

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

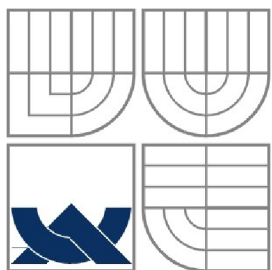
**FORENZNÍ OPTICKÝ KOMPARÁTOR OTISKŮ PRSTŮ**

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

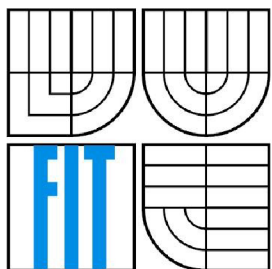
**AUTOR PRÁCE**  
AUTHOR

**Marek Dvořák**

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
**ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ**

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

# **FORENZNÍ OPTICKÝ KOMPARÁTOR OTISKŮ PRSTŮ**

FORENSIC OPTICAL FINGERPRINT COMPARATOR

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

Marek Dvořák

**VEDOUCÍ PRÁCE**  
SUPERVISOR

Doc. Ing, Dipl.-Ing. Martin Drahanský, Ph.D.

BRNO 2008

## **Abstrakt**

Tato bakalářská práce se zabývá návrhem a tvorbou aplikace, která nahrazuje funkci forenzního optického komparátoru. Jsou zde popsány základní vlastnosti otisků prstů a jejich význačné body, které slouží k rozpoznávání osob. Práce s otisky prstů je velmi náchylná na kvalitu jejich obrazu. Kvůli tomu je zde popsán a implementován Gaborův filtr. Dále jsou zde popsány základní funkce pro práci s obrazem.

## **Abstract**

My Bachelor Thesis deals with the design and creation of an application that replaces the function of forensic optical comparator. It also describes features of fingerprints and their prominent points, which are used for recognition of individuals. Work with fingerprints is very sensitive as for the quality of images. For this reason a Gabor filter has been implemented and described. Furthermore, the thesis describes basic functions used for work with images.

## **Klíčová slova**

forenzní optický komparátor, otisky prstů, snímání, lokální orientace hřebenů, lokální frekvence hřebenů, Gaborův filtr, tenčení, segmentace, OpenCV,

## **Keywords**

forensic optical comparator, fingerprint, scanning, local orientation of ridges, local frequency of ridges, Gabor filter, thinning, segmentation, OpenCV

## **Citace**

Marek Dvořák: Forenzní optický komparátor otisků prstů, bakalářská práce, Brno, FIT VUT v Brně, 2015

# Název bakalářské práce v jazyce práce

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Doc. Ing, Dipl.-Ing. Martin Dražanský, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Marek Dvořák  
17. května 2015

## Poděkování

Chtěl bych poděkovat panu doc. Ing, Dipl.-Ing. Martinu Dražanskému, Ph.d. za poskytnutí odborné pomoci při řešení této bakalářské práce a tvorbě technické zprávy.

© Marek Dvořák, 2015

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

Obsah.....	1
1 ÚVOD.....	3
2 Daktyloskopie.....	5
2.1 Papilární linie.....	5
2.2 Fyziologické zákony papilárních linií.....	6
2.2.1 První zákon o neměnnosti obrazců papilárních linií.....	6
2.2.2 Druhý zákon o relativní neodstranitelnosti papilárních linií.....	6
2.2.3 Třetí zákon o individuálnosti papilárních linií.....	7
2.3 Daktyloskopické stopy.....	7
3 Základní popis otisku prstu.....	8
4 Snímání otisku prstu.....	9
5 Zpracování obrazu.....	10
5.1 Segmentace.....	12
5.2 Odhad lokální orientace papilárních linií.....	12
5.3 Odhad lokální frekvence papilárních linií.....	14
5.4 Gaborova filtrace.....	15
6 Tenčení.....	16
6.1 Zhaung-Suen tenčící algoritmus.....	17
7 Návrh aplikace.....	18
7.1 Rozbor problému.....	18
7.2 Použité knihovny.....	19
7.2.1 Knihovna QT.....	19
7.2.2 Knihovna openCV.....	20
7.3 Popis grafického rozhraní.....	21
8 Implementace.....	21
8.1 Použité implementační nástroje.....	21
8.2 Třída OtiskWindow.....	22
8.3 Třída PraveMenu.....	22
8.4 Třída ImageCV.....	23
8.5 Třída OpenCv.....	24
8.6 Třída Screen.....	28
8.7 Třída Snimani.....	28
8.8 Třídy HorniMenu a Menu.....	28
9 Budoucí úpravy a nedostatky.....	29

9.1 Zjištěné nedostatky.....	29
9.2 Možné rozšíření aplikace.....	30
9.3 Popis ovládacích prvků.....	30
10 Závěr.....	32
Literatura.....	33
Seznam příloh.....	34

# 1 ÚVOD

V současné době se s problémem identifikace osob setkáváme na mnoha místech. Nemusí se přitom nutně jednat o kriminalistické prostředí a pátrání po zločincích, potýkáme se s ním v každodenním životě. Principy identifikace využíváme jakožto soukromé osoby ve svých telefonech, při zabezpečování různých objektů apod.

Metody rozpoznávání jedinců jsou různé. Můžeme užít třeba různé druhy karet. Pro příklad nemusíme chodit daleko, například náš oběd v menze získáme na základě takovéto karty.

Další možností je takzvaný PIN kód. Tato zkratka vznikla z anglického personal identification number. Tento, obvykle čtyřmístný kód, nám slouží k autorizaci při používání našich mobilních telefonů. Zadáváme jej po zapnutí tohoto zařízení. Tato metoda je stále nejrozšířenější, avšak s vývojem technologií vnímáme také nové způsoby identifikace telefonů. Naši chytrí pomocníci nás dokáží rozpoznat na základě obličeje nebo otisku našeho prstu. [1]

Některá zařízení využívají více takových principů. Například v bance se k účtu obvykle přihlašujeme nejen za pomoci karty, ale také PIN kódu, čímž se riziko zneužití či odcizení finančních prostředků značně snižuje.

Nejmodernější, zejména tedy kriminalistická identifikace, probíhá prostřednictvím DNA. [2] Touto záležitostí se mimo jiné zabývá kriminalistická biologie. V DNA je ukryta informace o stavbě a struktuře našeho těla. Tento způsob identifikace osob se stále vyvíjí a zdokonaluje, proto je na velmi vysoké úrovni a domnívám se, že se jedná o velmi přesnou metodu.

Já osobně se zaměřím na určování identifikace na základě rozpoznání otisků prstů. Bylo dokázáno, že jsou pro každého z nás jedinečné a po celý život neměnné. [3] Na prstu každého z nás se nacházejí jiné papilární linie, tím pádem můžeme za jejich pomoci přesně určit, o čí otisk se jedná. Na rozdíl od takové identifikační karty nemůžeme svůj autentický otisk nikdy ztratit a odcizit jej je také velmi komplikované.

Cílem mé práce je vytvořit aplikaci pro počítač, která nahradí potřebu optického komparátoru. Má aplikace by měla splňovat řadu podmínek, vstupním zařízením by měl být optický snímač. Výstupem by měl být zvětšený obraz otisku prstu promítaný na jeden či dva monitory nebo projekční plátna. Tato zařízení mohou být různě kombinována a musí zde existovat možnost použít pouze jedno z nich. Aplikace tohoto typu by měla být využitelná v laboratoři, kde by měla nahradit klasické komparátory.

S podobnými aplikacemi se v praxi příliš často neseťkáváme, o důvodech můžeme polemizovat. Nejspíše je tento fakt způsoben skutečností, že zpracování otisků vyžaduje vysoký výpočetní výkon a také odborné znalosti v této oblasti.



## 2 Daktyloskopie

Daktyloskopie neboli dermatoglyfika[4] je nauka o kožních obrazcích, tedy tzv. papilárních liniích, které se vyskytují na článcích prstů rukou, dlaních, prstech nohou a chodidlech. Jedná se o jednu z nejstarších kriminalistických metod rozpoznávání osob. Jejich poznatků se využívá k odhalování, vyšetřování a prevenci trestných činů.[4]

*"V užším smyslu lze daktyloskopii chápat jako metodu určenou k identifikaci osob podle zvláštnosti kreseb jejich papilárních linií. V širším pojetí zahrnuje pojem daktyloskopie způsoby snímání otisků, jejich vyhledávání, zviditelňování, hodnocení, zkoumání jejich stáří i pravosti."*[5]

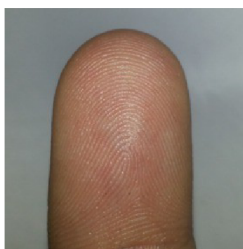
V minulosti se pojem daktyloskopie užíval jen pro studium otisků vytvořených vnitřní stranou posledního článku prstu. Ovšem vedle ní existovala také cheiroskopie, ta se zabývala stopami vytvořenými dlaněmi rukou, a pedoskopie, která zkoumala stopy chodidel a prstů nohou.

*„Daktyloskopie je charakterizována jako nauka o obrazcích papilárních linií vytvořených na vnitřní straně článků prstů, na dlaních, na prstech nohou a chodidlech.“* [6]Můžeme si tedy všimnout, že došlo k propojení těchto disciplín. Žádná jiná část lidského těla papilární linie neobsahuje.

Hlavním cílem daktyloskopie je identifikovat konkrétní osobu, která zanechala daktyloskopickou stopu a daktyloskopické srovnávací materiály. Jedná se jak o otisky podezřelých, tak o otisky osob, které se na místě pohybují. Může se jednat například o obyvatele bytu nebo osoby, které nemohou nebo nechťejí prokázat svou totožnost. Daktyloskopické stopy se třídí do sbírek a lze je zpětně využívat. [2] .

