$\mathbf{h}_{den}[i,j] = \mathbf{h}[i,j]\frac{\hat{\sigma}^2[i,j]}{\hat{\sigma}^2[i,j]+\sigma_0^2} \tag{4.20}$$

a nápodobně pro $\mathbf{v}[i,j]$ *a* $\mathbf{d}[i,j], (i,j) \in \mathcal{J}$.

4. *Opakovat kroky 1-3 pro každou úroveň a každý barevný kanál. Odšuměný obraz je získán použitím inverzní vlnkové transformace na odšuměné vlnkové koeficienty.*

[17](překlad autora)

4.6 WNNM filtr

Weighted nuclear norm minimatization je filtr založený na nelokální sebepodobnosti[5], to znamená, že předpokládá, že je v obrazu více opakovaných lokálních vzorů napříč celým přirozeným obrazem a tyto nelokální "záplaty"mohou pomoct při rekonstrukci jiných záplat[12]. Filtr je udávaný jako jeden z nejefektivnějších pro extrakci vzorového šumu, ale je výpočetně velmi náročný[25] .

Celý odšumovací algoritmus je shrnutý jako Algoritmus 1

Algoritmus 1: Odšumovací algoritmus WNNM pro obrazy

Input: Zašuměný obraz $\mathbf{y}$
Output: Čistý obraz $\hat{\mathbf{x}}^{(K)}$

Inicializuj $\hat{\mathbf{x}}^{(0)} = \mathbf{y}, \mathbf{y}^{(0)} = \mathbf{y}$
for *k=1:K* **do**
  Iterativní regularizace $\mathbf{y}^{(k)} = \mathbf{x}^{(k-1)} + \delta(\mathbf{y} - \hat{\mathbf{y}}^{(k-1)})$
  for *Každá záplata* $\mathbf{y}_j$ *v* $\mathbf{y}^{(k)}$ **do**
    Najdi podobnou záplatovou skupinu $\mathbf{Y}_j$
    Odhadni váhový vektor $\mathbf{w}$
    Singulární rozklad $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = SVD(\mathbf{Y}_j)$
    Odhadni: $\hat{\mathbf{X}}_\mathbf{j} = \mathbf{U}\mathbf{S}_\mathbf{w}(\mathbf{\Sigma})\mathbf{V}^\mathbf{T}$
  end for
  Shrň $\mathbf{X}_j$ pro složení čistého obrazu $\hat{\mathbf{x}}^{(K)}$
end for

4.7 CAGI filtr

Content adaptive guided image filter je vylepšením stávajícího Guided image filteru(GIF), který je jedním z nejúspěšnějších lokálních filtrů. GIF je v porovnání s ostatními jednodušší a rychlejší[29]. Pro pochopení CAGIF je nejprve nutné poznat jeho předchůdce.

Výstupní obraz $\hat{\mathbf{Z}}$ je lokálně vypočítáván jako lineární transformace naváděcího obrazu $\mathbf{I}$ kterým může být i vstupní obraz X nebo jiný obraz. GIF je zapsán jako

$$\hat{\mathbf{Z}}(\mathbf{p}) = \overline{a}_p I(p) + \overline{b}_p,^{6} \tag{4.21}$$

[5]NSS = nonlocal self-similarity
[6]Čára nad symbolem znamená průměr

kde p je lokální okolí s předem zvoleným radiusem r a

$$a_p = \frac{\frac{1}{|\Omega_r(p)|}\sum_{p' \in \Omega_{p'}(p)} I(p')X(p') - \mu_{I,r}(p)\mu_{X,r}(p)}{\sigma^2_{I,r}(p) + \epsilon}\,[7] \tag{4.22}$$

kde ϵ je předem zadaná konstanta

$$b_p = \mu_{X,r}(p) - a_p\mu_{I,r}(p) \tag{4.23}$$

Filtr je v MATLABu implementován podle následujícího algoritmu[8] 2

Algoritmus 2: Guided image filter

1: $\text{mean}_I = f_{\text{mean}}(I, r)$
$\text{mean}_p = f_{\text{mean}}(p, r)$
$\text{corr}_I = f_{\text{mean}}(I. * I, r)$
$\text{corr}_{Ip} = f_{\text{mean}}(I. * p, r)$

2: $\text{var}_I = \text{corr}_I - \text{mean}_I .* \text{mean}_I$
$\text{cov}_{Ip} = \text{corr}_{Ip} - \text{mean}_I .* \text{mean}_p$

3: $a = \text{cov}_{Ip}./(\text{var}_I + \epsilon)$
$b = \text{mean}_p - a .* \text{mean}_I$

4: $\text{mean}_a = f_{\text{mean}}(a, r)$
$\text{mean}_b = f_{\text{mean}}(b, r)$

5: $q = \text{mean}_a . * I + \text{mean}_b$

kde $f_{\text{mean}}(\cdot, r)$ značí průměrující filtr s radiusem r.
V CAGIF je postup převážně stejný, ale jako rozšíření GIF je aplikováno váhování s ohledem na okraje (EAW). Udáme-li dva pixely p a p' v naváděcím obrazu. Je-li $I(p')$ na kraji a $I(p)$ je v ploché oblasti, hodnota $\sigma^2_{I,r}(p)/\mu^2_{I,r}(p)$ je většinou menší než $\sigma^2_{I,r}(p')/\mu^2_{I,r}(p')$. Na základě tohoto pozorování je váhování s ohledem na okraje počítáno za použití normalizovaného lokálního rozptylu všech pixelů [16].

Výpočet pro váhování je definován následovně:

$$\Gamma_I(p) = \frac{1}{N}\sum_{p'=1}^{N} \frac{\frac{(\sigma^2_{I,r}(p)+v_1)^\zeta}{(\mu^2_{I,r}(p)+v_2)^\zeta}}{\frac{(\sigma^2_{I,r}(p')+v_1)^\zeta}{(\mu^2_{I,r}(p')+v_2)^\zeta}} \tag{4.24}$$

kde v_1, v_2, ζ jsou tři konstanty. Hodnota v_1 je $(0,001 * L)^2$ kde L je dynamický rozsah vstupního obrazu. Hodnota v_2 je 10^{-9}. Hodnota ζ se v [16] volí jako 0,75.

[7] σ^2 -> rozptyl , μ -> průměr
[8] Algoritmus je převzat z [13]

Vypočítaná hodnota Γ_I se v 4.22 dosadí za ϵ ve tvaru $\frac{\lambda}{\Gamma_I(p)}$, takže vzorec pro výpočet hodnoty a_p v CAGIF se změní na

$$a_p = \frac{\frac{1}{|\Omega_r(p)|}\sum_{p' \in \Omega_{p'}(p)} I(p')X(p') - \mu_{I,r}(p)\mu_{X,r}(p)}{\sigma^2_{I,r}(p) + \frac{\lambda}{\Gamma_I(p)}} \tag{4.25}$$

Aby filtry zachovávaly co nejvíce informací o scéně a extrahovaly pouze šum, je zapotřebí nastavit parametr r flexibilně podle síly texturovanosti vstupního obrazu. Ve zkratce →čím více je obraz texturovaný, tím menší okolí by se mělo aplikovat. V [29] se používá pro určení parametru r následující postup:

Změření texturovanosti daného obrazu se provádí metrikou zvanou normalized total variation (TV)[9] která je definována jako Dll_1 norma prvních derivací v každém bodě dělená velikostí obrazu.

Podle dané hodnoty TV se poté určí velikost radiusu r[10][29]:

$$r = \begin{cases} 6 - \text{round}(\text{TV}/8) & \text{pro TV} < 36 \\ 2 & \text{jindy} \end{cases} \tag{4.26}$$

4.8 Anizotropní difuze

Anizotropní difuze je nejsložitější způsob filtrování implementovaný v této práci. Je založena na izotropní difuzi, ale ta ve výsledku rozmazává hrany a textury, proto by se zbytky scény promítaly do získaného vzorového šumu[14]. Celý postup filtrace obrazu anizotropní difuzí je v [14] definován následovně:

Obraz se může skládat z různých homogenních a nehomogenních oblastí. Mohou zde být kontinuální (hladké) plochy přibližně stejné intenzity a nekontinuální plochy (textury a hrany) s rozdílnými intenzitami. Na této struktuře se zakládá algoritmus anizotropní difuze, kde se nejprve považuje kontinuita rovnice, ve které je explicitně předpokládáno, že intenzita $I(x, y, t)$ je zachovávaná hodnota:

$$\frac{\partial I(x, y, t)}{\partial t} = - \bigtriangledown \cdot J(x, y, t) \tag{4.27}$$

kde $J(x, y, t)$ značí nestabilitu obrazové intenzity a celá pravá strana rovnice značí rozdílnost dané nestability. Proměnná t v $I(x, y, t)$ značí víceúrovňový přístup; každé t značí jinou úroveň. Takže tato rovnice 4.27 říká, že obrazová intenzita je jednoduše přerozdělená v obrazu a tempo kterým se to děje se rovná negativní rozdílnosti nestability. Jinými slovy, je zde přerozdělení hodnot pixelů v malé blízkosti mezi sebou.

