



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

DEPARTMENT OF INTELLIGENT SYSTEMS

**IDENTIFIKACE ZVĚŘE NA ZÁKLADĚ BIOMETRICKÝCH
INFORMACÍ**

ANIMAL IDENTIFICATION BASED ON BIOMETRIC INFORMATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MAXIM PLIČKA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. ŠTĚPÁN RYDLO

BRNO 2023

Zadání bakalářské práce



146448

Ústav: Ústav inteligentních systémů (UITS)
Student: **Plička Maxim**
Program: Informační technologie
Specializace: Informační technologie
Název: **Identifikace zvěře na základě biometrických informací**
Kategorie: Zpracování obrazu
Akademický rok: 2022/23

Zadání:

1. Seznamte se s možnostmi identifikace na základě biometrických údajů konkrétně pomoci otisků prstů u lidí za účelem použitelnosti dané problematiky pro identifikaci zvěře. Zaměřte se na možnosti vyhledávání a identifikace na základě extrakce markantů.
2. Prostudujte metody používané pro identifikaci osob na základě otisků prstů a zaměřte se na možnosti použití k identifikaci zvěře na základě otisku rypáku.
3. Navrhněte aplikaci pro zpracování obrazu a extrakci vlastností za účelem vyhledávání markantů pro identifikaci zvěře.
4. Implementujte Vámi navržené řešení.
5. Zhodnoťte dosažené výsledky co se týče přesnosti hledání markantů a navrhněte případná rozšíření aplikace.

Literatura:

- J. R. PARKER. *Algorithms for Image Processing and Computer Vision*. John Wiley, 2010. ISBN 0470643854.
- MALTONI, Davide, Dario MAIO, Anil K JAIN a Salil PRABHAKAR. *Handbook of Fingerprint Recognition* [online]. 2nd ed. 2009. London: Springer London, 2009 [cit. 2022-10-19]. ISBN 978-1-84882-254-2.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 3.

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

Vedoucí práce: **Rydlo Štěpán, Ing.**
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 3.11.2022

Abstrakt

Tato bakalářská práce se zabývá získáním informací z fotografie vzoru nosu zvěře, které se využijí pro identifikaci jedince. Hlavním cílem je návrh a implementace aplikace, která dokáže tyto informace extrahovat ze vstupní fotografie a poté je vypsát, nebo najít fotografii v databázi, která se nejvíce shoduje se vstupní fotografií. Primárním účelem testování bylo určit přesnost extrakce těchto informací. Navržený přístup vykazoval 82,71% přesnost při databázi o velikosti 171 fotografií. Fotografie patřily zvěři typu jelen a srnka.

Abstract

This bachelor thesis is focused on obtaining information from a photograph of animal's muzzle pattern that will be used to identify the individual. The main goal is to propose and implement an application that can extract this information from the input photo and then output it, or find the photo in the database that most closely matches the input photo. The main purpose of the testing was to determine the accuracy of extracting this information. The proposed approach achieved 82.71% accuracy with a database size of 171 photos. The photographs belonged to deer and does.

Klíčová slova

identifikace zvěře, zpracování obrazu, vzor nosu, otisk prstu, biometrie

Keywords

animal identification, image processing, muzzle pattern, fingerprint, biometrics

Citace

PLIČKA, Maxim. *Identifikace zvěře na základě biometrických informací*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Štěpán Rydlo,

Identifikace zvěře na základě biometrických informací

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Štěpána Rydla. Další informace mi poskytla MVDr. Veronika Olejníčková, Ph.D.. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Maxim Plička
8. května 2023

Poděkování

Chtěl bych poděkovat svému vedoucímu bakalářské práce Ing. Štěpánu Rydlovi za odborné vedení, za pomoc a rady při zpracování této práce. Dále bych chtěl poděkovat MVDr. Veronice Olejníčkové, Ph.D. za poskytnutí cenných informací.

Obsah

1	Úvod	4
2	Fotografie nosu a otisk prstu	5
2.1	Biometrie	5
2.2	Fotografie nosu	6
2.3	Shrnutí dosavadních studií a jejich výsledků	9
2.4	Identifikace na základě otisku prstu	14
3	Předběžný návrh implementace	19
3.1	Navržené algoritmy	19
3.2	Návrh systému	26
4	Implementace	39
4.1	Použité technologie	39
4.2	Načítání fotografie	40
4.3	Předzpracování a extrakce markantů	41
4.4	Porovnávání šablon a export informací	47
5	Testování a vyhodnocení	48
5.1	Databáze	48
5.2	Přesnost extrakce markantů	48
5.3	Úspěšnost rozpoznávání	50
6	Závěr	51
	Literatura	52

Seznam obrázků

2.1	Struktura rhinarium [37]	7
2.2	Systém pro rozpoznávání dobytku na základě vzorů nosu [20]	10
2.3	Vytváření jednoho SIFT deskriptoru klíčového bodu: (a) vybraný SIFT klíčový bod z obrázku nosu, (b) výpočet gradientů pixelů v okně o velikosti 16x16, (c) výpočet deskriptoru klíčového bodu o velikosti 4x4 buněk s 8 orientacemi [20]	10
2.4	Porovnávání fotografií nosu na základě metody SIFT [20]	11
2.5	Systém rozpoznávání otisku prstu [5]	14
2.6	Předzpracování otisku prstu [39]	15
2.7	(a) skeletonizovaný otisk, (b) markanty [5]	17
2.8	Základní typy markantů [39]	17
2.9	Základní vzory papilárních linií, (a) spirála, (b) smyčka, (c) oblouk [39]	18
3.1	Aplikované jádra při filtraci. (a) obraz s hranou, (b) Gaussovo jádro, (c) bilaterální jádro [32]	20
3.2	Vyrovnaní histogramu [2]	21
3.3	Aplikace clip-limit na histogram [40]	21
3.4	Aplikace různých typů histogramové ekvalizace. (a) původní obraz, (b) AHE, (c) CLAHE [38]	22
3.5	Binarizace pomocí prahování. (a) původní obraz, (b) klasické prahování (hodnota práhu je 127), (c) adaptivní prahování na základě střední hodnoty, (d) adaptivní prahování na základě Gaussova jádra [31]	23
3.6	Způsoby aproximace kontury, nalevo se nachází kopie hran a napravo vrcholová reprezentace [31]	23
3.7	Huulova aproximace původní kontury [31]	24
3.8	Základní typy strukturních elementů [18]	25
3.9	Aplikace dilatace na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]	25
3.10	Aplikace eroze na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]	26
3.11	Aplikace morfologického otevření na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]	26
3.12	Navržený systém pro rozpoznávání zvěře na základě fotografie nosu	27
3.13	Ořezané fotografie nosů zvěře	28
3.14	Návrh získání oblasti zájmu	29
3.15	Anotace nosních dírek	30
3.16	Problémová fotografie pro anotaci	30
3.17	Orotovaná fotografie na základě delt	31

3.18	Návrh předzpracování oblasti zájmu	32
3.19	Obrázek před a po filtraci. (a) šedotónový obraz, (b) obraz po aplikaci bilaterálního filtru	33
3.20	Aplikace histogramové ekvalizace na obraz. (a) původní obraz, (b) histogramová ekvalizace, (c) CLAHE	33
3.21	Binarizovaný obraz	34
3.22	Opravený binarizovaný obraz	34
3.23	Odstranění vousků z obrazu. (a) původní obraz, (b) oblasti detekované ostřícím jádrem, (c) obraz po odstranění oblastí	35
3.24	Rozpojení dvou kontur. (a) původní obraz, (b) obraz po aplikaci morfologického otevření	35
3.25	Odstranění příliš malých/velkých kontur. (a) původní obraz, (b) obraz po filtraci	36
3.26	Odstranění kontur na základě pevnosti. (a) původní obraz, (b) obraz po filtraci	37
3.27	Odstranění kontur podle umístění. (a) původní obraz, (b) obraz po filtraci	37
3.28	Převod nalezených markantů na reprezentaci polem. (a) původní obraz s nalezenými markanty, (b) vizualizace pole s počtem markantů v jednotlivých buňkách	38
4.1	Použití různých parametrů bilaterálního filtru	42
4.2	Použití různých parametrů CLAHE	43
4.3	Použití různých způsobů pro určení hodnoty prahu, (a) průměr, (b) Gaussovo jádro	44
4.4	Použití různých parametrů adaptivního prahování	45
4.5	Použití různých parametrů morfologického otevření	46
5.1	Nejčastější problémy při určování markantů. Červeně jsou označeny špatně určené markanty. Modře jsou označeny správně určené markanty.	49
5.2	Graf reprezentující určení markantů	49
5.3	Graf reprezentující souvislost chybovosti na počtu extrahovaných markantů	50