### 2.1 Papilární linie

Papilární linie můžeme charakterizovat jako vyvýšené reliéfy, které vznikají již během vývoje lidského plodu. Jejich výška se pohybuje mezi 0,1 a 0,4 mm a šířka mezi 0,2 a 0,5 mm. Takto vyvýšené reliéfy tvoří složité obrazce tzv. dermatoglyfy, které můžeme vidět na prstech, dlaních a ploskách nohou. Jejich charakteristické znaky nazýváme markanty.[4]



Obrázek 2.1: Papilární linie.

Papilární linie souvisejí s citlivostí a hmatovými vlastnostmi pokožky, nicméně jejich přesný účel není zcela znám. Víme, že částečně vznikají při soustavném uchopování předmětů i u osob kterým byla amputována končetina, tzv. Bartošův fenomén. [7]

Je nutno zmínit, že papilární linie jsou charakteristické pro celou třídu savců, nejen pro člověka, ovšem u některých se nacházejí pouze na posledních hmatových člancích prstů. Pouze u člověka a lidoopů tvoří složité obrazce, díky čemuž jsou stopy zvířat jednoduše rozpoznatelné.

## **2.2 Fyziologické zákony papilárních linií**

Francois Gallon, žijící mezi léty 1812 až 1911, byl anglický přírodovědec, který vytvořil fyziologické zákony, díky nimž můžeme prokázat individuálnost, neměnnost a neodstranitelnost obrazců papilárních linií. Tyto zákony studovalo mnoho vědců, snažili se je potvrdit, či naopak vyvrátit, ale vždy došli ke stejnému jednoznačnému závěru a jejich platnost definitivně potvrdili. Díky tomu se daktyloskopie stala součástí soudních důkazů sloužících k identifikaci osob. [4]

### **2.2.1 První zákon o neměnnosti obrazců papilárních linií**

V průběhu lidského života se papilární linie relativně nemění, přestože se pokožka vyvíjí, což pochopitelně ovlivňuje i papilární linie. Na jejich sled, skladbu, návaznost a relativní vzdálenost tyto změny však vliv nemají.

Vzhledem k tomu, že k tvorbě papilárních linií dochází už v prenatálním období vývoje dítěte, jsou tyto linie viditelné už při narození. Můžeme je zachytit dokonce i po smrti jedince, což usnadňuje jeho následnou identifikaci. [4]

### **2.2.2 Druhý zákon o relativní neodstranitelnosti papilárních linií**

Pokud nedojde ke zničení zárodečné vrstvy kůže, nelze papilární linie odstranit. V případě spálení, seřiznutí nebo chemického poškození povrchové vrstvy kůže, tyto linie nezničíme, což dokázali svými experimenty, na sobě samých, francouzští kriminalisté Locard a Wikovsky.

Na povrchu kůže se nachází tzv. rohová vrstva, jejímž poškozením k trvalým změnám papilárních linií nedojde. Obrazce linií se sice na krátký čas změň, ale po určité době, v závislosti na různých faktorech, jako je stupeň poškození, věk jedince, apod., se zase obnoví. Dokonce ani v případě osob pracujících například v chemickém průmyslu nebo ve stavebnictví, kde jsou zejména

jejich ruce, tedy i papilární linie na nich neustále ve styku s agresivními chemickými látkami nebo hrubými materiály, nedochází k poškození.

V případě skutečného porušení zárodečné vrstvy kůže k obnovení do původní podoby sice nedojde, ale vytvoří se různé jizvy. Ty svým způsobem papilární linie nahradí a stanou se charakteristické pro otisk dané osoby. Aby došlo k úplnému zániku, bylo by zapotřebí operativního zákroku, při němž by byla zárodečná vrstva pokožky odstraněna, ovšem výsledek by byl zcela viditelný, navíc by došlo ke vzniku jizev, jež by danou osobu jasně charakterizovaly.

Co se týče změny papilárních linií přirozeným způsobem, může k ní dojít, ale jen v případě osob postižených malomocenstvím či mozkomíšní mrtvicí ve stádiu krátce před smrtí. Ale vzhledem k předpokládané snížené pohyblivosti a neschopnosti spáchat trestný čin, nejsou tyto případy pro samotnou kriminalistiku nijak zásadní, tudíž tento zákon nevyvrací. [4]

### 2.2.3 Třetí zákon o individuálnosti papilárních linií

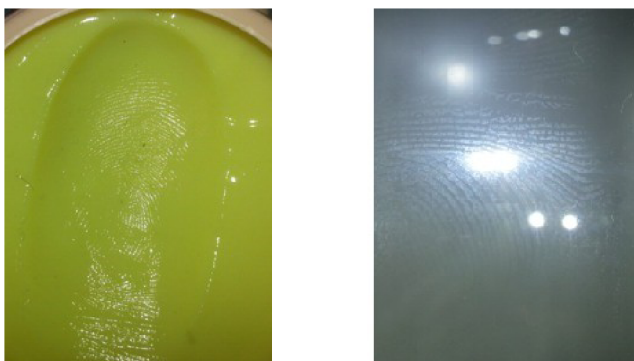
Bylo prokázáno, že na světě neexistují dva jedinci s naprosto totožnými obrazci papilárních linií. Vzhledem k velkému počtu markantů na jediném článku prstu, můžeme pravděpodobnost shodných obrazců papilárních linií označit za téměř mizivou. Italský vědec Balthazard odhadl pravděpodobnost jejich výskytu na  $1:10^{60}$ . [4]

Na rozdíl od analýzy DNA, díky studiu daktyloskopických stop od sebe dokážeme rozlišit dokonce i jednovaječná dvojčata. Přestože mohou být obrazy jejich papilárních linií podobné, nikdy nebudou úplně totožné. [4]

## 2.3 Daktyloskopické stopy

Daktyloskopickými stopami rozumíme vtisky, plastické stopy, nebo otisky, plošné stopy.

Otisky vznikají působením pokožky s papilárními liniemi na pevnou nedeformovatelnou plochu, zatímco vtisky vznikají působením na deformovatelnou plochu.



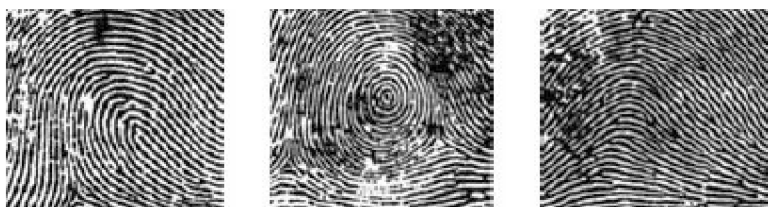
Obrázek 2.2: Daktyloskopické stopy vtisk a otisk.

Daktyloskopické stopy můžeme dělit podle viditelnosti na viditelné a skryté (latentní).[4][8] Co se týče výskytu, nejčastěji se setkáme s otisky skrytými (viz obr.2.2), které však okem nemůžeme za normálních podmínek vidět. Jak tyto stopy vzniknou? Latentní otisk je způsoben potem, který se nachází na lidské pokožce a při dotyku se na určitý předmět přenáší.

### 3 Základní popis otisku prstu

Každý obraz otisku prstu můžeme dělit na dvě části, a to na hřebeny a údolí.[4] Hřebeny představují papilární linie a pojmem údolí rozumíme mezery, jakési nevyplněné prostory, mezi nimi. Tyto části se pravidelně střídají asi po 0,5 milimetrech a vytvářejí strukturu otisku prstu.

Tu můžeme opět rozčlenit do pěti tříd[4]. Nalezneme zde oblouk, klenutý oblouk, vír, levou smyčku a pravou smyčku. Na jednom lidském prstu nalezneme pouze jednu z těchto tříd. V některých publikacích se však třídy hromadně označují jako singularity a body delta a jádro jako singulární body.[4]



Obrázek 3.1: Třídy papilárních linií smyčka, vír, oblouk [24].

Delta je definovaná jako místo, odkud se hřebeny rozbíhají do tří směrů. Jádro představuje střed otisku a definujeme jej jako bod na vrcholu nejvnitřnějšího zakřivení papilární linie.



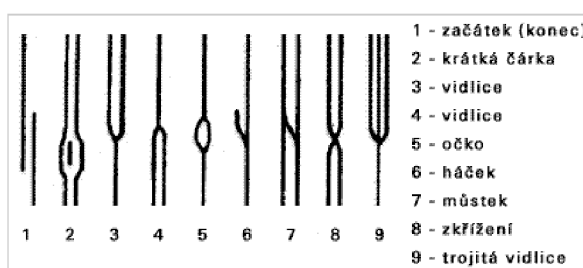
Obrázek 3.2: Delta a jádro.

Třídy můžeme popsat následovně. Obloukem rozumíme linii hřebenů vedoucích z jedné strany na druhou, přičemž nedochází k jejich zpětnému otočení. Obvykle zde není delta. Takový oblouk však nemá každý, nalezneme jej pouze na pěti procentech otisků všech lidí.

Některé linie hřebenu vytvářejí uzavřený okruh, ten se nazývá vír. Zmiňovaný vír nalezneme na dvaceti pěti až třiceti pěti procentech otisků. Pokud na otisku nalezneme právě tuto strukturu víru, obvykle taktéž objevíme dvě a více delt.[9]

Další částí otisku je takzvaná smyčka. Ta se skládá z jednoho či více hřebenu, které z určitého místa vstupují, následně se otáčejí zpět na stejnou stranu a buď se pomyslné čáry mezi deltou a jádrem dotýkají nebo jí zcela procházejí. Smyčky se dělí na radiální a ulnární podle jejich orientace. Tato třída otisků je nejčastější a spadá do ní šedesát až sedmdesát procent otisků. [9]

Otisky prstů dále dělíme podle charakteristických a významných znaků. Tyto znaky nazýváme markanty. Rozlišujeme několik základních typů, které jsou popsány na obrázku 3.3. Jedná se o začátek, krátkou čáru, vidlici, očko, háček, můstek, a zkřížení.