Nestabilita může být popsána jako:

$$J = -c(x, y, t) \bigtriangledown I(x, y, t) \tag{4.28}$$

Tato rovnice značí, že vysoké hodnoty intenzity obrazu "tečou"do hodnot nižší intenzity v závislosti na spádu obrazu a difuzním koeficientu c. Kombinací těchto dvou rovnic se složí rovnice anizotropní difuze:

$$\frac{\partial I(x, y, t)}{\partial t} = \bigtriangledown \cdot (c \bigtriangledown I) = c \bigtriangledown^2 I + \bigtriangledown c \cdot \bigtriangledown I = \frac{\partial}{\partial x}(c\frac{\partial I}{\partial x}) + \frac{\partial}{\partial y}(c\frac{\partial I}{\partial y}) \tag{4.29}$$

[9] Neexistuje jsem definitivní český překlad, ale doslovně by se výraz dal přeložit jako normalizovaný celkový rozptyl

[10] round() je zaokrouhlení

V závislosti na (lokálním) difuzním koeficientu c je obraz odšuměn. Pokud $c(x, y, t) = 1 \forall (x, y)$*, pak můžeme vidět že*

$$I(x, y, t) = I_0(x, y, t) * G(x, y, t), \tag{4.30}$$

$$G(x, y, t) = \frac{1}{2\pi t} e^{-(x^2+y^2)/2t} \tag{4.31}$$

je validní řešení; toto značí izotropní difuzi. Nicméně izotropní difuze ve výsledku rozmazává hrany a textury, čemuž se chceme vyhnout, protože to povede k zbytkovému obrazu v PRNU vzoru. Tím pádem optimizace odšumění schází dolů na nalezení vhodných difuzních koeficientů. Autoři navrhují použít obrazový spád jako parametr pro ovládání difuze. Toto vyvolává anizotropní difuzi (jiný difuzní parametr v každém směru)

Perona a Malik zvolili použití čtyř nejbližších sousedů a poté toto bylo rozšířeno na osm nejbližších sousedů.

Použitím diferenčních kvocientů jako aproximace derivátů 4.29, nalezneme že

$$\frac{\partial I(x, y, t)}{\partial t} = c_N \triangle I_N - c_S \triangle I_S + c_E \triangle I_E - c_W \triangle I_W \tag{4.32}$$

kde $\triangle$ *značí nejbližší rozdíly sousedů. Například* $\triangle I_N$ *značí rozdíl mezi aktuálním pixelem a pixelem nad ním ("Sever") a* C_N *značí difuzní parametr pro tento směr. Intuitivně, když je obraz I hladký, difuzní parametr by měl být blízko 1: Toto přibližuje izotropní difuzi ("Gaussovské rozmazání"). Na druhé straně, pokud obraz obsahuje textury, difuzní parametr bude menší aby předcházel rozmazání hran u hranic. Po přerozdělení intenzitních hodnot, obraz je upraven aby odrážel nové intenzity:*

$$I^{t+1} = I^t + \lambda(c_N \triangle I_N - c_S \triangle I_S + c_E \triangle I_E - c_W \triangle I_W) \tag{4.33}$$

kde λ *je integrační konstanta (*$0 \leq \lambda \leq 1/3$*) pro čtyři sousedy). Pro získání odšuměného obrazu na určité úrovni* t*, obraz na úrovni* $t - 1$ *je odšuměn. První úroveň může být získána odšuměním originálního obrázku (úroveň 0). Nakonec, malá* λ *udává lepší aproximaci originální rovnice 4.29, ale potřebujeme více iterací pro odšumění obrazu.*

Přidání čtyř diagonálních sousedů má za výsledek

$$\begin{aligned} I^{t+1} = I^t + \lambda((&c_N \triangle I_N - c_S \triangle I_S + c_E \triangle I_E - c_W \triangle I_W) \\ &+ \frac{1}{2}(C_{NW} \triangle I_{NW} - c_{NE} \triangle I_{NE} + c_{SE} \triangle I_{SE} - c_{SW} \triangle I_{SW})) \end{aligned} \tag{4.34}$$

s $0 \leq \lambda \leq 1/7$*. Všimněte si faktoru* $\frac{1}{2}$ *kvůli větší vzdálenosti k diagonálnímu pixelu. Z důvodu této větší vzdálenosti, tento pixel by měl mít menší vliv na pixel který uvažujeme. Při aplikování druhého derivátu, toto vydává faktor* $\frac{1}{2}$*.*

Pojem $\triangle I_N$ *může být získán z jednoduché konvoluce:*

$$\triangle I_N = I * g, g = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{4.35}$$

a nápodobně pro ostatní směry

Nakonec je zapotřebí vědět kde se musí difuze objevit a kde ne; to znamená, musíme mít odhad okraje abychom obdrželi difuzní parametr. Perona a Malik používají spád obrazu jako difuzní parametr. Malý spád se objevuje tam, kde je oblast homogenní, což je to kde chceme

aby se difuze (proto velký difuzní parametr) objevila a naopak. Navrhují dvě různé difuzní funkce založené na spádu:

$$\begin{aligned} c(x,y,t) &= \exp(-(|\triangle I|/K)^2) \\ c(x,y,t) &= \frac{1}{1+(|\triangle I|/K)^2} \end{aligned} \tag{4.36}$$

Hodnota K je určena při každé iteraci. Prvně se vypočítá spád celého obrazu na předchozí úrovni (na první úrovni se bere originální obraz). Po vzetí absolutní hodnoty tohoto spádu, vypočítá se histogram a hodnota pod kterou se objevuje 90% intenzitních hodnot je určena jako K. Po nastaveném počtu iterací je získán odšuměný obraz $F(I)$[14].(překlad autora)

Kapitola 5

Analýza současného stavu a návrh řešení

V této kapitole se věnuji shrnutí zásadních informací na základě předchozích kapitol. Pro smysluplnost práce ji také specializuji určitým směrem a vytyčuji cíle pro experimenty. Porovnávám zde popsané důvody zabezpečení proti falzifikaci snímků a objasňuji důvod výběru tématu identifikace na základě vzorového šumu senzoru. Závěrem vysvětluji zvolení technologických prostředků pro implementaci zamýšlených experimentů.

5.1 Specifikace zaměření práce a stanovení cílů

Výsledky programů je potřeba ověřit a vyvodit z nich efektivitu jednotlivých metod. Zkoumání by mělo být založeno na variaci velikosti, počtu barevných kanálů vstupních dat a použitých filtračních metod. Dále by se měla vyhodnotit úspěšnost rozpoznání senzoru podle zvolených fotografií.

5.2 Porovnání způsobů zabezpečení senzorů

Pro zjednodušení a lepší přehlednost porovnání sumarizuji požadavky a vlastnosti jednotlivých metod v následujících odstavcích.

Při identifikaci fotografie podle souboru je zapotřebí pořizovat fotografie zařízením, které informace o sobě automaticky vyplňuje do souborů s obrazy, nebo musí fotograf všechny informace vyplnit ručně sám. Jak již bylo v popisu metody řečeno, metoda není absolutně dostatečná pro zabezpečení proti podvržení. Metoda je vhodná maximálně pro třídění souborů a osobní využití.

Vodoznakem se dá fotografie efektivně chránit proti krádeži, například robustní vodoznak překrývající celou fotografii pomůže odhalit plagiátorství a při falzifikátech upravených s nízkou námahou lze poznat i upravenou oblast. Křehké vodoznaky jsou velmi nadějné pro zachování integrity snímků a při korektní implementaci jsou takřka neoblomné. V některých případech křehké vodoznaky dokáží i ukázat oblast obrazu která byla přepsána, což je velmi efektivní metoda obrany. Na druhou stranu je pro průkaznost nutné počítat s vodoznakováním už od doby pořízení obrazu a explicitně implementovat vodoznak do obrazu před jakoukoliv manipulací. Může se stát, že se kompresí obrazu vodoznak poruší a není

nadále čitelný, tím pádem metoda selže. Křehké vodoznakování se může používat například jako podpis jakéhokoliv digitálního obrazu (včetně renderů).

Použitím biometrické stopy se zajistí identifikace fotografa a zápis detailů okolností při pořizování fotografie. Jedná se o efektivní prostředek zabezpečení. Tato metoda je prakticky vylepšením vodoznakování za cenu složitějšího návrhu a potřebných dodatečných komponent ve fotoaparátu. Je zde potřeba člověka, který fotografii "podepisuje"svými biometrikami a metoda tím pádem není použitelná například u snímků z bezpečnostních kamer. Ideálním případem užití je fotografování důkazů forenzními techniky při dokumentaci místa činu.

Analýzou pixelových defektů lze identifikovat poškozené, nekvalitní, nebo zastaralé senzory. Metoda se ovšem v praxi neuchytila a je aplikovatelná pouze na omezenou skupinu senzorů. Tento typ identifikace může být efektivní při porovnávání malého počtu podobných snímků a určitým kladem je nevyžadování přímého fyzického přístupu k pořizovacímu zařízení.