Kapitola 1

Úvod

V současnosti se standardy pro identifikaci zvěře výrazně liší v závislosti na jejich druhu. V případě hospodářských zvířat je dle zákona 154/2000 Sb. (Zákon o šlechtění, plemenitbě a evidenci hospodářských zvířat a o změně některých souvisejících zákonů) vyžadováno, aby bylo zvíře označené. U domácích zvířat s výjimkou psů je toto označování spíše dobrovolné. Obecně můžeme značení dělit na trvalé (tetování, výžeh, elektronický čip) a dočasné (barva, obojek, visačky) [3]. Pro tuto práci je ovšem nejdůležitější divoká zvěř, která typicky žádné specifické značení nemá. Podle diplomové práce o myslivosti v etnologické perspektivě [34] je v České republice značení těchto zvířat prováděno až po jejich usmrcení myslivcem. Myslivce je ze zákona povinen mrtvé zvíře označit plombou. Jedná se o plastovou značku, která se umístí na zadní nohu a obsahuje státní znak a identifikační číslo. Toto číslo se také nachází v dokumentu o původu zvěře, kterým myslivce dokládá, že byl kus uloven legálně a je oprávněn jej převážet. Nevýhodou tohoto způsobu je hlavně nestálost této plomby. Toho by se dalo využít například jejím sundáním a přemístěním na nohu jiného zvířete. Tímto způsobem by se dal legalizovat přenos více kusů zvěře na stejný záznam.

Cílem této práce je pokusit se identifikovat zvěř na základě biometrických charakteristik, díky kterým by identitu zvířete nebylo možné nijak podvrhnout. Hlavní zkoumanou biometrickou charakteristikou zvěře bude v této práci fotografie nosu, na kterou se bude aplikovat přístup inspirovaný tomu z otisků prstů. Cílem navrženého přístupu bude extrahovat ze vstupní fotografie důležité body (markanty) a zhodnotit přenos jejich extrakce. Testování tohoto přístupu bude prozatím probíhat na fotografiích nosů srnek a jelenů (sarkatá zvěř), aby byla ověřena jeho použitelnost. V případě, že by se tento způsob projevil jako funkční, bylo by možné jej dále rozšířit i pro ostatní druhy zvěře jako neinvazivní (není potřeba zvíře nijak fyzicky narušit) způsob identifikace zvěře.

Kapitola 2

Fotografie nosu a otisk prstu

Tato kapitola je strukturována jako úvod do celého zbytku práce. Na začátku se věnuje obecnému úvodu do oblasti biometrie, která se aplikuje na zvířata. Dále se text zaměřuje na biometrickou charakteristiku, a to fotografii nosu, která byla zvolena pro rozpoznávání zvěře. Následuje část, která se zabývá shrnutím již provedených prací zabývajícími se rozpoznáváním zvěře na základě této charakteristiky. Kromě toho je na konci kapitoly stručně popsán způsob fungování již existujícího systému pro rozpoznávání lidí pomocí otisků prstů.

2.1 Biometrie

Biometrie neboli také biometrika je oblast vědy zabývající se zkoumáním biologických charakteristik jednotlivce, které slouží k identifikaci a případně autentizaci. Tyto charakteristiky mohou v případě lidí zahrnovat například otisky prstů, tvář, obličej a otisk dlaně. V dnešní době se s tímto pojmem nejčastěji setkáváme právě v případě lidí, ovšem existují již jisté studie, které biometrii aplikují i na zvířata, konkrétně tedy na psy [17] a hospodářská zvířata jako dobytek [26].

2.1.1 Biometrie v kontextu zvěře

V kontextu zvěře se biometrie využívá také pro identifikaci, která může sloužit například pro monitorování populace konkrétních druhů. Ornitologové mohou biometrické metody používat pro identifikaci ptáků na základě tvaru křídel nebo tvaru hlavy. Zoolozi mohou zvěř rozpoznávat na základě tvaru těla, velikosti nebo zbarvení. Zkoumat u zvěře můžeme například i tón zvuku nebo jejich chování. Mezi nejčastěji používané biometrické charakteristiky u parohaté zvěře patří zbarvení srsti, tvar rohů či parohů, zuby, oční duhovky, velikost těla a DNA. Jednotlivé charakteristiky byly konzultovány s MVDr. Veronikou Olejníčkovou, Ph.D a jsou popsány v následující části:

- Použití rohů a parohů pro univerzální identifikaci může být problematické, jelikož některé typy nebo pohlaví zvířat je nemají. Dalším problémem u rohů a parohů je nedostatečná variabilita.
- Oční duhovky jsou sice dostatečně variabilní ale také se nezdaří jako vhodný identifikátor a to hlavně z důvodu, že se po smrti jedince rychle zakalí a tím dojde k jejich degradaci.

- DNA je vhodná charakteristika pro identifikaci jedince, avšak je to velmi cenově nákladný a časově náročný způsob.
- Zuby by mohly být také vhodným identifikátorem, ovšem zajištění jednoduchého způsobu jejich zachycení by bylo náročné. Navíc se na zubech nachází pigment, který by se mohl měnit pod různým osvětlením.
- Co se týče barvy srsti, není dostatečně variabilní a stejně jako pigment zubů je velmi náchylná na osvětlení.
- U velikosti těla je značný problém s perspektivou. Bylo by velice obtížné zachytit celé tělo na fotografii a navíc do stejné fotografie zakomponovat nějaké měřítko na základě kterého by se dala odvodit reálná velikost. Navíc je stejně jako u řady jiných charakteristik problém s variabilitou.
- Další použitelnou charakteristikou by mohly být víry a chlupová pole srsti, které jsou popisované na hlavě skotu. Tyto útvary na hlavě nejsou náročné na světelné podmínky, ovšem jejich variabilita je zatím nezkoumaná a proto zatím nejsou vhodné pro identifikaci.

Fotografie nosu je snadno a rychle získatelná a není náročná na osvětlení. Nos navíc díky své velikosti a umístění většinou nebývá u odstřelu poničen. Další výhodou je, že po odstřelu narozdíl od duhovky nedochází k tak rychlé degradaci a tudíž se jeví jako vhodný biometrický identifikátor zvíře.

2.2 Fotografie nosu

Fotografie nosu zvířete se zaměřuje na detailní zobrazení oblasti hlavy, která obsahuje nos. Pro účely rozpoznávání vzorů nosů zvířat je důležitá vzorkovaná oblast nosu zvaná rhinarium.

Rhinarium (latinské označení pro „patří nosu“) nebo také *planum nasale* je neochlupená část nosu některých savců jako jsou například pes, kočka a jelen. Nachází se v okolí nosních dírek a typicky je povrch této části nosu méně či více vlhký, popraskaný a pokrytý útvary připomínající polygony [10]. Základní rozdíly v rhinarium u různých savců jsou velikost a šířka jednotlivých polygonů [37]. Tur domácí má například šířku jednotlivých polygonů 1950 - 3500 μm , zatímco takový pes má šířku polygonů pouze 750 - 881 μm [37]. Někteří savci mají tzv. *philtrum*, což je rýha, která vychází z prostředku vrchního rtu zvířete a končí zhruba uprostřed nosu. Obecně je tato část nosu vnímána jako smyslový orgán, díky kterému jsou jedinci schopni lépe rozpoznávat chemické složky vzduchu nebo také teplotu [37]. Díky tomu jsou například jeleni schopni rozpoznat, která potrava je pro ně bezpečná ke konzumaci a nebo jsou také schopni zjistit přítomnost predátora a to včetně směru odkud predátor přichází [1]. Podle studie o dobytku [33] je rhinarium jedinečné stejně tak jako je u lidí otisk prstu. Navíc je také struktura této části nosu během života jedince neměnná, a proto je ideální biometrickou charakteristikou pro identifikaci [9, 33].