Obrázek 3.3: Základní markanty [25].

Při porovnávání se však nezaměřujeme pouze na markanty, ale i na počet linií mezi dvěma body. Nejčastěji zkoumáme linie, které protínají pomyslnou čáru mezi deltou a jádrem. [10]

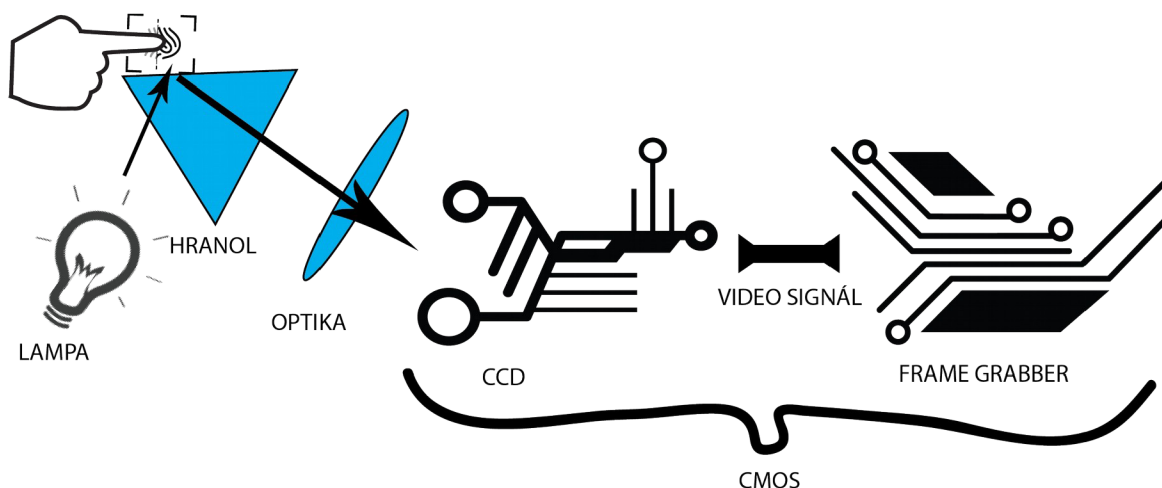
## 4 Snímání otisku prstu

Vzhledem k tomu, že potřebujeme každý otisk podrobně zkoumat, porovnávat a dále s ním pracovat, je nezbytně nutné jej převést do digitální podoby. Jednou z metod takového převedení je prosté nasnímaní inkoustového otisku scannerem. Toho se pořád využívá v kriminalistice při skenování daktyloskopických karet, avšak jedná se o zastaralé metody.

Dále můžeme ke snímání použít elektronické snímače. Nevýhoda této metody spočívá v nutné přítomnosti a následné spolupráci osoby, která otisk vlastní. Díky tomu se tato metoda nemůže uplatňovat v kriminalistice při zajišťování stop přímo na místě činu. V praxi využíváme dva typy těchto elektronických snímačů, a to kontaktní a bezkontaktní.

Jednotlivé kontaktní snímače jsou založeny na principech různých fyzikálních zákonů. Nejrozšířenějším snímačem je FTIR (Frustrated Total Internal Reflection) [11], ten využívá vlastnosti světla, které dopadá na rozhraní dvou látek. Mezi další typy patří snímače elektroluminiscenční, kapacitní, teplotní a další. Těmito druhy snímačů se však ve své práci dále nebudu zabývat.

Mezi bezkontaktní snímače patří ultrazvukový a optický. Optický snímač využívá vysokého rozlišení snímacího zařízení, například fotoaparátu, jehož prostřednictvím vznikne obraz v digitální podobě. Takto získaný obraz je poté nutno zpracovat, aby mohlo dojít k odstranění jeho negativních vlastností, mezi které patří například špatný kontrast mezi vrcholy a údolími. Mimo jiné je zde také nutné odstranit pozadí a nežádoucí šum. [12]



Obrázek 4.1: Schéma optického snímače.

## 5 Zpracování obrazu

Jelikož jsou možnosti dalšího zpracování otisků prstů závislé na kvalitě vstupních obrazů, je potřeba odstranit nežádoucí vlastnosti optického snímače a pokusit se kompenzovat některé chyby obrazu. Pokud je lokální orientace hřebenů stejná jako lokální orientace údolí a dochází-li k jejich pravidelnému střídání, mluvíme o ideálním obrazu otisku prstu.

Avšak takovýto obraz není ve skutečnosti snadné získat. Často dochází k poruše papilárních linií v důsledku ne zcela ideálního stavu pokožky. Můžeme na ní nalézt řadu přirozených defektů, mezi které řadíme jizvy, vrásky, popáleniny či oděrky, způsobené například manuální prací. Zajistit příkladný prst tedy není vůbec jednoduché a v praxi téměř nemožné.

Vznik nekvalitního obrazu však nemusí být zapříčiněn jen jakýmsi nedostatkem a nedokonalostí majitele prstu. Chyba může vzniknout také v případě nanesení přílišného množství inkoustu nebo silným přitlačením prstu. V důsledku těchto omylů může dojít ke spojení hřebenů, čímž se vzniklý obraz znehodnotí. Obraz často obsahuje několik oblastí s dobrou, střední a špatnou kvalitou.



Obrázek 5.1: Oblasti otisku prstů a) špatná kvalita, b) střední kvalita, c) dobrá kvalita

Obečně se rozlišuje několik typů poruch v obrazech papilárních linií. Může dojít například k situaci, kdy je hřeben na jednom či více místech přerušovaný nebo od sebe nemusí být rozlišitelné souběžné hřebeny. Dojde k jejich slití či k jejich spojení vlivem šumu. Důsledkem toho je špatná separace jednotlivých linií a možnost vzniku nebo zániku jednotlivých markantů. Poslední kategorií poruch jsou již zmiňované estetické či zdravotní postižení osob. Někdy je ale můžeme naopak považovat za tzv. další markanty a využít je k identifikaci daných osob.

Z velmi poškozených oblastí je získání struktury hřebenů velmi obtížné, proto se na otisky se špatnou kvalitou musí aplikovat algoritmy k jejich zlepšení. Tyto algoritmy jsou aplikovány pouze na obnovitelné oblasti, což jsou místa, kde jsou hřebeny poškozeny jen mírně, malými mezerami, vráskami, ale jsou stále viditelné a sousední hřebeny poskytují dostatečné informace k jejich rekonstrukci.

Nejčastěji využívanou metodou pro zkvalitnění obrazu papilárních linií je Gaborova filtrace [14], jež představuje druh algoritmizace, který zohledňuje lokální frekvenci a směr papilárních linií, podle kterých je aplikován vhodný filtr na obraz otisku prstu.

Gaborova filtrace spočívá v několika krocích. Nejprve uskutečníme takzvanou segmentaci, kdy oddělíme pozadí od popředí, následně určíme lokální orientaci papilárních linií, poté se zaměříme na jejich frekvenci a na závěr aplikujeme Gaborův filtr.

## 5.1 Segmentace

Je proces oddělení pozadí od popředí. Popředí je v tomto případě otisk prstu a pozadí je okolí, které neobsahuje informace o daném otisku, tzn. nadbytečné informace, které nepotřebujeme.

Obraz s otiskem prstu, převedený do úrovní šedi, se vyznačuje malým střídáním úrovní, kdežto samotný otisk má toto střídání o poznání větší. Díky tomu můžeme použít rovnici: [13][14]

$$V = \frac{1}{W^2} \sum_{i=0}^{W-1} \sum_{j=0}^{W-1} [I(i, j) - M]^2 \quad (5.1)$$

Kde  $V$  je výsledná úroveň,  $I(i, j)$  je intenzita v bodě  $i, j$   $W$  se nachází velikost bloku a  $M$  představuje střední hodnotu intenzity pixelů vycházející ze vztahu: [13][14]

$$M = \frac{1}{W^2} \sum_{i=0}^{W-1} \sum_{j=0}^{W-1} I(i, j) \quad (5.2)$$

Poté dojde k porovnání výsledku s globálním prahem a porovnáním zjistíme, zda se právě rozdělený obraz nachází v popředí či v pozadí.



Obrázek 5.2: Segmentace obrazu

## 5.2 Odhad lokální orientace papilárních linií

Odhad lokální orientace papilárních linií je proces vytváření orientovaného obrazu otisku prstu. V každém bodě papilární linie určíme odhadovaný úhel, který svírá tečna v daném bodě s horizontální osou. Tento úhel odhaduje směr dalšího průběhu papilární linie a v praxi se počítá pro malou oblast okolí bodu. Ve skutečnosti papilární linie nemají nijak definovaný směr. Díky tomu úhly nabývají pouze hodnot  $0^\circ$  až  $180^\circ$ .