Jelikož se vzorový šum objevuje v každém obrazovém senzoru, není zde žádný speciální požadavek na externí implementaci identifikátoru. Ve vzorovém šumu se promítnou i defekty pixelů, takže přesnosti metody pomohou i součásti analýzy pixelových defektů. K přibližnému určení otisku senzoru není potřeba fyzický přístup k senzoru a pro senzor se otisk v čase mění pouze minimálně a to jen defekty, proto je možné pracovat se snímkovou sadou po celou dobu jeho životnosti. Na druhou stranu je potřeba velké množství obrazů pro aproximaci otisku jednoho senzoru a pokud jsou obrazy komprimovány vysoce ztrátovou kompresí, snižuje se efektivita metody.

Jako výslednou metodu jsem vybral identifikaci senzoru podle vzorového šumu. Na metodě je pro mě nejpřitažlivější koncept efektivity pro všechny existující senzory, pokud by metoda fungovala opravdu všude efektivně v rámci možností, jednalo by se o nejuniverzálnější způsob zabezpečení. Při porovnání s ostatními metodami zde není nárok na přidání externí implementace do obrazového zařízení. Při získání fyzického přístupu k zařízení je možné provádět analýzu bez předchozích úprav.

5.3 Požadavky pro implementaci a experimenty

Hlavním požadavkem pro možnost zkoumání metody identifikace obrazových senzorů je přístup k dostatečnému množství fotografií pořízených senzory (obrazových sad) o jejichž identifikaci se budu pokoušet. Minimální potřebný počet se v různých výzkumech liší, ale většinou se jedná o hodnotu přibližně 50 snímků. Obsah snímků určených pro aproximaci je většinou doporučován jako dobře osvětlené netexturované pozadí, daly by se například pořídit fotografie čistého nebe nebo rovnoměrně osvíceného papíru, při použití takových postupů minimální hranice potřebného počtu klesá, jelikož se PRNU projevuje v takovém případě nejvíce, ale není problém použít i průměrné snímky běžně osvětlených scén fotografovaných běžnými uživateli. Zkoumané obrazy u kterých potřebuji určit jejich původní senzor by neměly pocházet z obrazových sad určených pro aproximaci vzorového šumu, proto je tedy potřeba získat ještě snímky u kterých se určuje původ. Pro ověření možnosti rozpoznání dvou senzorů stejné výroby od sebe, je zapotřebí obrazových sad alespoň ze dvou zařízení stejného typu a výrobního procesu.

Získané obrazové sady se analyzují v programu, který umožňuje jejich načtení, filtrování a aproximaci výsledného vzorového šumu různými postupy. Vzorové šumy musí být poté

porovnány se zbytkovými šumy přefiltrovaných snímků za použití stejného filtru který byl použitý při aproximaci otisku. V teoretické části byly navrženy 2 metody pro postup získávání otisku a porovnání obrazu, proto by pro výsledné porovnání bylo vhodné, kdyby programy dokázaly pracovat s oběma metodami.

Kapitola 6

Implementace

Tato kapitola se věnuje popisu požadavků a provedení softwarové implementace programů pro získávání vzorového šumu a porovnávání zbytkových šumů se vzorovými otisky.

6.1 MATLAB

Pro implementaci obou výpočetních částí jsem se rozhodl využít programové prostředí MATLAB R2020a. Jsou v něm zapsány všechny mnou vytvořené výpočetní skripty pro získávání otisků a porovnávání zkoumaných obrazů, využívám i cizích implementací pro statickou funkci PCE a křížovou korelaci. Vestavěné matematické a filtrační funkce disponují výbornou dokumentací a ušetří spoustu implementačního času oproti jiným programovacím jazykům. V MATLABu je možné také explicitně upravit vestavěné funkce a přizpůsobit je podle svých potřeb, čehož využívám například při implementaci CAGI filtru vylepšením Guided Image Filteru.

6.2 Struktura pracovního adresáře

Adresářová struktura pro správný chod skriptů musí být následovná (f v závorce za názvem značí složku,<> značí volitelný název):

- filtered(f) →složka kam se průběžně ukládají filtrované fotografie ze vstupních datasetů
- Functions(f)
  - matlab-edit(f)
    - algcaimguidedfilter.m →upravená vestavěná funkce pro CAGIF
    - caimguidedfilter.m →upravená vestavěná funkce sloužící k získání parametrů pro volání algcaimguidedfilter.m
  - cleanArtefacts.m →funkce pro dočištění JPEG artefaktů z vypočítaného otisku senzoru
  - crop.m →funkce pro vyříznutí středu načteného obrazu
  - croscorr.m →převzatý kód funkce provádějící křížovou korelaci[1]

[1] Funkce byla převzata z http://dde.binghamton.edu/download/camera_fingerprint/

    - getPRNU.m →funkce pro extrakci otisku ze zadaného datasetu
    - PCE.m →převzatý kód funkce vypočítávající statistiku PCE[2]
    - TV.m →funkce pro vypočítání totální variance obrazu
- images(f) →složka se vstupními obrazy pro vypočítání otisku senzoru
    - <název kamery>(f)
        - <název datasetu>(f)
            - <název fotografie>.(jpg/png)
            - ...
        - ...
    - ...
- matchingImages(f) →složka se vstupními obrazy pro porovnání s otisky
    - <název zkoumaného datasetu>(f)
        - <název fotografie>.(jpg/png)
        - ...
    - ...
- PRNU(f) →složka pro vypočítané otisky metodou B
- PRNU2(f) →složka pro vypočítané otisky metodou A
- results(f) →složka pro výsledky porovnání vstupních obrazů s otisky senzorů
- compareImages.m →skript pro porovnání vstupních obrazů s otisky senzorů
- extractPRNU.m →skript pro vypočítání otisků senzorů ze vstupních datasetů

6.3 Aproximace otisku senzoru

Otisky senzoru se aproximují spuštěním skriptu `extractPRNU.m` vstupní parametry jsou zapsány přímo v kódu na začátku skriptu. Jedná se o

- fnc →Textový řetězec obsahující zkratku použité funkce pro používaný filtr, možnosti jsou : "cagif", "gif", "wiener2", "diffexp"a "diffquad"
- resolution →Pole o velikosti 2 s požadovanými rozměry otisku ve formátu [šířka výška] (pokud hodnota přesahuje velikosti obrazů v datasetu, sníží se na jejich velikost)
- grayScale →booleovská hodnota true/false udávající zda se budou obrazy převádět před filtrací na stupně šedi (true pro převod)

[2]Funkce byla převzata z http://dde.binghamton.edu/download/camera_fingerprint/

extractPRNU.m

Skript vyhledá ve vstupní složce `images/` všechny senzory a jejich datasety. Pro každý z datasetů zavolá funkci `getPRNU`, která vrátí otisk vypočítaný z datasetu, a poté získaný otisk uloží ve tvaru `<název kamery>_<název datasetu>_<použitá funkce pro filtraci>_<počet použitých obrazů z datasetu>.mat` (například `C_2_JPG_RANDOM_cagif_95.mat`).

Otisky jsou ukládány do složek `PRNU/` jako proměnné `PRNU` pro metodu B a `PRNU2/` jako proměnné `PRNUavg` pro metodu A. Skript vypíše dobu aproximace otisku pro každý dataset .

getPRNU.m

V souboru je zapsána funkce, která ze zadaných souborů(parametr `files`) vyfiltruje obrazy a z nich poté aproximuje šum metodou, která je zadána v parametru `fnc`. V parametru `dimensions` se očekává pole o velikosti 2 ve tvaru `[šířka, výška]`, které udává největší možné rozměry výsledného otisku. Posledním parametrem je boolean `grayScale`, který udává, zda se vstupní obrazy převádějí na stupně šedi před filtrací.

První nalezený soubor je referenčním pro zbytek souborů, takže se podle vstupních parametrů a jeho velikosti zvolí finální velikost vypočítaného otisku.

Každý zadaný obraz je postupně načten do paměti a oříznut na kýženou velikost finálního otisku. Obraz je poté přefiltrován kýženou metodou v parametru a jeho zbytkový šum je započítán podle obou metod A i B. Vyfiltrovaný obraz je poté uložen do složky `filtered/images/` se zapsaným filtrem v názvu a se stejnou složkovou hierarchií jako má vstupní obraz. Kroky se opakují, do té doby než se projdou všechny zadané soubory v parametru `files`.

Ke konci program uloží výsledky do proměnných `PRNU` pro metodu B a `PRNUavg` pro metodu A. Proměnné se potom upraví funkcí uloženou v `cleanArtefacts.m`, která odečte průměr řádků od každého pixelu, tím pádem výsledné otisky budou mít průměr v nule - tímto se odstraní lineární vzor a zachovají se jeho parametry.