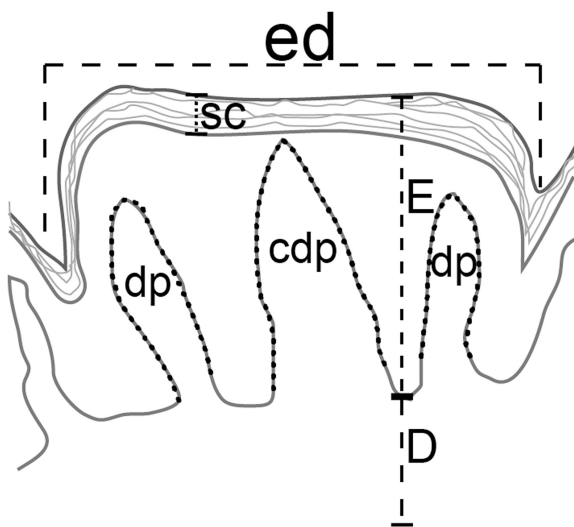
Právě z těchto důvodů se v této práci bude pracovat s rhinarium jako s oblastí zájmu pro určení markantů. **Markanty** zde budou reprezentovat středy jednotlivých polygonů. Jelikož jsou kombinace pozic, velikostí a tvarů těchto polygonů různé, jsou jejich středy vhodné pro rozpoznávání jedinců. Další možností, kterou by tyto markanty mohly představovat by byly například spojení rýh ohraničující tyto polygony, nebo případně ukončení rýh.

2.2.1 Struktura rhinarium

Struktura rhinaria se může odlišovat v závislosti na druhu, ale obecně se skládá z vícevrstvé kůže, která obsahuje velké množství nervových zakončení a také folikul (výhonky chlupů). Tyto nervové zakončení jsou spojeny s mozkem a umožňují zvýšené vnímání pachů. Kromě toho obsahuje rhinarium také mnoho malých otvorů, které umožňují přístup vzduchu do nosní dutiny, kde jsou vůně následně rozpoznány čichovými buňkami. Někteří jedinci si rhinarium olizují a tím zajišťují lepší zachytávání pachů [1]. Na obrázku 2.1 je nejdůležitější částí *epidermal dome*, což je vystouplá část kůže ohraničená rýhami. V této práci bude termín polygon používán jako synonymum pro *epidermal dome*, aby byl zdůrazněn jeho charakteristický tvar.

Na obrázku níže se nachází schéma popisující strukturu rhinarium 2.1. Výnamy zkratk jsou následující:

- ed - *epidermal dome*
- E - *epidermis*
- D - *dermis*
- sc - *stratum corneum*
- cdp - *central dermal papilla*
- dp - *dermal papillae*



Obrázek 2.1: Struktura rhinarium [37]

2.2.2 Poškození rhinaria

Obecně lze říci, že poškození postihující oblast rhinaria existuje značné množství a projevují se řadou možných defektů na kůži, které by mohly způsobit horší kvalitu rozpoznávání zvíře, jelikož by mohly narušit útvary na základě kterých se bude rozpoznávání v této práci provádět. V současné době se tyto poškození v databázi výrazně nevyskytují, a tak

nejsou pro tuto práci zcela klíčové. Nicméně, v případě rozšíření databáze by bylo vhodné je zahrnout, aby mohly být využity v souvislosti s navazujícími pracemi. Informace ohledně těchto poškození byla poskytnuta od MVDr. Veroniky Olejníčkové, Ph.D.

Poškození oblasti rhinaria je možné rozdělit na vrozené a získané. Získané poškození může být způsobeno zraněním, infekcí nebo může být výsledkem metabolického, imunitně zprostředkovaného či neoplastického procesu. V následující části budou popsány různé typy onemocnění rozdělené dle různých kritérií:

- Mezi infekční onemocnění popsané v Evropě patří *Dermatophilus congolensis*, *Parapoxvirus of red deer* a orf virus. Tyto onemocnění se projevují vznikem krust v oblasti rhinaria. Mezi další evropská onemocnění patří Papilomaviry, které se projevují vznikem kožních nádorů.
- Mezi infekční onemocnění zasahující oblast rhinaria popsané jinde ve světě patří *Deerpoxvirus*, *Herpesvirus* nebo také svrab. *Deerpoxvirus* se projevuje vznikem kožních poškození, krust a vředů. *Herpesvirus* způsobuje zánět kůže, vznik krust a vředů. V případě svrabu se poškození projevuje svěděním při kterém si zvěř nos odře. Dalším nádorovým onemocněním popsaným ve Finsku u soba polárního je kožní lymfom.
- Následující onemocnění zatím nebyly popsány u parohaté zvěře nebo se neprojevily přímo v oblasti rhinaria. *Virus vesicularis stomatitidis* je infekce u skotu, která vede ke vzniku puchýřků v oblasti rhinaria. *Capripoxvirus* je původcem nodulární dermatitidy skotu, která je charakteristická vznikem kožních uzlíků u skotu (poškození se projevuje i na oblasti rhinaria). *Mokokutánní pyoderma* je běžné onemocnění postihující oblast rhinaria a je způsobeno bakteriemi *Staphylococcus spp.*. Onemocnění se projevuje vznikem krust a erozí. *Stephanofilaria okinawaensis* způsobuje zánět kůže společně se vznikem krust. Plísňové onemocnění *Dermatophytózy* se vyskytuje u psů a projevuje vznikem papul, krust, erozí a vředů. *Canidiobolomykóza* je onemocnění, které bylo popsáno u psů a lamy. Onemocnění se projevuje vznikem vředů. Dalším onemocněním je *Besnoitia besnoiti*. Toto onemocnění bylo popsáno u soba polárního a domácích přežvýkavců v Německu, Francii a na jihu evropy. Následkem tohoto onemocnění je vasculitída se vznikem krevních sraženin a trhlin. *Leishmanióza* je onemocnění skotu popsané ve Švýcarsku. Poškození se projevuje jako hyperkeratóza, ulcerativní či papulární až nodulární dermatitida. *Neospora caninum* může jako mezihositele využít srnce nebo domácí přežvýkavce. Projevuje se na kůži rhinaria ulcerativní papulonodulární dermatitidou. *Prototheca spp.* je ojedinělé onemocnění postihující divoká i domácí zvířata ve světě. Projevuje se vředy, které se tvoří na místě, kde se spojují kůže a sliznice, hlavně kolem nozder a čenichu. *Pyemotes tritici* je roztoč, který zasahuje savce a vede ke vzniku poranění na čenichu.
- Dalším typem onemocnění, které by mohlo poškodit oblast rhinaria jsou metabolické a endokrinní onemocnění, které jsou popsány u domácích přežvýkavců. Mezi tato onemocnění patří dědičný nedostatek zinku, který vede ke vzniku kožních krust. Dále sem patří i *Hypothyreóza*, což je snížená funkce štítné žlázy a projevuje se šupinatou kůží s možným vznikem krust.
- Mezi autoimunitní onemocnění postihující kůži přežvýkavců patří *Epidermolysis bullosa*. Toto onemocnění se projevuje vznikem vředů a krust s možností sekundární infekce. Dále sem patří nemoc chladových algutininů, která vzniká postinfekčně na

autoimunitním podkladě. Nemoc se zhoršuje v chladu a projevuje se modřinkami, vředy až odumíráním tkáně na končetinách, včetně nosu.

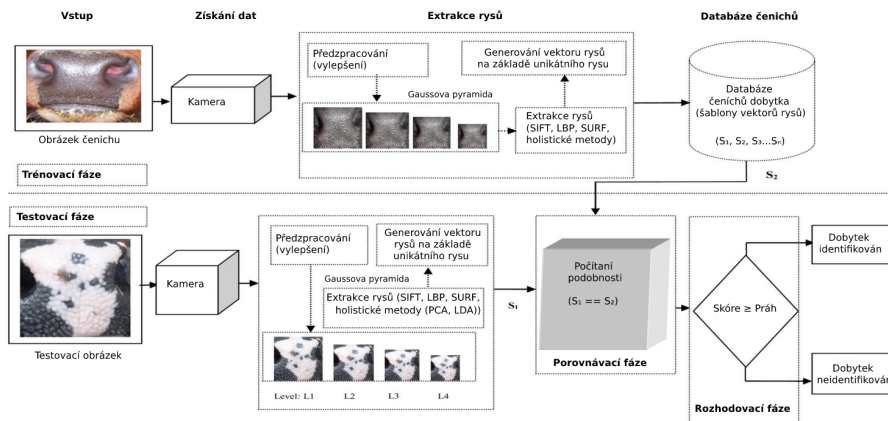
- Posledním typem onemocnění jsou onemocnění způsobená rostlinami a plísněmi. Mezi plísněná patří Stachybotryotoxikóza a T-2 toxikóza. Stachybotryotoxikóza je zapříčiněna toxiny plísně Stachybotrys spp., které mohou způsobit vznik vředů a odumírání tkáně na kůži domácích přežvýkavců. T-2 toxikóza je mykotoxin z plísně rodu Fusarium a spolu s jiným mykotoxinem, aflatoxinem, způsobuje především vyrážky s vředy na kůži. Mezi onemocnění způsobená rostlinami patří Fotosensitizující dermatitida a kontaktní fotodermatitida. Tyto onemocnění jsou zapříčiněna požitím nebo kontaktem s některými rostlinami a opět se projevují vznikem vyrážek a ostatních poškození. Poslední příklad těchto onemocnění vzniká požitím glykoalkaloidů solaninu a chaconinu a projevuje se vyrážkou a vředy na kůži v oblasti nosu.