Pro výpočet směrů je nejjednodušší použít gradient obrazu. „Gradient je v obecném smyslu slova směr růstu. Ve formálním jazyce matematiky označuje diferenciální operátor, jehož výsledkem je vektorové pole vyjadřující směr a velikost největší změny skalárního pole.“ [15]

Ve formálním zápisu se používá pro jeho označení  $\nabla$ . Gradient je tedy dvojrozměrný vektor  $[\nabla_x(x, y), \nabla_y(x, y)]$ , prvky  $\nabla_x$  a  $\nabla_y$  jsou gradienty ve směru  $x$  a  $y$ . Tyto gradienty, lze získat pomocí



Sobelova filtru [16], což je hranový detektor. Výsledný obraz gradientů získáme konvolucí Sobelovy masky o velikosti tři a obrazem otisku prstů. [17]

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (5.3)$$

Nejprve rozdělíme obraz otisku prstu na bloky  $W$  o velikosti  $w * w$  a za použití obrazu gradientů vypočítáme výslednou velikost úhlu  $\theta$  v daném bloku pomocí následujících rovnic: [13][14]

$$V_x(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2 \nabla_x(u, v) \nabla_y(u, v) \quad (5.4)$$

Díky přítomnosti šumu a různých chyb vstupního obrazu nemusí být takto získané směry papilárních linií správné. Ve skutečnosti dochází v papilárních liniích k pozvolným změnám svého průběhu, pokud se nejedná o místa, kde se vyskytují singulární body. Toho můžeme využít a nesprávně odhadnuté směry papilárních linií odstranit pomocí filtru dolní propust, ale nejprve musíme obraz orientací převést na spojité vektorové pole definované těmito vztahy: [13][14]

$$V_x(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} \nabla_x(u, v)^2 \nabla_y(u, v)^2 \quad (5.5)$$

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \left( \frac{V_x}{V_y} \right) \quad (5.6)$$

$$\Phi(i, j)_x = \cos(2\theta(i, j)) \quad (5.7)$$

$$\Phi(i, j)_y = \sin(2\theta(i, j)) \quad (5.8)$$

Kde  $\Phi_x$  a  $\Phi_y$  jsou komponenty vektorového pole. Ty dále použijeme ve filtru typu dolní propust  $W$  s velikostí jádra  $w \times w$ :

$$\Phi'_x(x, y) = \sum_{u=-\frac{w}{2}}^{+\frac{w}{2}} \sum_{v=-\frac{w}{2}}^{+\frac{w}{2}} W_\phi(u, j) \Phi_x(x-u, y-v) \quad (5.9)$$

$$\Phi'_y(x, y) = \sum_{u=-\frac{w}{2}}^{+\frac{w}{2}} \sum_{v=-\frac{w}{2}}^{+\frac{w}{2}} W_\phi(u, j) \Phi_y(x-u, y-v) \quad (5.10)$$

A následnou orientaci papilární linií vypočítáme pomocí arkus tangens jako:

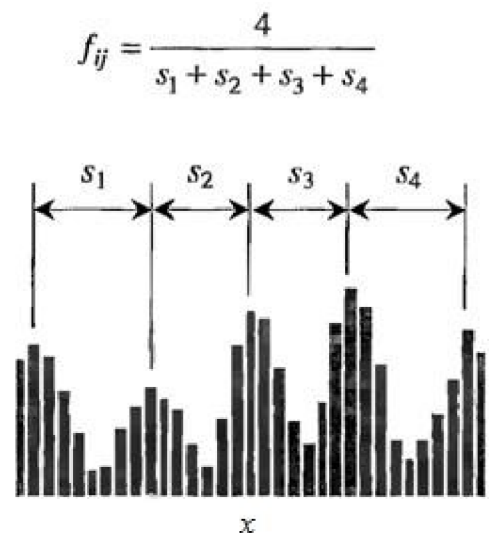
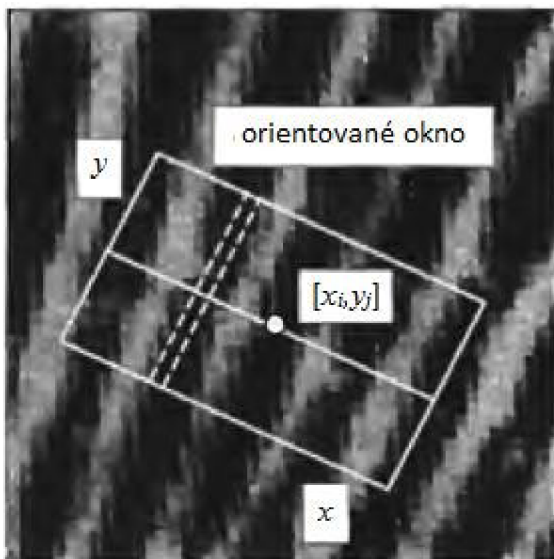
$$O(x, y) = \frac{1}{2} \tan^{-1} \left( \frac{\Phi'_y(x, y)}{\Phi'_x(x, y)} \right) \quad (5.11)$$



Obrázek 5.2: orientovaný obraz otisku prstu [23]

### 5.3 Odhad lokální frekvence papilárních linií

Lokální frekvence papilárních linií je další z důležitých vlastností obrazu otisku prstu. V šedotónovém obraze můžeme pravidelné střídání hřebenů a údolí, kolmých na lokální orientaci, popsat funkcí sinus s určitou frekvencí. Díky této vlastnosti poté můžeme odhadnout frekvenci papilárních linií. Frekvence se určí jako průměrný počet pixelů mezi lokálními maximy jejich intenzit.



Obrázek 5.3: Orientované okno [22]

K odhadu používáme normalizovaný obraz  $G$  a obraz orientací  $O$ . Normalizovaný obraz rozdělíme na bloky o velikosti  $w \times w$  a pro každý takto vytvořený blok použijeme tzv. orientované okno o velikosti  $l \times w$ , které je orientované kolmo na orientaci daného pixelů se středem  $(i, j)$ . V tomto okně vypočítáme x-signatury pomocí rovnice: [13][14]

$$X[k] = \frac{1}{w} \sum_{d=0}^{w-1} I(u, v), k=0, 1, \dots, l-1 \quad (5.12)$$

$$u = x + \left(d - \frac{w}{2}\right) \cos(O(x, y)) + \left(k - \frac{l}{2}\right) \sin(O(x, y)) \quad (5.13)$$

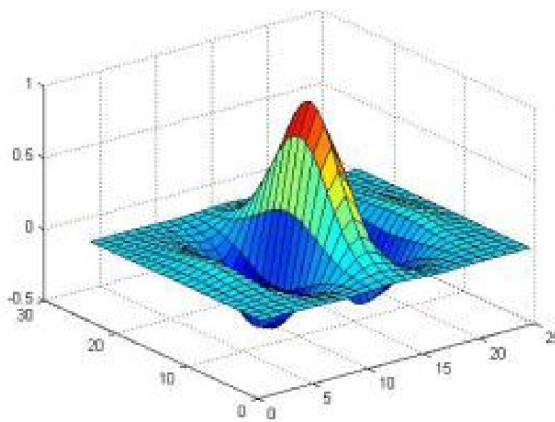
$$v = y + \left(d - \frac{w}{2}\right) \sin(O(x, y)) + \left(\frac{l}{2} - k\right) \cos(O(x, y)) \quad (5.14)$$

Jestliže se v orientovaném okně neobjeví singulární body, výsledkem je tvar diskrétní sinusoidy se stejným průběhem jako u hřebenů a údolí. Dále musíme vypočítat vzdálenost mezi dvěma prvky, což určuje periodu  $T$  a výsledná frekvence se vypočítá ze vztahu frekvence a periody:

$$F(x, y) = \frac{1}{T(x, y)} \quad (5.15)$$

## 5.4 Gaborova filtrace

Mezi jeden z nejčastějších algoritmů vylepšujících kvalitu otisku prstu patří Gaborova filtrace [13][14]. Tento filtr patří mezi pásmové filtry a díky svým frekvenčně a směrově selektivním vlastnostem je velmi vhodný k odstraňování šumu a ke kompenzaci chyb i v nekvalitním obraze. Gaborův filtr se zakládá na matematickém modelu otisku prstu, to znamená, že se sinusový tvar hřebenů pomalu mění ve směru lokální orientace. [13][14]



Obrázek 5.4: Gaborův filtr [22].

Matematický popis Gaborova filtru je obecně definován takto:[13][14]

$$G(x, y, \varphi, f) = \exp \left\{ \frac{-1}{2} \left[ \frac{(x_\varphi)^2}{(\delta_x)^2} + \frac{(y_\varphi)^2}{(\delta_y)^2} \right] \cos(2\pi f x_\varphi) \right\} \quad (5.16)$$

$$x_\varphi = x \cos(\varphi) + y \sin(\varphi) \quad (5.17)$$

$$y_\varphi = -x \sin(\varphi) + y \cos(\varphi) \quad (5.18)$$

Kde  $\varphi$  je orientace Gaborova filtru,  $f$  frekvence sinusové vlny a  $\delta_x$ ,  $\delta_y$  jsou prostorové konstanty gaussovské obálky podél os  $x$  a  $y$ . Jako frekvenci můžeme dosadit lokální frekvenci, za orientaci lokální orientaci a konstanty jsou volené kompromisem.

V případě, že jsou konstanty příliš veliké, může dojít ke ztrátám nebo vytváření nových hřebenů. Takto zvolené konstanty jsou zase velice odolné vůči šumu. Na druhou stranu při zvolení malých konstant není filtr dostatečně efektivní v jejich odstraňování.

## 6 Ztenčování papilárních linií

Ztenčení je důležitou součástí zpracování otisků prstů. Jedná se o ztenčení papilárních linií na jeden pixel bez jejich přerušení. Většina algoritmů pracuje na principu opakovaného mazání pixelů. V binarizovaném obraze ověřujeme, zda smazání každého pixelu nenaruší spojitost. V případě, že tomu tak není, může být prvek smazán. Takto postupujeme, dokud hrana nedosahuje šířky jednoho pixelu. K tzv. Ztenčování můžeme užít několik metod. [18][19]

### 6.1 Zhaung-Suen tenčící algoritmus

Předpokládejme že, černé pixely mají hodnotu jedna a bílé nula, vstupním obrazem je obdélník o velikosti  $N \times M$  obsahující právě tyto hodnoty. Algoritmus Zhaung – Suen [26] pracuje nad černými pixely  $P_1$ , které mají osm sousedů, kvůli tomu neprochází hraniční pixely obrazu. Pořadí sousedů je následující:[20]

$P_9$	$P_2$	$P_3$
$P_8$	$P_1$	$P_4$
$P_7$	$P_6$	$P_5$

Definujeme  $A(P_1)$  jako počet přechodů černé a bílé v pořadí  $P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9$ .