6.4 Porovnání vstupních obrazů s aproximovanými otisky

Vstupní obrazy se s aproximovanými otisky porovnávají spuštěním skriptu zapsaného v `compareImages.m`.

compareImages.m

Skript po spuštění vyhledá ve složce `matchingImages/` všechny složky obsahující obrázky a každou složku bere jako oddělený dataset. V každém datasetu načítá zkoumané fotografie po jedné. Každou fotografii z daného datasetu profiltruje všemi implementovanými filtry, nejprve se všemi barevnými kanály, a poté ve stupních šedi. Po přefiltrování fotografií projde celé složky `PRNU/` a `PRNU2/` a načte předem aproximované otisky skriptem uloženým v `extractPRNU.m`. Pro korektní funkcionalitu je nutné, aby každý z aproximovaných otisků byl vypočítaný minimálně za použití filtru CAGIF oběma metodami (aproximační skript počítá s oběma metodami automaticky), ostatní filtry jsou volitelné a pokud by otisky pro ně chyběly, výsledky budou vycházet nulové. Pro každý načtený otisk se poté vypočítá podobnostní statistika podle adekvátní metody a postupu(viz tabulka druhů postupů 6.1). Pokud je obraz nebo otisk menší než požadovaná velikost oblasti, zvolí upraví se velikost podle nejmenšího porovnávaného prvku pro aktuální cyklus. Výsledky se uloží

do složky `results/` ve tvaru `<název porovnávaného datasetu>_<postup>.xls`(například `panaC3_H.xls`). Struktura souborů s výsledky je zapsána v následující tabulce 6.2.

Identifikátor postupu	Metoda	Filtrování před převedením fotografie na stupně šedi	Velikost porovnávané oblasti
A	B	ano	celý obraz oříznutý 32px od okrajů
B	B	ne	celý obraz oříznutý 32px od okrajů
C	B	ano	512x512 uprostřed
D	B	ne	512x512 uprostřed
E	A	ano	celý obraz oříznutý 32px od okrajů
F	A	ne	celý obraz oříznutý 32px od okrajů
G	A	ano	512x512 uprostřed
H	A	ne	512x512 uprostřed

Tabulka 6.1: Druhy postupů při porovnávání obrazu ve skriptu

Taková variace postupů zajišťuje možnost ověření výsledného vlivu všech uvedených faktorů.

Název fotografie	CAGIF	Anizotropní difuze exponenciální	Anizotropní difuze kvadratická	GIF	Wiener2	Název otisku
S1060017 .JPG	0.023844	0.042219	0.042177	0.05884	0.054136	C_3_JPG _FLAT _cagif_51 .mat

Tabulka 6.2: Formát výsledků s ukázkovými daty

6.5 Implementace obrazových filtrů

Jak již bylo nastíněno ve vstupních parametrech aproximačního skriptu, ve finální implementaci používám 5 druhů filtrů. Jedná se o CAGIF, GIF, Wienerův filtr a dva typy anizotropní difuze (exponenciální a kvadratická). Vlnkový filtr není použit jelikož byl podle předchozích zkoumání výkonnostně překonán. Pro WNNM sice existuje implementace v

MATLABu přímo od návrhářů této metody, ale ve finální verzi skriptů není využitý. Způsob filtrování je natolik komplexní, že by mi nevystačil výpočetní čas v rozsahu této práce.

Anizotropní difuze je již implementována v MATLABu jako funkce `imdiffusefilt()`. Pokud se funkce zavolá s jediným parametrem a tím je obraz, používá exponenciální vyhodnocování. Poskytnutím dalších parametrů je možné vybrat kvadratický typ vyhodnocení a toho se dá docílit zapsáním parametrů `'ConductionMethod','quadratic'` za sebou.

Wienerův filtr je také základní součástí MATLABu, je možné ho zavolat funkcí `wiener2()`. Jako první parametr funkce očekává černobílý obraz(stupně šedi) a jako druhý parametr očekává vektor se dvěma hodnotami udávající šířku a výšku zkoumaného okolí. Výsledkem je přefiltrovaný obraz. Jak získávám parametr velikosti okolí popisuji v sekci 6.6.

GIF je v MATLABu implementován jako funkce `imguidedfilter()`. Jako parametry očekává filtrovaný obraz a referenční obraz. Druhý argument není potřebný a pokud se nepoužije, tak se jako referenční obraz se bere vstupní obraz. Funkci se dá také zadat velikost okolí (6.6) používaného pro filtrování, podobně jako je tomu u Wienerova filtru. Rozměry okolí se zapisují také ve vektoru se dvěma hodnotami, ale parametru velikosti musí předcházet parametr `'NeighborhoodSize'`. Filtrování obrázku by mohlo být zapsáno například následovně: `vysledek = imguidedfilter(image, 'NeighborhoodSize', [sirka vyska]);`.

CAGIF

Filtr CAGI je málo známým rozšířením GIF a jako takový není v MATLABu implementovaný, implementace použitá v [29] není veřejně dostupná. Při prvotních pokusech o vlastní implementaci jsem sice dosáhl korektních výsledků, ale výpočetní náročnost byla příliš vysoká a v žádném případě se ani zdaleka neblížila rychlosti demonstrované v [29].

Konečným řešením se stalo nakonec upravení základní implementované funkce pro výpočet GIF `imguidedfilter()`. Její implementační algoritmus pro GIF je popsán v algoritmu 2. Jako vlastní úpravu jsem do algoritmu vnesl váhování a to přidáním dalších kroků mezi 2. a 3. krok dosavadního pseudoalgoritmu podle rovnic 4.25 a 4.24. Výsledkem nahradím hodnotu epsilon ve třetím kroku.

Mnou upravený pseudo-algoritmus který používám pro výpočet CAGIF je tedy zapsán následovně v algoritmu 3. Kde jsou všechny konstanty zvolené podle [16], takže $v_1 = (0,001 * L)^2$ kde L je dynamický rozsah obrazu. Úpravou vestavěných funkcí jsem dosáhl mnohem efektivnějšího výpočtu než vlastní implementací.

Bylo zapotřebí upravit funkce `imguidedfilter()` a `algimguidedfilter()`. První funkce slouží pro parsování parametrů a poté volání funkce druhé ve které jsou již prováděny výpočty. Mnou upravené funkce jsem pojmenoval `caimguidedfilter()` a `algcaimguidedfilter()`. Tím pádem je postup pro filtrování CAGIFem totožný s postupem pro GIF, jen se upraví název volané funkce.

6.6 Parametr velikosti okolí pro filtrační funkce

Wienerův filtr, CAGIF a GIF používají parametr okolí. V matematické definici je tento parametr zapsán jako radius r. Parametr určuji na stejné bázi jako v [29] a to:

$$r = \begin{cases} 6 - \text{round}(\text{TV}/8) & \text{pro TV} < 36 \\ 2 & \text{jindy} \end{cases} \tag{6.1}$$

Určený parametr r je předáván funkci vždy vektorem o velikosti dvou, kde jsou hodnoty výšky a šířky zadány jako r.

Algoritmus 3: Content adaptive guided image filter

1: $\text{mean}_I = f_{\text{mean}}(I, r)$
$\text{mean}_p = f_{\text{mean}}(p, r)$
$\text{corr}_I = f_{\text{mean}}(I. * I, r)$
$\text{corr}_{Ip} = f_{\text{mean}}(I. * p, r)$

2: $\text{var}_I = \text{corr}_I - \text{mean}_I .* \text{mean}_I$
$\text{cov}_{Ip} = \text{corr}_{Ip} - \text{mean}_I .* \text{mean}_p$

3: $\text{EAW} = ((\text{var}_I + v_1)./(\text{mean}_I^2 + 10^-9))^{0,75}$
4: $\text{divider} = 1./\text{EAW}$
5: **for** *každé J v EAW* **do**
6: weight[indexJ] = průměr(J.*divider)
7: **end for**
8: $\text{divider} = 1./\text{EAW}$

9: $a = \text{cov}_{Ip}./(\ \text{var}_I + (64. * \text{weight}))$
$b = \text{mean}_p - a .* \text{mean}_I$

10: $\text{mean}_a = f_{\text{mean}}(a, r)$
$\text{mean}_b = f_{\text{mean}}(b, r)$

11: $q = \text{mean}_a . * I + \text{mean}_b$

Pro výpočet TV používám funkci zapsanou v souboru `TV.m`. Funkce TV() projde sekvenčně všechny pixely v zadaném obrazu pro každý barevný kanál. Pro každý pixel vypočítá derivace(rozdíly) směry do sousedů vpravo a dolů a všechny je sečte. Výsledný součet se poté podělí počtem derivací pro získání průměru, který finálně udává hodnotu TV.

Kapitola 7

Experimenty a vyhodnocení

Tato kapitola je věnována popisu vybavení použitého pro pořízení fotografií a výpočtu výsledků. V prvním experimentu vyhodnocuji získané výsledky a z těchto vyhodnocení vyvozuji závěry pro efektivitu použitých metod. Poté aplikuji získané informace pro ověření případů z reálného světa v experimentu B.