2.3 Shrnutí dosavadních studií a jejich výsledků

Poslední dobou lze pozorovat zvýšený zájem o oblast pro rozpoznávání zvěře na základě biometrických vlastností. Některé studie se věnují rozpoznávání zvěře na základě jejich biometrických charakteristik jako jsou třeba pruhy zebry, vzory na těle gepardů a tygrů a nebo třeba stopy po zranění zanechané na těle mořských živočichů jako jsou žraloci a velryby [20]. Pro tuto práci budou ovšem nejpodstatnější výzkumy zabývající se rozpoznávání na základě vzorů na nosu. Pro rozpoznávání této konkrétní biometrické charakteristiky existuje již také řada studií. Většina se jich týká rozpoznávání dobytka u kterého je možné získat a porovnat úspěšnost jednotlivých přístupů. Tato část bude vycházet převážně z textu o dosavadních metodách pro rozpoznávání dobytka [20] a bude doplněna o shrnutí novějších studií založených na strojovém učení [15] a to včetně práci ohledně identifikace psů na základě jejich nosního vzoru [17].

2.3.1 Systém pro rozpoznávání dobytka na základě vzoru nosu

V této části textu se bude nacházet ilustrační obrázek toho, jak systém pro rozpoznávání dobytka může vypadat. Systém se dělí na dvě části, trénovací a testovací. V trénovací části se nasnímají fotky nosů a vyextrahují se šablony na základě rysů, ty se poté uloží do databáze. V testovací části se nasnímaný nos porovnává s ostatními nosy v databázi a snaží se najít jemu nejpodobnější. Obě tyto části se skládají z více kroků viz obrázek 2.2.



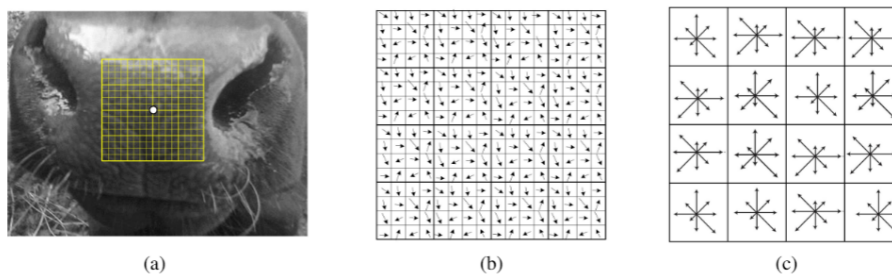
Obrázek 2.2: Systém pro rozpoznávání dobytku na základě vzorů nosu [20]

Získání dat

Prvním krokem tohoto systému je získání dat. V tomto kroku jsou data pomocí senzoru nasnímána a poté uložena do databáze. Důležitým faktorem při pořizování těchto dat je dostatečné rozlišení a kvalita.

Předzpracování a extrakce rysů

Dalším krokem je předzpracování a extrakce rysů. V tomto kroku se snažíme redukovat šum a ostatní chyby, které mohly vzniknout při pořizování obrázků. Redukce šumu lze dosáhnout použitím doplnpropustního filtru jako je například Gaussova pyramida viz 2.2. Dalším úkolem předzpracování je úprava otisku pro lepší extrakci rysů. Z tohoto důvodu se v závislosti na extrakčním algoritmu používá například skeletonizace a histogramová ekvalizace. Po předzpracování jsou z databáze extrahovány rysy, které popisují daný nos. K tomu se využijí například Scale invariant feautre transform (SIFT) [7, 30], Speeded up robust feature (SURF) [29]. Pro ukázkou je zde uveden obrázek, který popisuje postup vytváření jednoho SIFT deskriptoru 2.3.



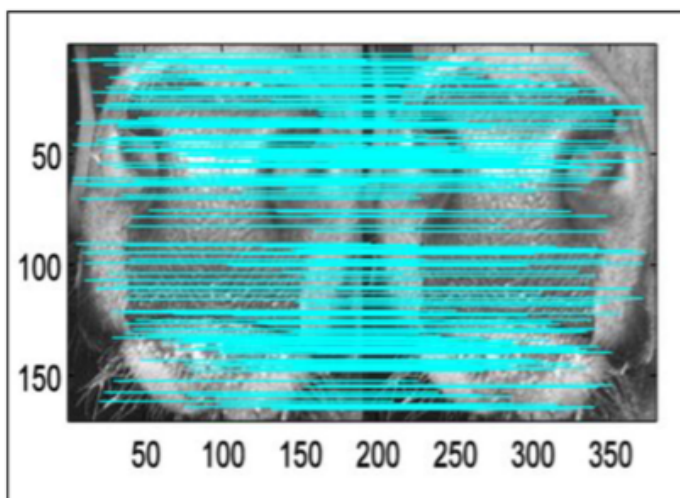
Obrázek 2.3: Vytváření jednoho SIFT deskriptoru klíčového bodu: (a) vybraný SIFT klíčový bod z obrázku nosu, (b) výpočet gradientů pixelů v okně o velikosti 16x16, (c) výpočet deskriptoru klíčového bodu o velikosti 4x4 buněk s 8 orientacemi [20]

Generování šablon

Následuje generování šablon reprezentující původní vzor nosu na základě extrahovaných rysů. Rysy vybrané pro reprezentaci nosu jsou vybrány na základě jejich unikátnosti. Vygenerovaná šablona je poté uložena do databáze.

Porovnávání

Poté nastane porovnávací fáze. V této fázi se rysy extrahované zvolenou metodou porovnávají s šablonou z databáze. Tím vznikne hodnota podobnosti dvou šablon, která v případě že dosahuje určitého prahu způsobí, že se nosy prohlásí za shodné. Pro ukázkou je na obrázku 2.4 uvedeno porovnávání na základě metody SIFT.



Obrázek 2.4: Porovnávání fotografií nosu na základě metody SIFT [20]

2.3.2 Konkrétní přístupy a jejich výsledky

První systém pro identifikaci dobytka na základě nosu byl v roce 2002 navržen v práci *Identification of beef cattle by analyzing images of their muzzle patterns lifted on paper* [28]. Systém pro rozpoznávání využíval pixely, které spojovaly drážky nacházející se na nosu. Nalezení těchto bodů bylo dosaženo pomocí řady technik pro zpracování obrazu jako filtrování, binarizace a skeletonizace obrazu. Tento systém jako vstup využíval naskenované otisky nosů zanechané na papíře. Výsledkem této práce byl systém, který dokázal rozpoznat dobytek s 30% úspěšností.

Další systém pro rozpoznávání zvíře pomocí počítačového vidění byl od autorů Ary Noviyanto a Aniaty Murni Arymurthy [30]. Tento systém využíval metody SIFT pro získání konkrétních rysů. Výsledný systém vykazoval přesnost 0.0167 equal error rate (EER) ¹. Navíc autoři vylepšili tuto metodu pomocí orientací jednotlivých rysů. Tímto se eliminovali některé milné shody. Výsledkem toho byl systém, který vykazoval přesnost 0.0028 EER, což je lepší než byl původní navržený způsob.

¹EER se používá jako ukazatel přesnosti biometrických systémů, přičemž čím víc se hodnota blíží k nule, tím lépe. Konkrétně se jedná o hodnotu, která vyjadřuje míru chybně pozitivních a chybně negativních výsledků.

V knize *2013 Federated Conference on Computer Science and Information Systems* [7] byl navržen systém, který využíval metody SIFT pro extrakci rysů a následné hledání shody. Pro větší robustnost systému byl SIFT skombinován s Random sample consensus (RANSAC). Z důvodu malé databáze obrázků, byly na obrázky aplikovány různé rozmazání, rotace a řezy, aby se otestovala robustnost tohoto systému. Díky kombinaci SIFT a RANSAC výsledný systém dosahoval přesnosti 93,3%, což bylo více než předešlé navržené systémy.

V roce 2013 byl v časopise *Transactions of the ASABE* navržen systém, který pro zpracování digitálních obrazů využíval Principal Component Analysis (PCA) a Euklidovské metriky pro klasifikaci. Hlavním elementem pro zvyšování úspěšnosti rozpoznávání tohoto přístupu bylo zvyšování počtu obrázků při trénování.