Definujeme  $B(P_1)$  jako množství černých pixelů v okolí bodu  $P_1$ .

## Krok 1

Každý pixel je testován, zda splňuje následující podmínky:

1. černý pixel má osmi okolí
2.  $2 \leq B(P_1) \leq 6$
3.  $A(P_1) = 1$
4. Alespoň jeden z pixelů  $P_2$ ,  $P_4$  a  $P_6$  je bílý
5. Alespoň jeden z pixelů  $P_4$ ,  $P_6$  a  $P_8$  je bílý

Po otestování každého pixelu v obraze je každý pixel, který splňuje všechny podmínky z kroku 1 nastaven na nulu.

## Krok 2

Znovu se každý pixel testuje na následující podmínky:

1. černý pixel má osmi-okolí
2.  $2 \leq B(P_1) \leq 6$
3.  $A(P_1) = 1$
4. Alespoň jeden z pixelů  $P_2$ ,  $P_4$  a  $P_8$  je bílý
5. Alespoň jeden z pixelů  $P_2$ ,  $P_6$  a  $P_8$  je bílý

Znovu jsou všechny pixely, které splňují všechny podmínky z kroku dva nastaveny na nulu.



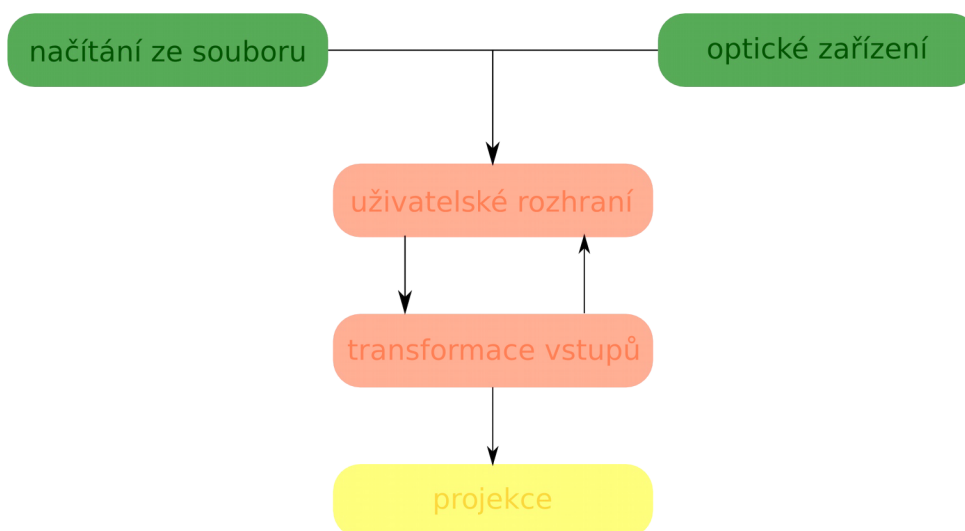
Obrázek 6.1: Zobrazené papilární linie na tloušťku 1 pixelu.

# 7 Návrh aplikace

Následující kapitola se zabývá návrhem aplikace, která má sloužit jako forenzní optický komparátor se vstupem, jímž je soubor nebo optický snímač. Nejprve provedeme rozbor problému v kapitole 7.1, dále se budeme věnovat použitým knihovnám 7.2 a nakonec popisu grafického rozhraní 7.3.

## 7.1 Rozbor problému

Cílem je vytvořit aplikaci, která nahradí potřebu forenzního optického komparátoru se zaměřením na otisky prstů. Tato aplikace musí využívat optické snímače a promítat otisk na více zobrazovacích zařízeních.



Obrázek 7.1: Schéma navrhované aplikace. Zeleně jsou označena vstupní data aplikace, která zadává uživatel. Žlutě je znázorněn výstup a červená barva představuje programové části aplikace.

Podle tohoto schématu můžeme aplikaci rozdělit na tři části, a to na sekce: vstupní, zpracovávání a zobrazování.

Vstupními daty rozumíme buď načtení obrazu otisku prstu ze souboru nebo snímek z optického zařízení. Takto získaný obraz můžeme v grafickém rozhraní upravovat a následně uložit do souboru. Vstupním souborem je obrázek ve formátu podporovaném knihovnou openCV viz. Kapitola 7.2.2.

Aby bylo možné s každým otiskem odděleně dále pracovat, otevírá se každý v novém okně. Takto zobrazený otisk prstu je možné buď umístit na jakoukoli pozici na projekčním zařízení nebo je zde možnost umístit zobrazované okno pomocí volby v menu viz. kapitola 8.3.

## 7.2 Použité knihovny

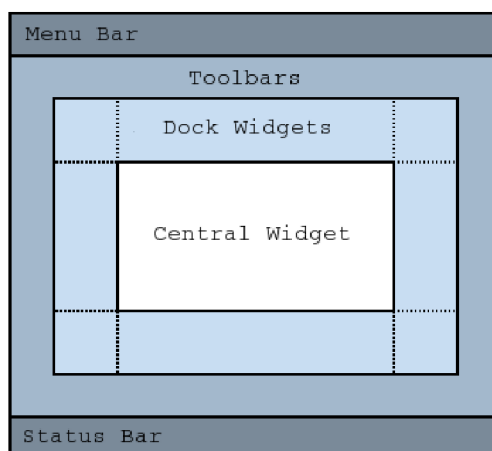
Zde popisují vybrané části knihovny QT<sup>1</sup> 6.2.1 a OpenCV<sup>2</sup> 6.2.2 využité při implementaci.

### 7.2.1 Knihovna QT

Qt patří mezi nejpopulárnější multiplatformní knihovny pro tvorbu grafického uživatelského rozhraní. Přestože tato knihovna existuje pro různé programovací jazyky, v praxi se ve valné většině případů využívá pouze jazyk C++. [21]

Jakýsi základní stavební prvek tvoří třída `QObject`. Všechny objekty v QT tuto třídu dědí, díky čemuž obsahují hlavní výhodu této knihovny. Onou výhodou sledujeme systém komunikace mezi objekty signál a slot.[21]

Následující třídou použitou v aplikaci je `QMainWindow`. Tato třída poskytuje základní rozložení okna. Lze zde přidat a nastavit menu bar, status bar a central widget, který slouží jako centrální oblast pro přidávání objektů. V našem případě je do této oblasti přidáno několik widgetů a `QGraphicsView`, ta slouží jako zobrazovací plocha otisku prstu. Nicméně hlavní funkcí této třídy je vytvoření a propojení jednotlivých tříd nejen mezi sebou, ale také jejich propojení s grafickým uživatelským rozhraním.[21]



Obrázek 7.2: Schéma třídy `QMainWindow`[21]

`QGraphicsView` poskytuje objekt pro práci a zobrazování scény. Jedná se tedy o obyčejný widget, který si můžeme představit jako plátno, na které se přidávají jednotlivé prvky, které jsou vektorově vykreslované. To znamená, že prvky jsou zadávány pomocí parametrů. Této vlastnosti se používá pro vykreslování ztenčených takových papilárních linií polygonů a pro označování markantů na nich.[21]

<sup>1</sup>Qt Documentation [online]. [cit. 2015-05-10]. Dostupné z: <http://doc.qt.io/>

<sup>2</sup>Opencv. [online]. [cit. 2015-01-28]. Dostupné z: <http://opencv.org/>

## 7.2.2 Knihovna openCV

OpenCV (Open Source Computer Vision Library) je svobodná a otevřená multiplatformní knihovna sloužící ke zpracování obrazu a k počítačovému vidění. V této aplikaci se bude využívat rozhraní pro jazyk C++, ale obsahuje i rozhraní pro jazyky C, Python, Java a Matlab.[17]

Pro přenos veškerých informací o obraze se používá třída `Mat`, která obsahuje dvě části, hlavičku matice a ukazatel na hodnoty pixelů. V hlavičce jsou uloženy informace o velikosti, metodě uložení, adresa v paměti atd. Tato třída se používá jako vstupní a výstupní parametry funkcí na filtrování a úpravu obrazu.[17]

Jelikož vstupními daty aplikace jsou soubory s obrazy, je na jejich načítání používána funkce `imread`. Mezi vstupy, které podporuje, patří:[17]

- JPEG files - `*.jpeg`, `*.jpg`, `*.jpe`
- JPEG 2000 files - `*.jp2`
- Portable Network Graphics - `*.png`
- Portable image format - `*.pbm`, `*.pgm`, `*.ppm`
- Sun rasters - `*.sr`, `*.ras`
- TIFF files - `*.tiff`, `*.tif`
- Windows bitmaps - `*.bmp`, `*.dib`

A pro vstup z web kamery je v OpenCV připravena třída `VideoCapture`.

## 7.3 Popis grafického rozhraní

Návrh aplikace počítá s grafickým rozhráním, kvůli možnosti zobrazování a zpracování obrazu otisku prstu. Grafické rozhraní by mělo obsahovat jednoduché menu obsahující jednotlivé funkce na zpracování obrazu. Mezi funkce by mělo patřit ořezání, protože ne vždy je potřeba velký obraz, zmenšení velikosti, Gaborova filtrace, segmentace, normalizace a návrat k původnímu obrazu. Dále je potřeba zobrazovací plocha, na níž bude možnost pomocí kolečka myši přiblížit nebo oddálit obraz. Nebude chybět ani možnost označení markantů a významných bodů.

Horní menu bude obsahovat schopnost vytvořit nové okno, pro zobrazení dalšího otisku, načtení souboru, uložení zobrazovací plochy a zavření okna.



## 8 Implementace

V této kapitole budou zmíněny implementační nástroje a popis implementovaných tříd podle návrhu v kapitole 7. Také zde budou podrobně rozebrány implementované třídy `OtiskWindow` 8. , `PraveMenu` , `Screen` , `Snimani` , `ImageCV` , `OpenCV` , `Menu`.