7.1 Vybavení

K dispozici jsem měl 6 kamer a 2 fotoaparáty, jejich přehled je v následující tabulce 7.1

Identifikátory	Výrobce	Typ	Počet
AE,B0,B1	Manta	G125-B(černobílá)	3
C1,C2,C3	Panasonic	HC-VX980EP-K	3
EK-GN120, GNX	Samsung	Galaxy NX EK-GN120+	2

Tabulka 7.1: Přehled kamer použitých v experimentech

Použití kamer MANTA

Průmyslové kamery Manta G125-B jsou napájeny skrz PoE a pro jejich bezproblémový chod je zapotřebí využití standardů IEEE 802.3 1000BASE-T (Gigabit Ethernet) a IEEE 802.3af (PoE). Pro optimální rychlost a zamezení elektromagnetické interference se musí použít FTP kabely 6. kategorie[26]. Jelikož se potřebuje zařídit gigabitová rychlost a napájení skrz ethernet, musí se použít speciální switch, který předchozí standardy podporuje - já jsem použil Switch Tenda TEG1105P PoE. Bohužel switch neumí pracovat s tzv. jumbo frames, čímž se snížila přenosová frekvence snímků na 24FPS. V této práci to však není na škodu, protože zkoumám pouze fotografie a ne videa. Pro zachycení snímků jsem použil software dodávaný výrobcem Vimba Viewer verze 2.2.1.

Počítačové vybavení

Všechny výpočty byly prováděny na osobním počítači s operačním systémem Windows 10 Pro verze 1903, s procesorem AMD Ryzen 5 3600(6 jader, 3.6GHz) a s kapacitou RAM paměti 16GB. Jelikož je daný procesor rychlejší při výpočtech používaných k filtraci než moje aktuální grafická karta, upustil jsem od výpočtů na grafické kartě a ta by neměla mít na rychlost výpočtu ani výsledky vliv. Verze programu MATLAB 9.8.0.1359463 Update 1.

7.2 Datové sady

Každým zařízením jsem pořídil datové sady(většinou přes 50 snímků na jednu sadu) pro aproximaci otisků jejich senzorů a alespoň 10 "běžných"snímků pro finální porovnávání, u těch je důležité aby nebyly zahrnuty v žádném z datasetů určených pro aproximaci otisku. Datových sad je dohromady 18 a jejich detaily jsou zapsány v tabulce 7.2, pro úsporu místa jsou v řádcích sady se stejnými názvy kombinovány.

Kamera	**identifikátor**	**počet snímků**	**obsah snímků**
C1,C2,C3	FLAT	76,53,51	lehce texturovaná bílá zeď
C1	OUTSIDE	52	náhodné venkovní snímky v areálu fakulty
C1	PAPER	158	kombinace rovnoměrně osvětleného papíru a snímků z vnitřku knihovny FIT
C2	PAPER	177	rovnoměrně osvětlený bílý papír
C2,C3	INDOOR	47,38	vnitřek knihovny FIT
C2	RANDOM	95	náhodné snímky s občasným použitím přiblížení
C3	108CDPFQ	138	kombinace rovnoměrně osvětleného papíru a snímků z vnitřku knihovny FIT
AE,B0,B1	OUTSIDE_SUNNY	55,54,55	výhled z okna za velmi slunného počasí a záběry silnice
AE,B0,B1	OUTSIDE_ CERVANKY	60	výhled z okna v podvečer při menším světle
GNX,EK-GN120	SUNNY	52,53	náhodné venkovní snímky za slunného počasí

Tabulka 7.2: Obsah datasetů pořízených dostupnými zařízeními

Fotografie pořízené kamerami Panasonic byly nafoceny v automatickém režimu a výsledném rozlišení 4992x2808 ve formátu JPEG. Fotoaparáty Samsung Galaxy mají rozlišení 5472x3080, nastavení bylo použito na automatický režim a fotografie jsem uložil ve formátu SRW který jsem později převedl na nejkvalitnější možný JPEG. Kamery značky manta mají rozlišení 1292x964 pro dataset SUNNY a rozlišení 692x440 pro dataset CERVANKY. Všechny fotografie z těchto kamer byly uloženy v bezztrátovém formátu PNG.

Aproximace otisků

Všechny uvedené fotografie jsem vložil do složky `images/` v kořenovém adresáři skriptů podle struktury definované v 6.2. A pro každý implementovaný filtr kromě CAGIF spustil skript `extractPRNU.m` s parametry `fnc = «zkratkaFiltru>"`, `resolution = [9999 9999]`, `grayscale = false`. Pro CAGIF byl skript spuštěný s parametry `fnc = "cagif"`, `resolution = [512 512]`, `grayscale = true`.

Přibližný výpočetní čas jednotlivých použití různých filtrů je shrnut v tabulce 7.3

Filtr	**cagif**	**gif**	**wiener2**	**diffexp**	**diffquad**
Čas na snímek průměr	205s	65s	22s	75s	66s
Celkový čas	273470s	86710s	29348s	100050s	88044s

Tabulka 7.3: Přibližná doba výpočtu senzorových otisků ze všech datových sad

Poté jsem vložil fotografie pro porovnávání do složky `matchingImages/` rozdělené do složek podle identifikátoru kamery a spustil skript uložený v `compareImages.m` abych získal výsledky. Skript běžel přibližně 9 hodin.

7.3 Interpretace dat

Výsledná data jsou hodnoty PCE pro metodu B a korelace pro metodu A. Hodnoty vypovídají o míře podobnosti mezi aproximovaným otiskem senzoru a šumem fotografie získaného za použití stejného filtru. Mělo by platit, že čím vyšší hodnota, tím je větší pravděpodobnost původu zkoumané fotografie z daného senzoru. Nemám k dispozici dostatek hodnot pro vytvoření detailního histogramu, ale uvažuji, že získané hodnoty budou vždy pro jeden senzor Normálně rozděleny, proto se by se dala ze získaných hodnot křivka rozložení hodnot aproximovat. Pokud aproximuji jednu křivku(H_0) získanými hodnotami fotografií, které nepocházejí ze senzoru a druhou křivku(H_1) proložím získanými hodnotami fotografií, které pocházejí ze senzoru, tak by se v ideálním případě neměly protnout a metoda by měla 100% efektivitu. Ovšem pokud se křivky budou protínat, lze podle dané hodnoty zjistit alespoň pravděpodobnost pravdivosti hypotézy, že zkoumaná fotografie pochází ze senzoru. Pro křivky musí platit $\int H_0 >= \int H_1$, čím více je dostupných hodnot pro aproximaci H_1, tím přesnější jsou vyvozené závěry o metodě. Detailněji je způsob vyhodnocení popsán grafem 7.1.

Kde H_0 je vyznačená modře a H_1 červeně. Průsečík křivek P je v bodě (0.08759,6.865).

Práh pro určení původu fotografie se dá zvolit jako EER(equal error rate) podle průsečíku kritérií FAR(false acceptance rate) a FRR(false rejection rate). Hodnota FAR pro hodnotu zkoumané fotografie x udává, jaká je pravděpodobnost nepravdivého určení původu fotografie z daného senzoru. Hodnota FRR pro x udává, jaká je pravděpodobnost nepravdivého odmítnutí původu fotografie z daného senzoru. Při prvním pohledu na graf 7.1 je jasné, že hodnoty FRR a FAR jsou v poměru 1:1 v bodě P a tím pádem se dá průsečík považovat jako EER, čímž získáváme hodnotu prahu. Hodnoty FRR a FAR pro práh t zvolený jako x-ová souřadnice průsečíku P se vypočítají jako 7.1

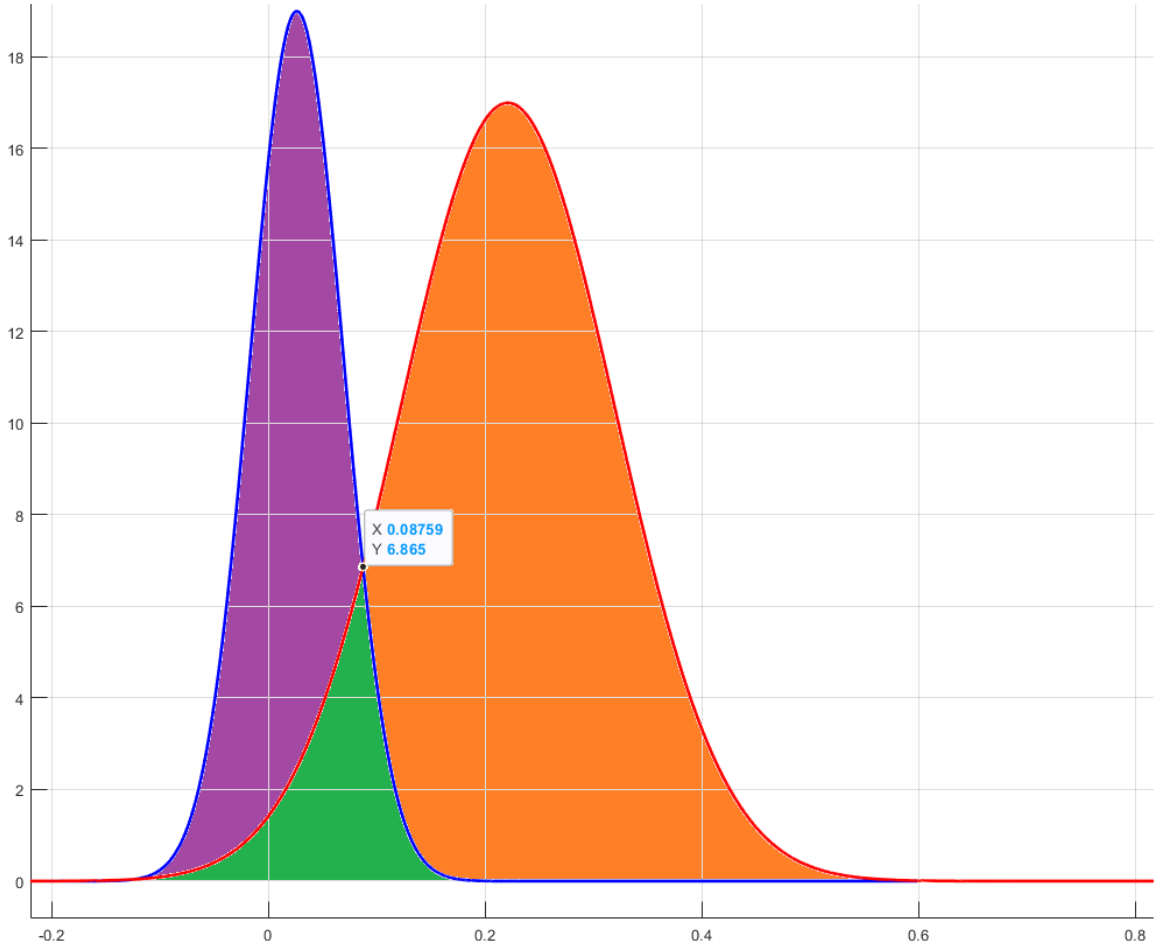
$$FAR(t) = FRR(t) = \frac{\int_{-\infty}^{t} H_1}{\int_{-\infty}^{t} H_0 - H_1} \tag{7.1}$$