V knize *Proceedings of the 3rd European conference of computer science, ECCS* [29] byly rysy extrahovány pomocí metody SURF. Na základě provedených experimentů se dokázalo, že tento systém zvládne překonat předchozí výzkumy, které extrahovali rysy pomocí vlastních vektorů. S dostatečným počtem vstupních dat se ukázalo, že SURF může mít více než 90% přesnost. Dále zkoumali metodu Upright-SURF (U-SURF), která dokázala překonat i dříve zmíněnou metodu SURF, ale pouze za předpokladu, že data byla orientačně normalizována.

Další navržený systém byl popsán v časopise *International Journal of Image Mining* [27]. V tomto případě jsou vstupní obrazy prvně předzpracovány. K tomu autor využívá histogramové ekvalizace a filtrování pro redukci šumu. Následuje tzv. box-counting algorithm (BCA) pro detekci rysů daného nosu. Aby autor dosáhl lepších výsledků využil multiclass support vector machines (MSVM) pro klasifikaci.

Autoři v knize *Advanced Machine Learning Technologies and Applications* [35] pro extrakci rysů využili Gáborovy rysy a pro identifikaci využili strojové učení. Konkrétně se jednalo o support vector machines (SVM) s různými jádry. Na základě testování se ukázalo, že Gaussovo jádro vykazovalo nejlepší výsledky. Dále podobný tým autorů navrhl systém, který extrahoval rysy pomocí LBP [36]. V tomto výzkumu zkoušeli využít různých identifikačních metod, konkrétně tedy neural network (NN), k-nearest neighbour (K-NN), SVM a naivního bayese.

Výzkumů na toto téma je značné množství, ovšem z těch novějších lze vyvodit, že trend rozpoznávání dobytka na základě nosu spěje směrem ke strojovému a hlubokému učení. Jelikož se ovšem kategorie výzkumů, které využívají hlubokého učení, zaměřuje převážně na identifikaci dobytka na základě odlišných biometrických charakteristik než je fotografie nosu (například vzor srsti), tak zde nebudou brány v potaz. Mezi nejčastěji používané modely strojového učení pro identifikaci dobytka patří SVM, K-NN a artificial neural network (ANN) [15]. Nejčastějšími metodami pro extrakci samotných rysů patří convolutional neural network (CNN), local binary pattern (LBP), SURF, SIFT, Inception a Visual Geometry Group (VGG) [15]. Pro přehlednost zde bude uvedena tabulka, která obsahuje studie týkající se rozpoznávání dobytka pomocí metod strojového učení 2.1. Významy zkratk v tabulce, které doposud nebyly zmíněny jsou následující:

- F = F-skóre ²
- GSRC = Group sparsity residual constraint
- FLPP = Fuzzy linear preserving projections
- WLD = Weber's native descriptor

Reference	Nejlepší model	Velikost databáze	Metoda pro extrakci	Úspěšnost
Kusakunniran [25]	SVM	217	LBP	100%
Awad [6]	SVM	105	SURF	93%
Kumar [22]	GSRC	5000	LBP	93,87%
Kumar [21]	ANN	5000	LBP	96,74%
Kumar [23]	K-NN	5000	SURF	93,87%
Kumar [24]	SVM	5000	FLPP	96,87%
Gaber [13]	AdaBoost	217	WLD	98,9%
Zaorálek [41]	SVM	322	SVD	75F
Ahmed [4]	SVM	217	SURF	100%
Tharwat [35]	SVM	217	Gáborův filtr	99,5%
Tharwat [36]	SVM	217	LBP	99,5%
El Henawy [11]	ANN	1060	BCA	99,18%

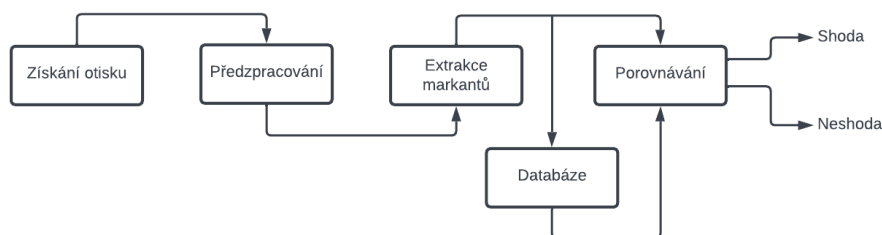
Tabulka 2.1: Tabulka shrnující přístupy strojového učení pro rozpoznávání dobytka na základě vzoru nosu [15]

V roce 2020 byl navrhnout podobný systém pro identifikaci psů [17]. Ten pro předzpracování využíval histogramové ekvalizace a změny velikosti obrazu. Pro extrakci rysů z otisku bylo využito několika metod u kterých bylo provedeno následné srovnání úspěšnosti extrakce. Těmito metody byly konkrétně SIFT, SURF, Binary Robust Invariant Scalable Keypoints (BRISK) a oriented FAST and rotated BRIEF (ORB). Porovnávání nalezených rysů se dělalo dvěma způsoby. V případě SIFT a SURF byl použit Fast library for approximate nearest neighbors (FLANN). V případě BRISK a ORB byla použita hammingova vzdálenost. Testování probíhalo na databázi o velikosti 990 obrázků, které byly vytvořeny deformacemi, rotacemi a změnami osvětlení u původních 55 obrázků. Těchto 55 obrázků nosů patřilo 11 různým psům. Z výsledků vyplynulo, že nejlépe si vedla metoda ORB, která měla 0,35 EER.

²F-skóre je metrika pro vyhodnocení kvality klasifikace modelu strojového učení, což znamená rozdělení do dvou kategorií, například pozitivní a negativní. F-skóre se vypočítává se na základě přesnosti a úplnosti. Přesnost je definována jako počet správně klasifikovaných jako pozitivních zatímco úplnost vyjadřuje, kolik z celkového počtu pozitivních bylo klasifikováno jako pozitivní. F-skóre se poté vypočítá následovně $F\text{-skóre} = \frac{2 * (\text{presnost} * \text{uplnost})}{\text{presnost} + \text{uplnost}}$.

2.4 Identifikace na základě otisku prstu

Následující text se bude věnovat celému procesu rozpoznávání otisku prstu. První část ohledně předzpracování otisku je převzata z [39, 19], druhá část ohledně extrakce markantů je z [16]. Poslední část, kde je stručně sepsáno rozpoznávání a klasifikace otisku je převzata z [39]. Na následujícím obrázku lze vidět jak celý systém rozpoznávání otisků prstů může vypadat 2.5.



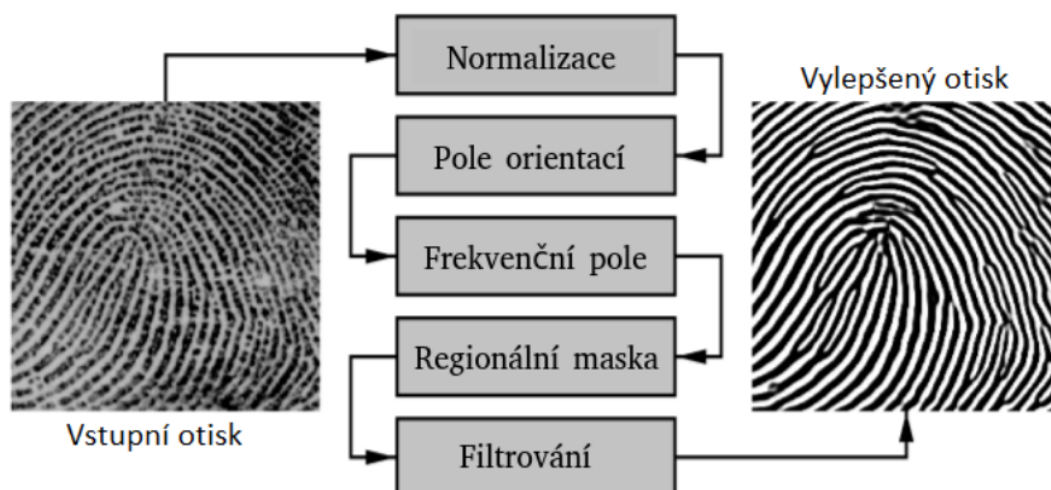
Obrázek 2.5: Systém rozpoznávání otisku prstu [5]

2.4.1 Otisk prstu

Otisk prstu je jedinečný a nezaměnitelný vzor tvořený papilárními liniemi, který vzniká kontaktem pokožky prstu s jiným povrchem. Na otisku prstu se nachází tzv. papilární linie, což jsou výběžky kůže, na základě kterých se celé rozpoznávání řídí. Tyto linie vytvářejí různé dráhy a křivky, které jsou specifické pro každého jednotlivce a tudíž nejsou stejné u žádných dvou osob. Otisk prstu je tvořen hřebeny a brázdami. Hřeben je vystouplá část kůže tvořící papilární linie zatímco brázda je naopak prohlubeň mezi hřebeny. Zde lze pozorovat podobnost s rhinariem, kde jsou polygony ohraničené rýhami. Na otiscích uvedených níže budou hřebeny reprezentovány černou barvou a brázdy naopak barvou bílou.