Třídy `PraveMenu` a `OpenCV` ve schématu odpovídají transformaci vstupů. Snímání odpovídá optickému zařízení a `ImageCV` odpovídá projekci. Ostatní třídy jsou využívány zejména jako pomocné třídy sloužící např. na nastavení polohy okna na monitoru.

### 8.1 Použité implementační nástroje

Celá aplikace byla implementovaná v jazyce C++ s využitím knihovny `OpenCV` ve verzi 2.4.11. Nejvíce využívaná třída je `cv::Mat`, která uchovává údaje o zpracovávaném obraze, mezi další využívané třídy řadíme: pro filtrování obrazu `cv::getGaborKernel`, `cv::filter2D` a pro prahování pomocí funkce `cv::threshold`, dále je to `cv::VideoCapture`, určená pro přístup k optickému zařízení (web kamera).

Grafické uživatelské rozhraní vytváříme s pomocí knihovny `QT`. Celá aplikace vznikla a byla testovaná na platformě `PC` s operačním systémem `Ubuntu 14.10`.

### 8.2 Třída *OtiskWindow*

Jedná se o základní třídu celé aplikace, která dědí `QMainWindow`. Je zde implementováno propojení jednotlivých tříd mezi sebou a také s uživatelským rozhraním. `OtiskWindow` vytváří menu jako `menuBar` okna, připojuje `widgedy` pro ovládání a zobrazování obrazu. Všechny objekty se nacházejí v rozložení této třídy.

Podle návrhu je v `menuBar` umístěna jedna položka `file`, která obsahuje funkci na vyvolání nového okna, kdy dojde k vytvoření nové instance této třídy. Po stisknutí tlačítka `load` se vyvolá dialog, ze kterého získáme jméno načítaného souboru, a pomocí již zmiňované funkce `imread` nahrajeme obraz jako matici. Funkce `save` se podobná funkci `load`, znovu je vyvolán dialog, pomocí jehož se vybere umístění a název souboru, následně se projekční plocha převádí na objekt `QImage` a uložena. Důvodem převádění plochy na objekt knihovny `QT` a nepoužití třídy `Mat` je kreslení do projekční plochy.

Je zde také umístěna logika zasouvacího `PravehoMenu`, což je vytvořeno pomocí animací a provázáno se zobrazovací plochou. Pokud je vzdálenost pozice myši v zobrazovací ploše a jejího levého okraje menší než dvacet pixelů je menu zobrazeno a projekční plocha je o jeho šířku zúžena. Vysouvací menu jsem zvolil kvůli zvětšení zobrazovací plochy a také z důvodu rušivého efektu při stálém zobrazení.

V této třídě se nachází nastavení pozic jednotlivých objektů a metodou `resize` je nastavována jejich velikost při změně.

## 8.3 Třída *PraveMenu*

Třída `PraveMenu` vznikla k účelu ovládání. Jedná se o `QWidget` od kterého dědí, a její vytvoření je ve třídě `OtiskWindow`. Jako vstupní parametry pro vytvoření jsou použity zobrazovací plocha a obrazovka.

Zobrazovací plocha zde slouží k provázání s ovládacími prvky aplikace. Toho je docíleno pomocí signálů a slotů. Nejvyužívanějším signálem sledujeme `clicked`, který je pro každý ovládací prvek spojen se slotem v zobrazovací ploše.

Mezi jeden z ovládacích prvků patří snímání a jeho zastavení. Tato tlačítka slouží ke spuštění a zastavení vstupu z optického zařízení. Ovládací prvky jsou zobrazeny na stejné pozici a pomocí módu `visible` zpřístupňovány. K přepínání dochází signálem `stopEnable`, který je generován po vybrání optického snímače.

V této třídě se také vytváří ovládací prvek na nastavení pozice okna. Po stisku je zobrazeno menu, jehož položkami jsou dostupné zobrazovací plochy. Po výběru jedné z nich je zpřístupněno další menu s jednotlivými módy pro pozici okna. Jedná se o zobrazení v levé půlce, pravé půlce, maximalizovaného okna, okna přes celou plochu a normálního okna. Okno přes celou plochu a maximalizované okno se od sebe liší jen nepatrně. Hlavní rozdíl se objeví při zobrazení okna přes celou obrazovku, kdy se menu systému překryje. Nicméně v tomto módu se menu pro otevírání, ukládání nezobrazuje.

Dále jsou zde přítomny ovládací prvky pro práci s obrazem otisku prstu. Jedná se o Gaborovu filtraci popsanou v kapitole 5.4, tenčení papilárních linií pomocí algoritmu popsaného v kapitole 6.1, zobrazení původního obrazu otisku prstu, invertování obrazu, což je důležité především v případě bílého otisku na černém pozadí, označení všech ukončení v obraze ztenčených papilárních linií, povolení označování významných bodů, segmentace popsaná v kapitole 5.1, normalizace obrazu, zmenšení obrazu a jeho ořezání.

## 8.4 Třída *ImageCV*

*ImageCV* je v aplikaci využívána jako plocha pro zobrazování otisku prstu. Ke kreslení zde slouží objekty `QgraphicsView` a `QgraphicsScene`. Nejprve je nastaven obraz v této třídě jako matice z knihovny `openCv`, která je převedena na objekt `Qimage` a následně přidána a vykreslena do objektu `QgraphicsScene`. Při tomto vykreslení je nastaven widget, který zobrazuje rozměry obrazu otisku prstu a pozici myši vzhledem k pozici ve scéně. Widget má nastavenou pozici v pravém dolním rohu scény a využívá se pro informování při změně velikosti obrazu otisku prstu, což je důležité pro Gaborovu filtraci.

Tato filtrace je náchylná na velikost matice obrazu, při pozorování jsem zjistil, že filtrace je nejlépejší při velikosti kolem  $500 \times 500$  pixelů. Příliš velká matice způsobuje, že ve výpočtovém okně o velikosti  $16 \times 16$  pixelů se nemusí vyskytovat papilární linie. To má za následek vznik nových papilárních linií a šumu. Ve filtraci se toto výpočtové okno používá pro výpočet lokálních orientací a lokálních frekvencí papilárních linií. Tato obálka je zvolena kompromisem na hodnotu 0,5. Při větších velikostech obrazu může dojít k větším rozdílům mezi papilárními liniemi. Tzn. vzdálenost mezi papilárními liniemi je větší než již zmíněných šestnáct pixelů, což má za následek, že ve výpočtovém okně se nevyskytuje papilární linie a že dojde k nepřesnému určení úhlu a frekvence v tomto okně.

V této třídě existují dva módy pro vykreslování. První mód vykresluje samostatnou matici pixelů jako objekt knihovny Qt `Qimage`, který je následně roztažený nebo zmenšený na velikost zobrazovací scény. Nicméně je zde potřeba kreslit do obrazu ztenčených linií. Kvůli zpřesnění označování významných bodů v otisku prstu jsem zvolil jejich vykreslování pomocí polygonů. Označované body se mohou vyskytovat pouze na polygonu ztenčených papilárních linií. Samozřejmě jsou ztenčené papilární linie znovu roztažené do prostoru celé scény se zachováním jejich poměrů.

Při pohybu myši ve kreslicí scéně, jsou generovány události myši `wheelEvent`, `mouseMoveEvent`, `mousePressEvent` a `mouseReleaseEvent`. `WheelEvent` slouží k přibližování a oddalování scény s měřítkem 1:1.1.

Při události `mouseMoveEvent` je nastavena pozice myši ve widgetu v pravém dolním rohu, ale také jsou zde emitovány signály pro animaci pravého menu. Jedná se o signály pro spuštění a její zastavení. Jelikož objekt `QpropertyAnimation` neobsahuje metodu pro restartování animace, jsou při vzdálenosti myši a levého okraje emitovány signály pro stop start a stop, nicméně jsou tyto signály emitované jen v případě celkově nebo částečně zobrazeného pravého menu.

Událost `mousePressEvent` se využívá pro již zmíněné označování významných bodů. Na pozici myši ve scéně je hledán objekt vykresleného polygonu a pokud je nalezen je, bod se označí.

Pokud je nalezen již označený bod, je označení odebráno. Je zde také uložena pozice myši, které se využívá pro ořezání obrazu v události `mouseReleaseEvent`.

`MouseReleaseEvent` v této události je programovaná logika ořezávání obrazu. V případě ořezávání dojde k výpočtu obdelníku. Výpočet obdelníku se provádí na základě znalosti počátečního a koncového bodu. Na základě těchto proměných určíme levý horní a pravý dolní bod výsledného obrazu. Obdelník se také musí transformovat do velikosti scény a tímto obdelníkem je provedeno ořezání matice, která je podle vykreslovacího módu zobrazena.

Jsou zde také obsaženy sloty k provázání ovládacích prvků pravého menu. Ve slotech je implementována celá logika úpravy obrazu. Ze slotů obsažených v této třídě jsou pouze dva, které slouží k přepínání mezi stavy povolení označování významných bodů a ořezávání. Ve zbylých slotech se nachází pouze volání metod třídy `OpenCv`.

## 8.5 Třída *OpenCv*

Ve třídě `OpenCv` se provádí veškerý výpočet na úpravu obrazu otisku prstu. Všechny metody v této třídě jsou statické a hojně se zde využívá knihovna `OpenCV` popsaná v kapitole 7.2.2. Je zde prováděna Gaborova filtrace na matici, segmentace, tenčení, invertování hodnot matice, rozpoznávání ukončení v papilárních liniích, normalizace a zmenšení matice.