Definuji-li hodnotu snímku a, pro kterou platí $a > t$, dá se pro ni určit pravděpodobnost korektní identifikace $p(a)$ podle 7.2.

$$p(a) = 1 - FRR(a) = 1 - \frac{\int_{a}^{\infty} H_0}{\int_{a}^{\infty} H_1} \tag{7.2}$$

Uvedené výpočty prakticky značí, že nejúčinnější metoda je taková, která má největší poměr oranžové oblasti vůči zelené v grafu 7.1. Nejúčinnější metoda nemá žádnou zelenou oblast. Výslednou účinnost vyhodnocení pro dané křivky (poměr oranžové a zelené oblasti) lze vyjádřit rovnicí 7.3

$$\text{Účinnost} = \frac{\int_{t}^{\infty} H_1 - H_0}{\int_{-\infty}^{P} H_1 + \int_{P}^{\infty} H_0} \tag{7.3}$$



Obrázek 7.1: Exemplární znázornění interpretace dat

7.4 Experiment A

Pro určení původu fotografie, je potřeba ze všeho nejdříve znát efektivitu zkoumaných metod které budou k identifikaci využívány. Zkoumání efektivity nazývám experimentem A. V experimentu používám všechna výstupní data která jsem získal skriptem `compareImages.m` - datové sady znázorněné v tabulce 7.2.

Vyhodnocení

Výpočty provádím přiloženým skriptem `makeGraphs.m` a funkcí `makeGraph.m`, kde prokládám histogramy hodnot normálním rozložením podle 7.3. Zkoumané intervaly znázorněné v grafu mají 10^6 vzorků po celé šířce, stejné vzorkování je používáno pro numerický výpočet integrálů.

Pro označení postupů používám značení z tabulky 6.1. Formát pro grafy používám stejný jako v obrázku 7.1, to znamená červená barva pro hodnoty snímků pocházejících ze senzoru a modrá barva pro cizí snímky.

Nejefektivnější postup

Hodnoty korelací pro metodu A, tedy postupy E-H, jsou podobné hodnotám očekávaným a dají se analyzovat statistikou účinnosti kterou jsem definoval v 7.3. Pro každou kombinaci metody, otisku a filtru jsem sestrojil graf a vypočítal hodnotu účinnosti. Nejefektivnější postup z množiny E-H vybírám na základě poměru hodnot účinností. Pro každou hodnotu účinnosti (a_i) metody a odečtu relevantní(stejný otisk a použitý filtr) hodnotu účinnosti(b_i) z metody b, kde $a \neq b$ a rozdíl je $z_{a,b,i}$. Poté pro každý pár a, b vyberu počet výsledných hodnot které jsou > 0 (počet$(z_{a,b} > 0)$), a poměřím ho s počtem všech výsledných hodnot počet$(z_{a,b})$,

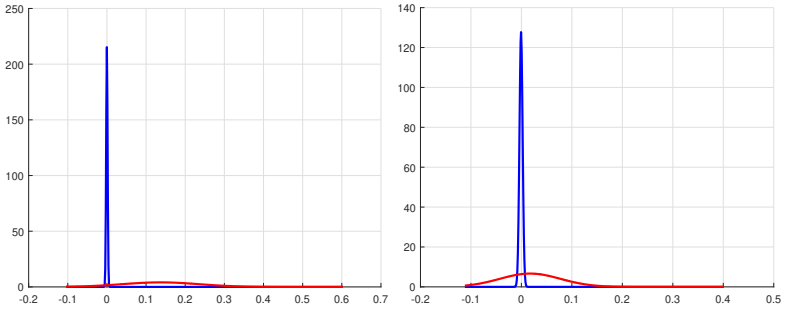
$$f(a,b) = \frac{\text{počet}(z_{a,b} > 0)}{\text{počet}(z_{a,b})} \tag{7.4}$$

Výsledné hodnoty jsou zapsány v následující tabulce7.4:

$f(a,b)$	**E**	**F**	**G**	**H**
E		0,25	1	0,75
F	0,75		1	0,98
G	0	0		0,4
H	0,25	0,02	0,6	

Tabulka 7.4: Zaokrouhlené výsledky $f(a,b)$ pro metody E-H

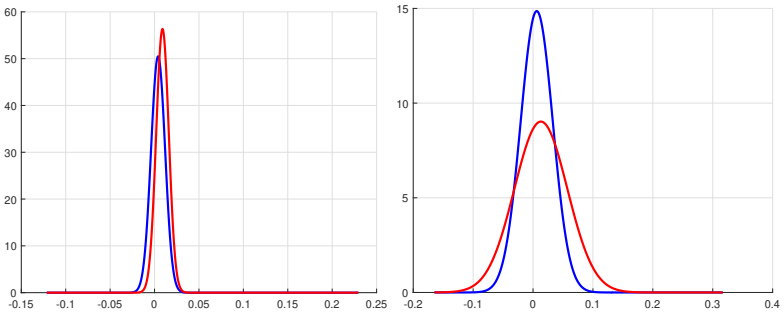
Pokud má výsledek $f(a,b)$ hodnotu větší než 0,5, znamená to, že postup a má větší efektivitu než postup b. Na základě získaných výsledků je nejefektivnější postup F - aproximace šumu z fotografií převedených na stupně šedi před aproximací, v celé velikosti s ořízntutím okrajů o 32 pixelů. Druhým nejefektivnějším postupem aproximace je E o polovinu horším oproti F, třetím H o polovinu horším oproti E a nejhorším vyšel postup G. Porovnání nejlepšího a nejhoršího způsobu je na obrázku 7.2.



Obrázek 7.2: Porovnání grafů nejlepší a nejhorší metody pro stejný otisk a filtr

Z výsledných hodnot účinnosti je zřejmé, že kombinace, které mají účinnost < 1, nejsou použitelné, jelikož je jejich efektivita menší než 50%. Vypsáním hodnot do tabulky u otisků `FLAT` z kamery C1 a `PAPER` z kamery C2, vypočtených jakýmkoliv postupem, je viditelné, že hodnoty účinnosti nikdy nedosáhnou na 1, proto otisky vyřazuji z porovnávání. Grafy pro vyřazené otisky získané postupem E za použití exponenciální anizotropní difuze jsou vykresleny v 7.3.

Postupem G vyšlo vyhodnocení účinnosti pro otisk `PAPER` z kamery C1 menší než 1 pro všechny filtry, proto byl otisk z porovnání postupů vyřazen při výpočtech s postupem G.



Obrázek 7.3: Grafy vyřazených otisků, C1_FLAT, C2_PAPER

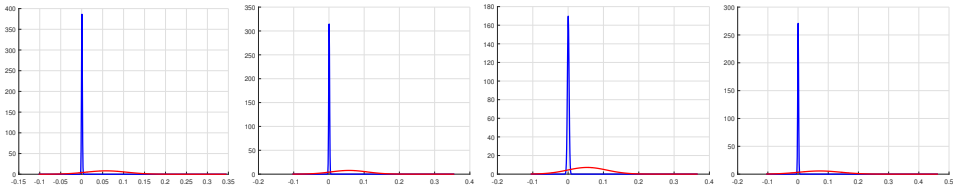
Nejefektivnější filtr

Jelikož jsou nejlepšími postupy zvoleny F a E, provádím porovnání filtrů pouze pro ně.

postup\filtr	**DIFFEXP**	**DIFFQUAD**	**GIF**	**WIENER**
E	13.4560216	9.488342573	3.066982333	8.941184067
F	14.5929374	10.51777793	3.329995133	8.903484867

Tabulka 7.5: Průměrné hodnoty účinnosti metody pro jednotlivé filtry

Do tabulky 7.5 byly zprůměrovány hodnoty účinnosti pro jednotlivé metody. Z dat je vidět zřejmý trend efektivity použitého filtru, na základě kterého volím exponenciálně vyhodnocenou anizotropní difuzi jako nejefektivnější filtr. Pro ilustraci je možné vidět na grafech 7.4 pozorovaný trend efektivity(poměr integrálu za prahem a před ním).