2.4.2 Předzpracování otisku prstu

Předzpracování slouží v případě otisku prstu k vylepšení kvality nasnímaného otisku. Zabývá se opravou obnovitelných regionů a odstraněním regionů neobnovitelných. Dělí se do kategorií podle velikosti oblastí na základě aplikace algoritmu použitého pro opravu. Díky těmto opravám jsme schopni v otisku odstranit chyby, které by mohly způsobit, že navazující algoritmus pro extrakci markantů nebude pracovat dostatečně dobře. Na následujícím obrázku je uvedena ukázka, jak předzpracování může vypadat jako sekvence po sobě jdoucích operací 2.6.



Obrázek 2.6: Předzpracování otisku prstu [39]

Normalizace

Prvním krokem předzpracování je normalizace. Během normalizace se zmenšují rozdíly intenzit pixelů mezi hřebeny a brázdami. Toho algoritmus vypočítá novou hodnotu pixelu na základě aktuální hodnoty pixelu a statických ukazatelů obrazu jako je průměr a rozptyl. Díky tomu nebude následující algoritmus pro rozpoznávání orientací rušen šumem a dalšími nevyžádanými prvky jako je například různý kontrast mezi hřebeny a brázdami, který může být způsoben rozdílnou silou přitlaku [19].

Pole orientací

Pole orientací je způsob reprezentace směru papilárních linií v obrazu rozděleném na nepřekrývající se bloky. Jeden tento blok si můžeme představit jako vyřiznutou část vstupního obrázku například o velikosti čtverce. Algoritmus pro určení orientací využívá vlastnosti, že směr linie v bloku jsou ortogonální vůči jejímu gradientu.

Frekvenční pole

Frekvenční pole spolu s polem orientací slouží jako vstup pro finální vylepšení otisku pomocí Gáborova filtru. Toto pole, stejně jako je tomu u pole orientací, obsahuje jednotlivé bloky obrazu. Tentokrát je ovšem blok reprezentován sinusovou funkcí, jejíž průběh kopíruje výskyt papilárních linií se směrnici kolmou k orientaci linií v daném bloku. Tento postup se aplikuje pouze v případech, kdy se v bloku nenachází žádný markant nebo singulární bod [39].

Regionální maska

Regionální maska je reprezentace otisku po regionech, kde každý jeden region má přidělen hodnotu na základě jeho kvality. Ta se dělí do těchto tří typů:

- Dobře specifikovaný region

- Obnovitelný poškozený region
- Neobnovitelný poškozený region

První dvě kategorie lze sumarizovat jako kategorie obnovitelné a tu poslední jako neobnovitelnou. Pro určení do které kategorie daný region spadá se využívá řady kritérií. Po určení kategorie všech regionů, lze stanovit poměr obnovitelných regionů vůči regionům neobnovitelným. Tento poměr představuje úroveň kvality. V případě potřeby se tímto způsobem dá rozlišit kvalitní a nedostatečně kvalitní vstupní otisk.

Filtrování

Posledním krokem v procesu předzpracování je filtrování. V této části se využije vypočítaných frekvencí a orientací v blocích, které poslouží jako vstupy do Gáborova filtru. Tento filtr následně po blocích odstraňuje šum a zároveň neupravuje hřebeny a brázdy. Výsledek po filtrování lze vidět na obrázku 2.6.

2.4.3 Extrakce markantů

Extrakce markantů je proces, jehož vstupem je předzpracovaný obraz. Cílem tohoto procesu je lokalizace a zakódování důležitých bodů (markantů), které budou blíže popsány v části o rozpoznávání a klasifikace otisku prstu [5]. Kromě polohy markantu se většinou ještě uchovává jejich typ a směr. Existuje spousta metod pro extrakci markantů. Některé z nich využívají přímo šedotónového obrazu, z kterého za pomoci Gáborova filtru extrahují markanty [42]. Většinou se ovšem na vylepšený obraz aplikuje následující postup.

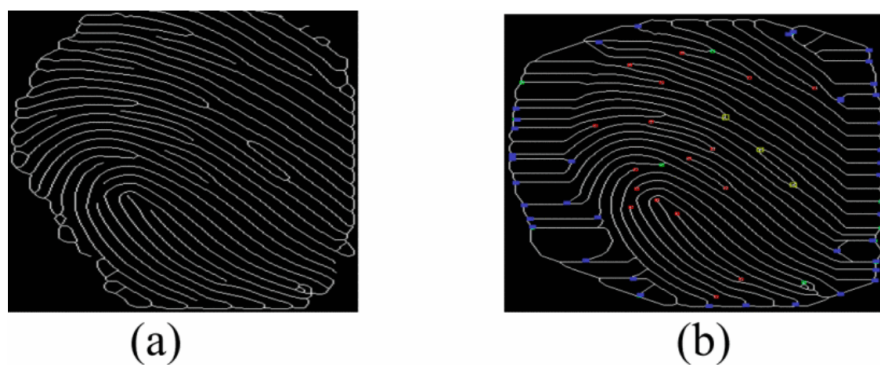
Binarizace

Prvním krokem je binarizace, což je převod šedotónového obrazu do pouze dvou úrovní (černá a bílá). Tento proces se provádí za pomoci hodnoty prahu, která určuje jestli se aktuální hodnota pixelu změní na bílou a nebo černou.

Skeletonizace

Následně se provádí skeletonizace obrazu. Tento proces převádí binarizovaný obraz na jeho kostru. Tuto kostru lze chápat jako výsledek morfologických operací (eroze), které postupně odstraňují vnější část linií až do doby, kdy už linie nejde více ztenčit.

Následně se na skeletonizovaný otisk aplikuje algoritmus pro vyhledávání markantů. Výsledek extrakce markantů bude vyznačen na následujícím obrázku 2.7.



Obrázek 2.7: (a) skeletonizovaný otisk, (b) markanty [5]

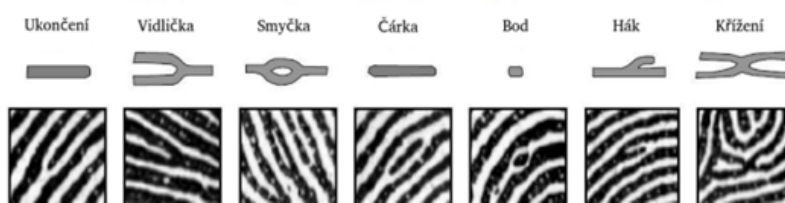
2.4.4 Porovnávání a klasifikace otisku prstu

Následující text bude rozčleněn do dvou částí, kde se první část bude zabývat procesem porovnávání otisků prstů a druhá bude o klasifikaci otisku prstu.

Porovávání otisků prstů

Porovnávání otisku prstu využívá získané množiny markantů z předchozího kroku. Typickým příkladem porovnávání na základě množiny markantů je Hongova a Rathova metoda [12]. V prvním kroku se porovnávané otisky zarovnají (typicky podle singulárního bodu typu jádro) a následně se provede porovnávání markantů. Během toho se porovnávají dříve zjištěné vlastnosti jednotlivých markantů jako je jejich pozice, směr a typ. V případě, že si je vstupní otisk dostatečně podobný s otiskem v databázi, jsou otisky prohlášeny za shodné. Míra této shody závisí na nastavené hranici. V případě, že je tato hranice nastavena jako příliš malá, bude pravděpodobněji docházet ke shodě dvou různých otisků. Naopak v případě, že hranice je nastavena příliš vysoko, mohou i drobné nečistoty či jiné odchylky způsobit, že systém otisky označí za neshodné.

Markanty jsou specifické útvary tvořené papilárními liniemi a mohou se dělit do různých typů na základě útvaru, který tvoří papilární linie. Na základě těchto bodů lze unikátně reprezentovat jakýkoli otisk prstu. Pro tuto práci budou markanty obzvlášť stěžejní, jelikož se pomocí nich bude popisovat nos zvěře viz 2.2. Pro ukázkou je na obrázku 2.8 uvedeno pár základních typů markantů u otisku prstu.