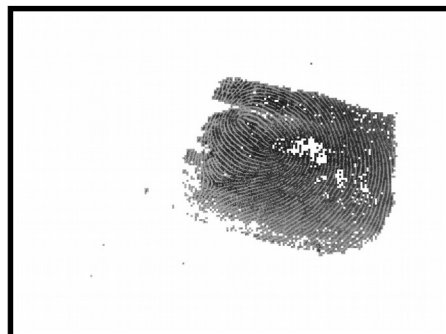
Pro Gaborovu filtraci je nejprve spočítána lokální orientace papilárních linií v každém bodě matice, podle rovnic 5.3 – 5.11 z kapitoly 5.2., výsledná matice je poté upravena pomocí filtru dolní propusti. Filtr dolní propusti spočívá ve vytvoření jádra filtru jako matice  $3 \times 3$  s hodnot  $1/9$  a konvolucí této masky s maticí lokálních orientací. Poté je spočítána frekvence papilárních linií pomocí rovnic 5.12 – 5.15 z kapitoly 5.3. Tato frekvence je počítána vždy pro středový prvek výpočtového okna  $16 \times 16$  pixelu. Takto vypočítané matice lokálních orientací a lokálních frekvencí papilárních linií slouží jako vstup do metody pro výpočet Gaborovy filtrace. Nejprve je obraz filtrovaný Gaussovým filtrem, který odstraní nežádoucí šum, poté je v této metodě rozdělen původní obraz otisku prstu na bloky o velikosti výpočtového okna a na každý blok je aplikována Gaborova maska, která je vypočtena ve funkci `cv::getGaborKernel` z knihovny `OpenCV`. Vstupním parametrem funkce je  $\sigma$ , neboli směrodatná odchylka Gaussovy obálky. Další je  $\theta$ , což je lokální orientace papilárních linií, neboli natočení Gaborova filtru, a poslední  $\lambda$ , určující lokální frekvenci papilárních linií, neboli frekvenční vlastnost Gaborova filtru. Nicméně po aplikaci Gaborovy masky jsou zvýrazněné i hranice každého bloku. Ve výsledném obrazu se tak vytvoří

mřížia. Z tohoto důvodu dojde k odstranění veškerých pixelů na hranicích každého bloku, kromě těch které mají na pozici o jedna větší než je hranice bloku nenulový pixel. Tímto dojde k odstranění nežádoucí mřížky. Jelikož výsledný filtrovaný obraz trpí spoustou přerušení a šumem v obraze papilárních linií, je tento obraz filtrován pomocí mediánového filtru. Tento filtr odstraní šum. Pro přerušení papilárních linií jsem se snažil využít Gaussův filtr, který dokáže nahradit rozpojené papilární linie, nicméně takto filtrovaný obraz je náchylný ke spojování i souběžných papilárních linií. Po binarizaci takového obrazu nelze jasně určit práh, aby došlo k potlačení této vlastnosti Proto není Gaussův filtr pro spojování papilárních linií v mé práci použit. Avšak existuje zde možnost filtr v budoucnu použít a pokusit se nežádoucí vlastnosti potlačit.



Obrázek 8.1: Obraz otisku prstu před a po aplikování Gaborova filtru.

Segmentace je zde implementována pomocí rovnic z kapitoly 5.1, nejprve tedy obraz rozdělíme na bloky o již zmíněné velikosti  $16 \times 16$  pixelů a v každém takto vytvořeném bloku vypočítáme střední hodnotu, z čehož vyvodíme hodnotu střídání úrovní. Jako kompromis jsem zvolil hodnotu sto. Pokud je tato hodnota větší než vypočtená hodnota, je celý blok označen jako pozadí a všechny jeho pixely jsou nastaveny na hodnotu 0. Bohužel i tento algoritmus má nedostatky pro obraz s velkým rozlišením. V takovém obraze mohou být mylně označeny i bloky s velice širokými nebo mírně slitými papilárními liniemi.



Obrázek 8.2 Segmentovaný obraz otisku prstu.

Tenčení je prováděno na základě algoritmu z kapitoly 6.1. Je vypočtena matice ztenčených linií a v této matici jsou pomocí funkce `findContours` nalezeny všechny obrysy objektů, což odpovídá polygonům, které jsou poté vykreslovány. Tyto obrysy jsou dále využívány k hledání vrcholů polygonů, neboli ukončení papilárních linií. To se provádí pomocí procházení jednotlivých bodů polygonu.

Mějme bod  $i$  jako bod polygonu, pokud bod  $i+2$  je roven bodu  $i$ , tak je bod  $i+1$  označen jako vrchol polygonu. To vychází z úvahy, že vrchol v obrysu papilární linie o velikosti jeden pixel má pozici bodu o jedna menší a pozici bodu o jedna větší stejnou. To znamená, že při dosažení vrcholu je směr linie otočen a vrací se po stejných bodech.



Obrázek 8.3: Tenčený otisk prstu a označení ukončení.

Dále jsou zde pomocné metody pro úpravy obrazu. Jedná se o invertování hodnot matice, což spočívá ve vytvoření stejné matice s hodnotou každého pixelu 255 a odečtení původní matice. Pro zmenšování je použita funkce `resize` a pro obě osy je zvoleno měřítko 0,8. Stejně tak pro normalizaci je použita funkce `normalize`. Všechny zmíněné funkce jsou z knihovny `OpenCV`.

## 8.6 Třída *Screen*

Tato třída ovládá a upravuje pozici okna na zobrazovací ploše např. monitoru, projektoru, vestavěného displeje notebooku aj. Je zde obsažena metoda, která vrací počet všech dostupných zobrazovacích ploch. To informace je získána přímo ze třídy `QApplication` voláním metody `desktop`. Nicméně je zde potřeba rozlišovat zobrazovací režimy systému. Hlavně pak duplikaci

plochy nebo jejím rozšířením, proto je zde určována vždy pozice levého horního rohu a pokud má některá další zobrazovací plocha stejný tento bod, je také plocha duplikací již přidané plochy.

Dále jsou zde obsaženy metody, které okno přesunou na vybranou zobrazovací plochu a podle zvolené metody nastaví jeho pozici a jeho rozměry. Pozice nastavovaného okna představují buď střed plochy levý horní okraj, nebo střed horního okraje. Jeho velikost se nastaví na půlku plochy, celou plochu nebo minimální velikost okna. Po tomto nastavení jsou emitovány signály na restartování pravého menu.

## 8.7 Třída *Snimani*

Zde je implementována logika k ovládní optického zařízení. Nejprve je adresář `/sys/video4linux/` prohledán a pro každé zde uložené zařízení je získáno jeho jméno. Jedná se o jména zobrazovaná v seznamu dostupných optických zařízení.

Pro přístup k jednotlivým optickým zařízením je používána třída `VideoCapture` z knihovny `OpenCV`. Toto otevřené zařízení je spojeno s časovačem a každých 33 milisekund je snímán obraz z optického zařízení.

## 8.8 Třídy *HorniMenu* a *Menu*

Třída horní menu slouží pouze k zobrazování jména aktuálně otevřeného souboru s obrazem otisku prstu, pakliže není žádný soubor otevřen je zobrazen nápis nový.

Třída `Menu` zde slouží jako pomocná třída k předávání a uložení indexu v menu pro přístup k seznamu optických zařízení a zobrazovacích ploch. V tomto menu je důležité si uchovávat a následně odesílat číselnou hodnotu, která slouží jako index v přístupu k optickému zařízení nebo zobrazovací ploše. Je zde také generován signál pro přepínání ovládacích prvků.

# 9 Budoucí úpravy a nedostatky

V této kapitole jsou shrnuty zjištěné nedostatky současné aplikace. U některých nedostatků je navrženo řešení pro případnou budoucí úpravu aplikace. Dále zde navrhuji možná rozšíření a nakonec popisují jednotlivé ovládací prvky aplikace.

## 9.1 Zjištěné nedostatky

Základním a hlavním problémem aplikace je kvalita otisku prstu. Při velmi nekvalitním obrazu nelze jednotlivé papilární linie rozlišit a kvůli tomu nelze s takovým obrazem automaticky pracovat.

Dalším nedostatkem aplikace, sledávám vysoké nebo naopak příliš malé rozlišení obrazu. To může mít za následek špatné rozlišení jednotlivých papilárních linií a při používání filtrů na vylepšení obrazu může dojít k vytváření nových papilárních linií nebo jen jejich spojení. Rozlišení by se dalo omezit na jen přípustnou velikost a každý vstupní obraz by se na tuto velikost upravil.

Obrazy s velkým rozlišením jsou také náchylné na vytváření šumu a pomalé zpracovávání. Rychlejšího zpracování by bylo možné dosáhnout použitím jiné knihovny např. OpenGL namísto OpenCv, nebo by mohla být část výpočtů zpracovávána na grafické kartě.



Obrázek 9.1: Filtrovaný obraz ve vysokém rozlišení.

Při zkvalitňování obrazu otisku prstu dojde vlivem použití Gaborova filtru jen na bloky obrazu k mírnému posunutí těchto bloků a při opakovaném použití této filtrace se tato chyba kumuluje a výsledek je poté velmi nepřesný. V tomto případě dojde i k filtraci hran bloku a k vytvoření mřížky mezi bloky ve výsledném obratu. Tohle odstraníme tak, aby byla zachována spojitost papilárních linií tzn. Jen některé pixely na hranici bloků jsou nahrazeny. I nekvalitní obraz otisku prstu může být vylepšen, nicméně dochází k častému přerušování papilárních linií. Vypozoroval, že tento jev lze odstranit pomocí Gaussova filtru. Bohužel tento filtr rozmazává i hrany papilárních linií, tudíž dochází k jejich spojování.

## 9.2 Možné rozšíření aplikace

I když vytvořená aplikace splňuje zadání této práce, pořád zde zůstává možnost jejího rozšíření. Rozšíření mohou být prováděna na v celém zdrojovém kódu aplikace. Největší možnost pro



rozšíření vidím v rozpoznávání markantů, nicméně by se musel vyřešit problém s častým přerušováním papilárních linií. Pro rozpoznávání markantů se nejčastěji používá Poincarého index.