Obrázek 7.4: Vyhodnocení pro otisk SUNNY fotoaparátu EK-GN120.Filtry diffexp, diffquad, gif, wiener

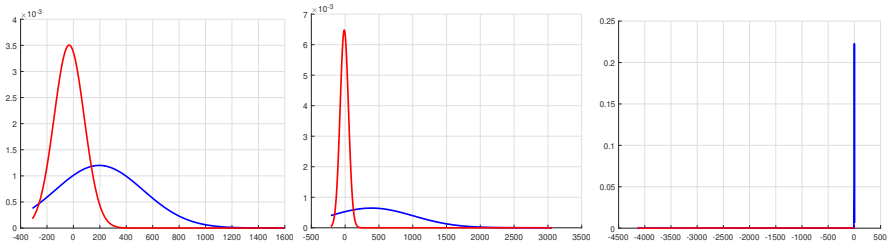
Pro CAGIF jsou relevantní výsledky pouze pro postup H, ale jelikož nebyly hodnoty účinnosti v žádném z případů lepší než pro jakýkoliv jiný filtr, považuji ho za absolutně neefektivní.

Vyhodnocení hodnot PCE (metoda B)

Při prvním pohledu na výsledné hodnoty PCE pro metodu B je zřejmé, že nejsou podobné hodnotám které jsem očekával, například v datech se objevuje hodnota -2786.92914 vypočítaná za použití exponenciální anizotropní difuze pro otisk C_MANTA_AE_OUTSIDE_SUNNY [1], která udává velmi silnou rozdílnost.

[1]Fotografie pochází ze senzoru pro který byl tento otisk aproximován

Na grafech 7.5 sestrojených pro hodnoty vypočítané postupem A[2] je možné vidět extrémní rozdíly mezi pozicemi křivek. Zvolil jsem příkladné zástupce ze 340 možných grafů.



Obrázek 7.5: Postup A, křivky pro C1, C2 a AE

V některých případech je střed křivky H_1 v extrémně záporných hodnotách a v jiných případech se promítá v kladných hodnotách. Při vizuální analýze nemají data zjevný trend a proto s nimi nedokážu pracovat. Pro postupy A-D nedokážu určit efektivitu a proto pouze zobrazuji nestálost dat grafem. Dále se postupy A-D nezabývám a prohlašuji je za nepoužitelné.

7.5 Experiment B

Identifikace původu fotografie je hlavní motivací zkoumání řešení. Nejlépe by se tato činnost dala demonstrovat na reálné situaci pojaté jako slovní úloha, kde potvrzují hypotézu s určitou přesností.

Lze si představit následující situaci : "Fotografovi byl odcizen fotoaparát, ke kterému nemá doklad o koupi a ani jiný důkaz o jeho vlastnictví, ale má sadu neupravených snímků (`dataSet`) pořízených fotoaparátem z dřívější práce s ním. Policie zabavila kradený fotoaparát(F) stejné specifikace a před předáním chce ověřit alespoň s 90% úspěšností, že zabavený fotoaparát je ten, který byl odcizen fotografovi, aby mu ho mohla navrátit. Policie pořídí fotografii(i) fotoaparátem F a porovná ji s otiskem získaným ze snímkové sady `dataSet` fotografa."

Ze situace vyplývá, že je zapotřebí zajistit pravděpodobnost hypotézy(H_0) alespoň 90% pro navrácení fotoaparátu:

$H_0 = i$ pochází ze senzoru s otiskem aproximovaným z `dataSet`

Jako `dataSet` používám datovou sadu SUNNY z fotoaparátu GNX a jako fotografii i používám snímek 7.6 Ze sady `dataSet` se aproximuje otisk $\hat{O}$ za použití anizotropní difuze postupem F a ten se poté koreluje s co největším počtem zbytkových šumů, získaných anizotropní difuzí, fotografií jiného původu než F, pro aproximaci křivky A_0[3]. Vypočítá se hodnota korelace $Corr(\hat{O}, i)$ zbytkového šumu fotografie a aproximovaného otisku. Křivka A_0 a hodnota $Corr(\hat{O}, i)$ se poté vnesou do grafu 7.7.

Hodnota $p(H_0)$ poté udává pravděpodobnost hypotézy H_0

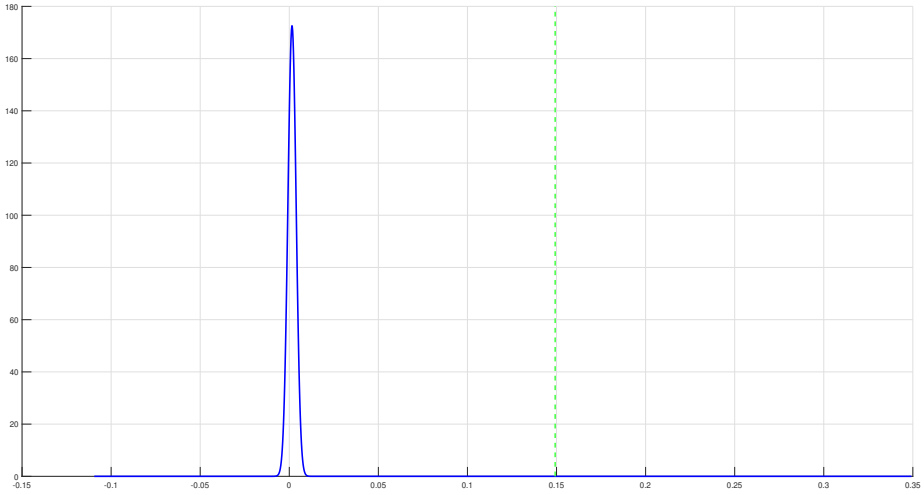
$$p(H_0) = 1 - \int_{Corr(\hat{O},i))}^{\infty} A_0 \tag{7.5}$$

Numericky vypočítaná pravděpodobnost hypotézy je v tomto případě 99%, což značí, že fotoaparát F je původním majetkem fotografa.

[2] V postupu A používám metodu B, viz 6.1

[3] Nutné použít jiné značení než při interpretaci dat protože H_0 je hypotéza

Obrázek 7.6: Snímek i z kradeného fotoaparátu



Obrázek 7.7: A_0 pro $\hat{O}$(modře) s vynesenou hodnotou $Corr(\hat{O}, i)$(zeleně)

Je důležité podotknout, že pro aproximaci křivky $\hat{O}$ byl použitý velmi malý soubor cizích fotografií, v reálném případě by se měly počítat výsledky z histogramů za použití dostatečného množství cizích snímků, nejlépe původem ze senzoru stejné značky a typu výroby. Experiment byl proveden spíše jako demonstrace identifikace s ručně vybranými vstupními daty.

7.6 Další možné experimenty

V předchozích sekcích jsem již ukázal experimenty s praktickými daty, ale pro vyjádření dalších vizí ohledně využití implementované metody navrhuji další případy užití, některé z nich by mohly být s dostatečným počtem vstupních dat dosažitelné. Každá odrážka znamená jeden případ užití.

- Jsem soudním expertem a potřebuji podložit tvrzení soudu, že fotografie byla pořízena daným senzorem s účinností alespoň 97%. Jaká je pravděpodobnost že se mi to podaří?
- Mám přístup ke 100 fotoaparátům stejné výroby a 1 fotografii, u které potřebuji určit původ. Podle požadované přesnosti dokáži alespoň zúžit výběr fotoaparátů.

- Zařizuji kamerový systém určený k strážení cenného majetku a vím, že budu daný postup zabezpečení používat jako důkazní materiály, na jaké společné rysy senzorů se mám zaměřit při jejich pořízení pro nejvyšší efektivitu metod?
- Provozuji kamerový systém a pravidelně kontroluji, zda se přibližný otisk nezměnil. Pokud by se projevila odchylka, vím že byl senzor/zařízení zaměněno za falzifikát a podniknu obratem inspekci zařízení pořizujícího obraz.

Pro některé případy jsou již nastíněny odpovědi v sekci 7.3. Pokrytí takovýchto případů užití se již nevejde do rozsahu vymezeného pro tuto práci, případy užití tedy slouží pro zamyšlení a nasměrování k různým oblastem budoucích výzkumů.

Kapitola 8

Závěr

Cílem této práce bylo zhodnocení aktuálních postupů zabezpečení senzorů a implementace jedné z metod s diskutováním vlastností řešení. Cíl práce byl úspěšně dosažený.