Obrázek 2.8: Základní typy markantů [39]

Klasifikace otisků prstů

Klasifikace otisků se provádí na základě počtu a pozic singulárních bodů, což jsou důležité globální body. Tyto body se dělí na delty a jádra. Jádro je definováno jako nejvyšší bod nejvnitřnější smyčky a delta je středem trojúhelníkových oblastí. Na obrázku 2.9 jsou zobrazeny základní vzory papilárních linií. Tyto vzory se od sebe liší pozicemi a počtem singulárních bodů. Klasifikace otisků má za cíl zvýšit efektivitu srovnávacího algoritmu tím, že umožňuje porovnávat otisky, které spadají do stejné kategorie.



Obrázek 2.9: Základní vzory papilárních linií, (a) spirála, (b) smyčka, (c) oblouk [39]

Kapitola 3

Předběžný návrh implementace

V první části této kapitoly jsou uvedeny některé základní pojmy a algoritmy, které je nutné znát pro porozumění návrhu a implementaci systému pro rozpoznávání zvířet. Druhá část popisuje samotný návrh tohoto systému. Kromě popisu jednotlivých částí, ze kterých se systém skládá, jsou zde uvedeny také krátké náčrtky myšlenek, které nebyly použity v konečné verzi implementace, včetně vysvětlení důvodů, proč od nich bylo opuštěno.

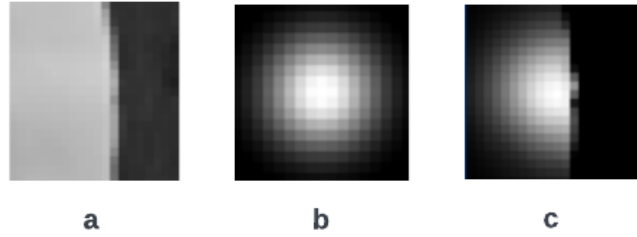
3.1 Navržené algoritmy

V této podkapitole budou uvedeny všechny pojmy a algoritmy potřebné pro pochopení návrhu implementace.

3.1.1 Bilaterální filtr

Filtr je založen na Gaussově filtru, který pro filtrování využívá konvoluce¹ okolních pixelů s 2D Gaussovým jádrem. Problémem tohoto filtru je, že pro výpočet nové intenzity pixelu bere v potaz pouze jeho okolí, což může zapříčinit, že na místech kde se nachází velké rozdíly intenzit pixelů (hrany) se tyto rozdíly vyhladí. Z tohoto důvodu se aplikuje ještě druhý Gaussov filtr, který se dívá na intenzity pixelů v okolí. Ten způsobí, že při aplikaci filtru budou významnější pixely, které mají podobnější intenzitu. Díky tomu bilaterální filtr zachovává rozdíly intenzit pixelů v místech, kde se nachází hrany a zároveň vyhlazuje obraz a odstraňuje šum. Ukázka vytvoření bilaterálního jádra na základě původního obrazu a Gaussova jádra se nachází na následujícím obrázku 3.1.

¹Konvoluce je matematická operace, která spojuje dvě funkce reprezentující obraz a jádro. Jádro lze chápat jako tabulku, která obsahuje váhy určené na základě koeficientů filtru. Konvoluce se provádí tak, že se jádro posouvá přes obraz a v každém bodě se vypočítá vážený průměr hodnot v okolí zpracovávaného bodu. Tento průměr poté představuje novou hodnotu pixelu.



Obrázek 3.1: Aplikované jádra při filtraci. (a) obraz s hranou, (b) Gaussovo jádro, (c) bilaterální jádro [32]

Dále zde bude rozepsán vzoreček, který bilaterální filtr používá a to včetně popisku k jednotlivým částem, z kterých se skládá 3.1 3.2.

$$BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{o_s}(\|p - q\|) G_{o_r}(|I_p - I_q|) I_q \quad (3.1)$$

kde

$$W_p = \sum_{q \in S} G_{o_s}(\|p - q\|) G_{o_r}(|I_p - I_q|) \quad (3.2)$$

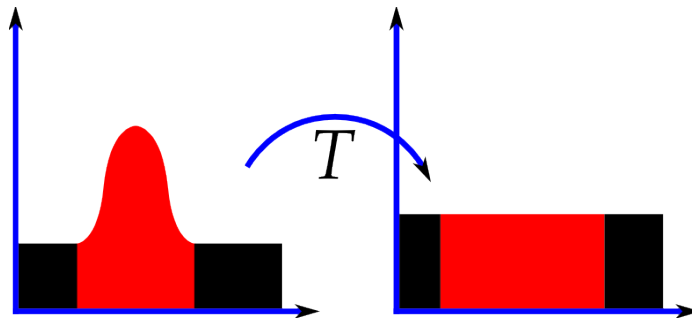
- $BF[I]_p$ - výsledná intenzita pixelu p
- $\frac{1}{W_p}$ - normalizační část, zajišťuje, že součet vah v okolí bude dohromady dávat 1
- $G_{o_s}(\|p - q\|)$ - váhy podle vzdálenosti od daného pixelu
- o_s - uvažované okolí pixelu
- $G_{o_r}(|I_p - I_q|)$ - váhy podle rozdílů intenzit pixelů
- o_r - minimální amplituda hrany, čím víc se blíží nekonečnu, tím víc se filtr podobá Gaussovu filtru
- I_q - intenzita pixelu q

3.1.2 Ekvalizace histogramu

Ekvalizace histogramu je globální metoda, která slouží k zlepšení kontrastu a jasnosti obrazu. Metoda je založena na analýze histogramu, který obsahuje rozložení intenzit pixelů v obraze. Hlavním úkolem histogramové ekvalizace je změnit rozložení intenzit pixelů v obraze tak, aby byly rovnoměrněji rozprostřeny přes celý rozsah intenzit šedi 3.2. Toho se dosáhne tak, že se nejprve normalizuje histogram, aby všechny jeho hodnoty spadaly do rozsahu 0 až 1. Tento krok se dělá tak, že každý sloupec histogramu bude reprezentován pravděpodobnostmi, že při výběru náhodného pixelu obrazu, bude tento pixel spadat do daného sloupce. Následně se vypočte kumulativní distribuční funkce (CDF)² histogramu.

²Kumulativní distribuční funkce je neklesající funkce, která určuje pravděpodobnost, že při výběru náhodné proměnné bude tato proměnná menší než zadaná hodnota.

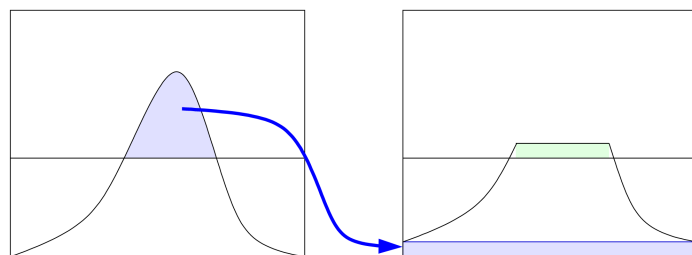
Nakonec se pomocí inverzní distribuční funkce vytvoří nová intenzita pro všechny pixely. Nevýhodami této metody jsou zvýraznění šumu v obraze a ztráta detailu v příliš tmavých nebo světlých oblastech.



Obrázek 3.2: Vyrovnání histogramu [2]

Vylepšením této metody je tzv. **adaptivní histogramová ekvalizace** (AHE), která rozděljuje obraz do bloků. Pro každý z těchto bloků se vypočítá histogram a následně se provede jeho vyrovnání. To umožňuje adaptivní vyrovnání histogramu, které je přizpůsobeno lokálnímu kontrastu a jasů v daném bloku. Po vyrovnání dojde k vyhlazení přechodů mezi jednotlivými bloky pomocí bilineární a lineární interpolace. Tento krok se dělá z důvodu, aby se zamezilo viditelným přechodům mezi jednotlivými bloky. Tato metoda ovšem také není ideální, jelikož může zesílit šum v částech obrazu s podobnými intenzitami pixelů (tzv. bílý šum).

Z toho důvodu existuje vylepšení této metody nazývaní se **kontrastně limitovaná adaptivní histogramová ekvalizace** (CLAHE). Ta před výpočtem nové hodnoty pixelů pomocí „clip-limit“ ořezává sloupce histogramu, které přesahují zvolenou hranici a rovnoměrně je rozděluje mezi ostatní sloupce viz 3.3. Díky tomu se zachová původní četnost intenzit pixelů a zároveň se zamezí zesílení šumu.