Dalším vhodným rozšířením by byla možnost přidání databáze otisku prstu a následné porovnávání. Existují i hardwarové rozšíření. Jednalo by se především o vytvoření optického zařízení, které by mělo pevně nastavenou vzdálenost od plochy snímání a také jeho doostření. Takové zařízení by zkvalitnilo snímání otisků a tím pádem i výsledek aplikace. Samozřejmě by zde bylo vhodné i použití různých snímačů otisků prstu. Při přidání takového zařízení by bylo možné aplikaci použít i přímo na snímání otisku přímo z prstu.

### 9.3 Popis ovládacích prvků



Obrázek 9.2: Zapnutí snímání. Implementována jako menu pro výběr optického zařízení.



Obrázek 9.3: Zastavení snímání.



Obrázek 9.4: Výběr zobrazovacího zařízení. Implementována jako menu.



Obrázek 9.5: Spuštění Gaborovy filtrace.



Obrázek 9.6: Ztenčení všech objektů v obraze na šířku jednoho pixelu.



Obrázek 9.7: Návratu k původnímu obrazu.



Obrázek 9.8: Invertování obrazu.



Obrázek 9.9: Vyznačení všech ukončení papilárních linií.



*Obrázek 9.10: Segmentace obrazu neboli odstranění pozadí.*



*Obrázek 9.11: Normalizace obrazu.*



*Obrázek 9.12: Zmenšení obrazu.*



*Obrázek 9.13: Ořezání obrazu.*

Všechny ikony byly kresleny nebo byly využity dostupné vektorové banky. Pouze ikona pro Gaborovu filtraci byla vytvořena zmenšením obrázku Gaborova filtru a následnou úpravou k zachování podobnosti s ostatními ikonami.

## 10 Závěr

Ve své práci jsem se snažil vytvořit aplikaci, která by nahradila optický komparátor otisků prstů. Dá se říci, že tato aplikace zmíněné požadavky splňuje. Snímá obrázek nebo využije již dříve získaný obrázek ze souboru. Obrázek dále zpracovává za použití Gaborovy filtrace, ztenčuje papilární linie a rozpoznává jejich ukončení. V obrázku je možno označovat významné body. Tímto jsem dosáhl nahrazení základních funkcí optického komparátoru.

Navíc jsem vytvořil nástroje sloužící k úpravě vstupního obrazu, čímž zajišťují maximální možnou kvalitu obrazu. Díky přehlednému designu se případný uživatel ve všech funkcích dobře vyzná. Má také možnost přepínat mezi barvami pozadí, může se rozhodnout buď pro černé s bílým otiskem nebo bílé s otiskem černým.

Po otestování vyplulo na povrch několik nedostatků. Program je příliš závislý na kvalitě vstupního materiálu, tedy otisku prstu. Dalším problémem shledávám rozlišení. Pokud je příliš vysoké, zpracování trvá příliš dlouho, ba co víc během filtrace vzniká v obraze šum a obraz je částečně znehodnocen. Je-li rozlišení naopak malé, dojde k nerozpoznání papilárních linií. U některých nedostatků jsou uvedeny možnosti řešení v případě budoucí úpravy této aplikace.

Díky této práci jsem se naučil spoustu nových pojmů i postupů. Doufám, že budu mít v budoucnu možnost dále se touto problematikou zabývat, doplnit tak své rezervy množstvím nových poznatků. Stejně tak bych si přál, aby i má práce našla další využití. Mohla by posloužit jako studijní materiál jiným kolegům nebo snad nalézt uplatnění přímo v praxi.

# Literatura

- [1] JAVŮREK, Karel. *Technologie otisku prstu: (ne)bezpečné zabezpečení*. Dostupné také z: <http://www.zive.cz/clanky/technologie-otisku-prstu-nebezpecne-zabezpeceni/sc-3-a-170784/default.aspx>
- [2] Vacula, Tomáš. Identifikace osob pomocí DNA: Bakalářská práce. Zlín: Fakulta aplikované informatiky UTB ve Zlíně, 2010, 52s. Vedoucí bakalářské práce Judr. Vladislav Štefka. Dostupné z: <http://hdl.handle.net/10563/12054>
- [3] STRAUS, Jiří, kolektiv. *Kriminalistika, kriminalistická technika: (pro kurz kriminalistických expertů)*. Vyd. 1. Praha: Vydavatelství PA ČR, 2004, 174s. ISBN 80-7251-165-3. S.24.
- [4] STRAUS, Jiří a PORADA Viktor. *Kriminalistická daktyloskopie*. Vyd. 1. Praha: Vydavatelství PA ČR, 2005, 285 s. ISBN 80-725-1192-0.
- [5] STRAUS, Jiří a PORADA Viktor. *Kriminalistická daktyloskopie*. Vyd. 1. Praha: Vydavatelství PA ČR, 2005, 285 s. ISBN 80-725-1192-0. S.48.
- [6] MUSIL, Jan; KONRÁD, Zdeněk; SUCHÁNEK, Jaroslav. *Kriminalistika*. Praha: Nakladatelství C. H. Beck, 2001. 512 s. ISBN 80-7179-362-0. S. 130.
- [7] Augustin Bartoš. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2015-05-10]. Dostupné z: [http://cs.wikipedia.org/wiki/Augustin\\_Barto%C5%A1](http://cs.wikipedia.org/wiki/Augustin_Barto%C5%A1)
- [8] Vichlenda, Milan. KRIMINALISTIKA. Karviná: Střední odborná škola ochrany osob a majetku s.r.o. Dostupné z: <http://lorenc.info/zaverecne-prace/priklad-citace-ze-skript.htm>
- [9] Fingerprint. *Crime Scene Forensics, LLC* [online]. [cit. 2015-05-10]. Dostupné z: <http://www.crimescene-forensics.com/Fingerprints.html>
- [10] DRAHANSKÝ, Martin a ORSÁG Filip. *Biometrie*. 1. vyd. [Brno: M. Drahanský], 2011, 294 s. ISBN 978-80-254-8979-6.
- [11] Biometrika. *Introduction to Fingerprints* [online]. [cit. 2015-05-10]. Dostupné z: [http://www.biometrika.it/eng/wp\\_fingintro.html](http://www.biometrika.it/eng/wp_fingintro.html)
- [12] HONG, Lin; JAIN, Anil. Fingerprint enhancement. In: *Automatic Fingerprint Recognition Systems*. Springer New York, 2004. p. 127-143. Dostupné z: <http://www.cse.msu.edu/biometrics/Publications/Fingerprint/cps96-45wacv.pdf>
- [13] HONG, Lin; WAN, Yifei; JAIN, Anil. Fingerprint image enhancement: algorithm and performance evaluation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1998, 20.8: 777-789.
- [14] MSIZA, Ishmael S., et al. Fingerprint segmentation: An investigation of various techniques and a parameter study of a variance-based method. *International Journal of Innovative Computing, Information, and Control (IJICIC)*, 2011, 7: 5313-5326.
- [15] Gradient (matematika). 2001-. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation [cit. 2015-05-10]. Dostupné z: [http://cs.wikipedia.org/wiki/Gradient\\_\(matematika\)](http://cs.wikipedia.org/wiki/Gradient_(matematika))

- [16] FILLA, D. Biometrie otisku prstu: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2011. 70 s. Vedoucí práce Ing. Petr Fedra.
- [17] Opencv. [online]. [cit. 2015-01-28]. Dostupné z: <http://opencv.org/>
- [18] GREENBERG, Shlomo, et al. Fingerprint image enhancement using filtering techniques. In: *Pattern Recognition, 2000. Proceedings. 15th International Conference on*. IEEE, 2000. p. 322-325.
- [19] SMĚKAL, O. Biometrie otisku prstu: Bakalářská práce. Brno: FEKT VUT v Brně, 2010, 42s.,2 příl. Vedoucí bakalářské práce Ing. Petr Fedra. Dostupné z: <[http://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=28074](http://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=28074)>
- [20] *Zhang-Suen thinning algorithm* [online]. [cit. 2015-05-10]. Dostupné z: [http://rosettacode.org/wiki/Zhang-Suen\\_thinning\\_algorithm](http://rosettacode.org/wiki/Zhang-Suen_thinning_algorithm)
- [21] *Qt Documentation* [online]. [cit. 2015-05-10]. Dostupné z: <http://doc.qt.io/>
- [22] MALTONI, Davide, et al. Handbook of Fingerprint Recognition. Second Edition. London :Springer, 2009. 496s. ISBN 978-1-84882-253-5.
- [23] Fingerprint matching in mobile devices (Android platform). *OpenCV* [online]. [cit. 2015-05-11]. Dostupné z: <http://answers.opencv.org/question/6364/fingerprint-matching-in-mobile-devices-android-platform/>
- [24] Biometriky nejen v pasech (1.). *Lupa.cz* [online]. [cit. 2015-05-11]. Dostupné z: <http://www.lupa.cz/clanky/biometriky-nejen-v-pasech-1/>
- [25] *Obrazce a znaky kůže* [online]. [cit. 2015-05-17]. Dostupné z: [http://krimi-spk.sweb.cz/02\\_exper/expertiz/02a\\_dakt/02a\\_kuze.htm](http://krimi-spk.sweb.cz/02_exper/expertiz/02a_dakt/02a_kuze.htm)
- [26] ZHANG, T. Y.; SUEN, Ching Y.. . A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 1984, 27.3: 236-239.

# Seznam příloh

## Příloha 1. Obsah CD

- /src/ - Zdrojové kódy aplikacemi
- /Doc/ - Tato písemná zpráva a zdrojový tvar písemné zprávy
- /Dokumentace/ - Úplná programová dokumentace
- /OpenCV/ - Použitá knihovna OpenCV
- /Otisky prstu/ - Uložené otisky pro ukázkou