Prostudoval jsem potřebnou literaturu ohledně postupu pořízení fotografie kamerovým senzorem, existujících typech zabezpečení a o mé zvolené metodě identifikace senzoru na základě jeho šumu. Nastudované informace jsou shrnuty v kapitolách 2-4. Na základě těchto informací jsem v kapitole 5. vybral požadované vlastnosti pro implementaci a postupy, kterými jsem jich chtěl dosáhnout. V 6. kapitole jsem popsal, jak jsem postupy implementoval a popsal jsem nástroje, které jsem k tomu využil. V 7. kapitole popisuji získávání dat pro požadované experimenty a porovnávám metody, které jsou použity v této práci. Posledním experimentem stavím nabyté informace do reálného světa a ověřuji postup identifikace na reálném případu.

Zvolil jsem na základě dat nejefektivnější implementovaný postup a filtr, výsledky mě překvapily. Očekával jsem, že vyhodnocení na bázi PCE za použití CAGIF bude nejefektivnější, ale výsledky svědčí o pravém opaku. Je ovšem možné, že převzatá funkce pro výpočet PCE nebyla dobře využita a výsledky jsem pouze neinterpretoval korektně. Dalším objevem byla neefektivita získání vzorového šumu z obrazů rovnoměrně osvětleného papíru v jednom případě, tento jev by se dal ovšem vysvětlit automatickým režimem kamery, která mohla post-processingem fotografie upravit.

Provedením experimentů jsem zjistil, že pro přibližné určování je množina mnou poskytnutých dat dostatečná a funguje bez větších problémů, ale pokud by se řešení využívalo například při forenzní analýze, kde se dbá na co největší přesnost, bylo by potřeba mnohem více vstupních dat pro provedení výpočtů. Je jasné, že tento způsob zabezpečení je smysluplný a přináší výsledky, přesnost se dá dalším výzkumem zlepšit.

V implementovaných postupech hraje roli velké množství faktorů, které dokážou velmi ovlivnit výsledky, pro další pokračování v práci se nabízí další prozkoumání vlivu dalších filtrů, vstupních snímků, barevné interpolace a post-processingu zařízení.

Literatura

[1] Adams, J., Parulski, K. a Spaulding, K. Color processing in digital cameras. *IEEE Micro*. 1998, sv. 18, č. 6, s. 20–30. ISSN 0272-1732.

[2] AV & IT Storage Systems, T. S. C. on a Equipment. *Exchangeable image file format for digital still cameras: Exif Version 2.2, JEITA CP-3451*. Japan Electronics and Information Technology Industries Association, duben 2002 [cit. 1.4.2020]. Dostupné z: https://www.exif.org/Exif2-2.PDF.

[3] Blažek, T. *Systém pro správu a výběr fotografií*. Brno, 2018. Bakalářská práce. Fakulta Informačních Technologií, Vysoké učení Technické v Brně. Vedoucí práce Zemčík, P.

[4] Blythe, P. a Fridrich, J. Secure Digital Camera. *DIGITAL FORENSIC RESEARCH CONFERENCE*. Srpen 2004, [cit. 1.4.2020]. Dostupné z: https://www.dfrws.org/sites/default/files/session-files/paper-secure_digital_camera.pdf.

[5] Chang Tsun, L. a Satta, R. On the Location-Dependent Quality of the Sensor Pattern Noise and Its Implication in Multimedia Forensics. *IET Conference Proceedings*. Leden 2011, s. 37. DOI: 10.1049/ic.2011.0134.

[6] Chen, M., Fridrich, J., Goljan, M. a Lukas, J. Determining Image Origin and Integrity Using Sensor Noise. *IEEE Transactions on Information Forensics and Security*. 2008, sv. 3, č. 1, s. 74–90.

[7] Corum, C. A., Connolly, K. M. a Bawolek, E. J. *Patent: Dark frame subtraction*. Intel Corp, US8259213B2 [cit. 1.4.2020]. Dostupné z: https://patents.google.com/patent/US6101287A/en.

[8] Cressler, J. D. *Silicon Earth: Introduction to Microelectronics and Nanotechnology, Second Edition*. CRC Press, 2016. 12–38 s. ISBN 978-1-4987-0825-8.

[9] Fridrich, J. *Digital Image Forensics Using Sensor Noise* [online]. 2009 [cit. 1.4.2020]. Dostupné z: http://www.ws.binghamton.edu/fridrich/Research/full_paper_02.pdf.

[10] Geradts, Z. J., Bijhold, J., Kieft, M., Kurosawa, K., Kuroki, K. et al. Methods for identification of images acquired with digital cameras. *Proceedings of SPIE - The International Society for Optical Engineering*. Únor 2001, sv. 4232, s. 505–512. DOI: 10.1117/12.417569.

[11] Goljan, M., Fridrich, J. J. a Filler, T. Large scale test of sensor fingerprint camera identification. *IS&T/SPIE Electronic Imaging, 2009, San Jose, California, United States*. SPIE. 2009, sv. 7254, s. 170 – 181. DOI: 10.1117/12.805701.

[12] Gu, S., Zhang, L., Zuo, W. a Feng, X. Weighted Nuclear Norm Minimization with Application to Image Denoising. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014. DOI: 10.1109/CVPR.2014.366.

[13] He, K. a Sun, J. *Fast Guided Filter* [online]. Květen 2015 [cit. 1.4.2020]. ArXiv:1505.00996v1. Dostupné z: https://arxiv.org/pdf/1505.00996.pdf.

[14] Houten, W. van a Geradts, Z. Using Anisotropic Diffusion for Efficient Extraction of Sensor Noise in Camera Identification. *Journal of Forensic Sciences*. 2012, sv. 57, č. 2, s. 521–527. DOI: 10.1111/j.1556-4029.2012.02057.x.

[15] Ishiga, K. *Patent: Digital camera and digital camera system*. Nikon Corp, US8259213B2 [cit. 1.4.2020]. Dostupné z: https://patents.google.com/patent/US8259213.

[16] Li, Z., Zheng, J. a Zhu, Z. Content adaptive guided image filtering. *2014 IEEE International Conference on Multimedia and Expo (ICME)*. 2014, s. 1–6. DOI: 10.1109/icme.2014.6890136.

[17] Lukas, J., Fridrich, J. a Goljan, M. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security*. 2006, sv. 1, č. 2, s. 205–214. ISSN 1556-6013.

[18] Lukáš, J., Fridrich, J. a Goljan, M. Determining digital image origin using sensor imperfections. *Proc SPIE*. Březen 2005, sv. 5685, s. 249–260. DOI: 10.1117/12.587105.

[19] MathWorks. *Wiener2: 2-D adaptive noise-removal filtering* [online]. 2020 [cit. 10.6.2020]. Dostupné z: https://www.mathworks.com/help/images/ref/wiener2.html.

[20] N. Kumar, N. K. Wiener filter using digital image restoration. *International Journal of Electronics Engineering*. 2011, sv. 3, č. 2, s. 345–348. ISSN 0973-7383.

[21] Novozámský, A. Source Camera Identification Based on PRNU Invariant to Zoom. *Doktorandské Dny 2011*. 1. vyd. Praha: České vysoké učení technické v Praze. Listopad 2011, sv. 1, č. 1, s. 163–173.

[22] Parulski, K. A. Color filters and processing alternatives for one-chip cameras. *IEEE Transactions on Electron Devices*. 1985, sv. 32, č. 8.

[23] Sang Won Lee, S. M. Optimized Optomechanical Anti-Aliasing Filter for Digital Camera Photography. *Journal of the Optical Society of Korea*. Říjen 2015, sv. 19, s. 456–466. DOI: 10.3807/JOSK.2015.19.5.456.

[24] Stojkovic, A., Shopovska, I., Luong, H., Aelterman, J., Jovanov, L. et al. The Effect of the Color Filter Array Layout Choice on State-of-the-Art Demosaicing. *Sensors*. Červenec 2019, sv. 19, s. 3215. DOI: 10.3390/s19143215.

[25] Tiwari, M. a Bhupendra, G. Enhancing Source Camera Identification Using Weighted Nuclear Norm Minimization De-Noising Filter: Proceedings of IC4S 2017, Volume 2. *Advances in Intelligent Systems and Computing*. Srpen 2019, sv. 2, s. 281–288. DOI: 10.1007/978-981-13-0344-924.

[26] Vision, A. *GigE VISION CAMERAS Manta Technical Manual*. 2020 [cit. 1.4.2020]. Dostupné z: https://cdn.alliedvision.com/fileadmin/content/documents/products/cameras/Manta/techman/Manta_TechMan.pdf.

[27] Wang, Y., Wang, J. a Gao, Q. A triple-exposure color PIV technique for pressure reconstruction. *Science China Technological Sciences*. Leden 2017, sv. 60. DOI: 10.1007/s11431-016-0270-x.

[28] Wong, P. W. A Watermark for Image Integrity and Ownership Verification. *IS&T's 1998 PICS Conference*. 1998, sv. 18, č. 6, s. 20–30.

[29] Zeng, H. a Kang, X. Fast Source Camera Identification Using Content Adaptive Guided Image Filter. *Journal of Forensic Sciences*. Prosinec 2016, sv. 61. DOI: 10.1111/1556-4029.13017.