Obrázek 3.3: Aplikace clip-limit na histogram [40]

Na následujícím obrázku se nachází ukázka rozdílu mezi AHE a CLAHE 3.4. Na obrázku lze vidět, že AHE sice zlepšuje kontrast, ovšem za cenu vytvoření nežádoucího šumu, zatímco CLAHE také zlepšuje kontrast a zároveň nevytváří nežádoucí šum.



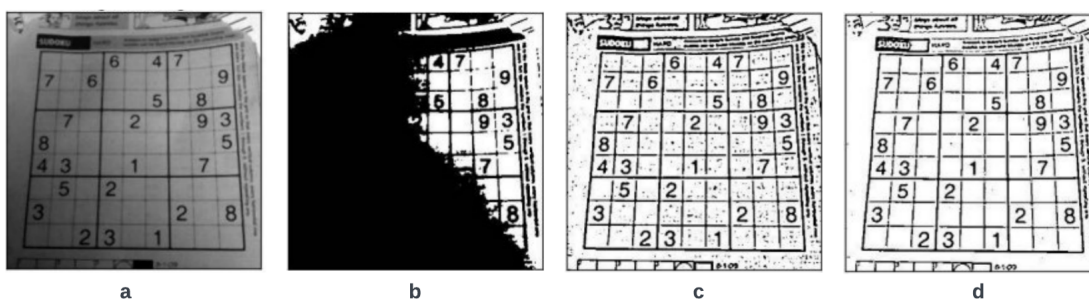
Obrázek 3.4: Aplikace různých typů histogramové ekvalizace. (a) původní obraz, (b) AHE, (c) CLAHE [38]

3.1.3 Binarizace - adaptivní prahování

Binarizace je převod šedotónového obrazu na jeho černobílou reprezentaci. Toho se dá dosáhnout pomocí **prahování**. Prahování je metoda, v které intenzita výsledného pixelu závisí na intenzitě vstupního pixelu a hodnotě prahu. Konkrétně se tak děje porovnáváním intenzity pixelu s hodnotou prahu a v případě, že je intenzita větší nežli je hodnota prahu se pixel obarví na bílo (intenzita s hodnotou 255), v opačném případě se pixel obarví na černo (intenzita s hodnotou 0). Hlavním problémem klasického prahování na základě jedné hodnoty prahu je binarizace obrazu s proměnlivým osvětlením. V tomto případě se totiž díky osvětlení intenzity pixelů liší v různých částech obrazu a tudíž je jedna hodnota prahu nedostačující.

Z tohoto důvodu existuje **adaptivní prahování**. Tato metoda rozděluje obraz na bloky, v kterých se pomocí statistických ukazatelů obrazu určí hodnota prahu pro daný blok a následně se v něm provede binarizace. Určení této hodnoty se může dělat několika způsoby. Mezi nejpoužívanější patří určení prahu na základě střední hodnoty³ pixelů v bloku, nebo výpočet hodnoty prahu na základě intenzit pixelů v bloku vynásobeným Gaussovým jádrem. Na následujícím obrázku bude uvedena ukázka rozdílu mezi klasickým prahováním a adaptivním prahováním s různými jádry 3.5.

³Střední hodnota vznikne sečtením hodnot všech pixelů v požadované oblasti. Následně se tato hodnota podělí počtem hodnot a tím vznikne střední hodnota neboli průměr.

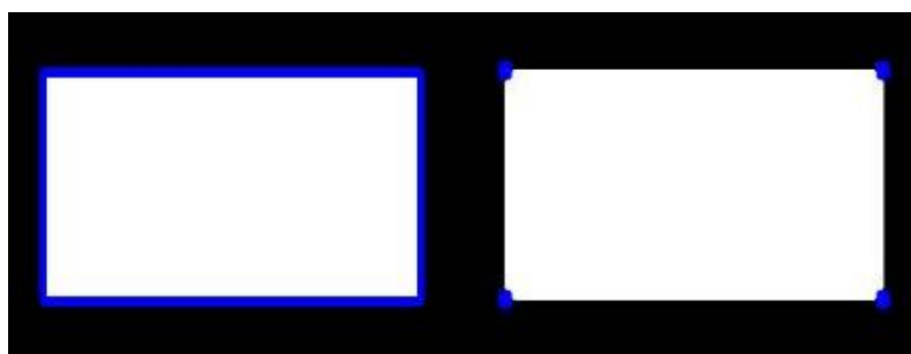


Obrázek 3.5: Binarizace pomocí prahování. (a) původní obraz, (b) klasické prahování (hodnota práhu je 127), (c) adaptivní prahování na základě střední hodnoty, (d) adaptivní prahování na základě Gaussova jádra [31]

3.1.4 Kontury - momenty, hierarchie, huul

Kontury jsou důležitým pojmem v oblasti počítačového vidění, konkrétně jsou užitečné pro analýzu tvaru nebo detekci a rozpoznání objektů v obraze. Konturu lze definovat jako křivku, která je tvořena body podél hranice se stejnou intenzitou. Tyto kontury lze vyhledávat jak v černobílém tak v šedotónovém obraze, ovšem jelikož intenzity hran v šedotónovém obraze nebývají stejné, může v něm být vyhledávání kontur nevhodné. Co se týče černobílého obrazu, je nutné aby byly hledané objekty reprezentovány bílou barvou, jelikož to aktuální implementace knihovny openCV vyžaduje pro správné nalezení kontur. Hlavní výhodou reprezentace objektů pomocí kontur je množina operací a vlastností, které nad nimi openCV poskytuje. Dále se text bude věnovat vysvětlením několika základních pojmů a operací, které jsou s konturami spjaty, jelikož se s nimi bude v práci dále pracovat.

Reprezentace kontury může být provedena buď pomocí vrcholů, které konturu tvoří, nebo pomocí úplné kopie hran. Kopie hran zabírá příliš místa a není pro tuto práci příliš důležitá, tudíž je výhodnější použít bodovou reprezentaci. Na následujícím obrázku bude zobrazen rozdíl mezi těmito dvěma reprezentacemi 3.6.



Obrázek 3.6: Způsoby aproximace kontury, nalevo se nachází kopie hran a napravo vrcholová reprezentace [31]

Dále jsou důležité struktury, v nichž lze reprezentovat **uspořádání kontur**. Pro tuto práci bude nejdůležitější reprezentace listová a stromová. Listová reprezentace je způsob, který nijak nereflektuje uspořádání kontur a pouze obsahuje všechny nalezené kontury. Stromová reprezentace na rozdíl od listové umožňuje zjistit, které kontury jsou obsaženy v jiných.

Momenty jsou matematické funkce, kterými se popisují různé vlastnosti dané kontury. Mezi vlastnosti důležité pro tuto práci patří plocha a střed kontury. Ukázka rovnice popisující konturu se nachází níže 3.3. Výpočet **plochy kontury** je poté už pouze suma světlých pixelů přes ohraničující okno neboli se jedná o hodnotu M_{00} dosazenou do vzorečku 3.3. Pro výpočet souřadnic **středu kontury** se poté používá vzorec 3.4.

$$M_{ab} = \sum_x \sum_y x^a y^b I(x, y) \quad (3.3)$$

$$\{x, y\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\} \quad (3.4)$$

Dalším pojmem je tzv. **Huulova aproximace**, která dokáže určit konvexní konturu na základě té původní. Ukázka vizualizace Huulovy aproximace je na následujícím obrázku 3.7. Původní kontura je zde reprezentována obrysem ruky a její Huulova aproximace je znázorněna červenou.



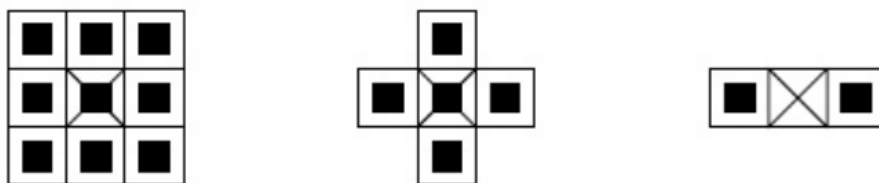
Obrázek 3.7: Huulova aproximace původní kontury [31]

Konvexní aproximaci lze využít pro určení úplnosti. **Úplnost** se vypočítá jako poměr ploch původní kontury vůči aproximované konvexní Huulově kontuře. Díky tomu lze v této práci určit, jak moc je kontura poškozená, respektive jak dobře byla kontura určena pro původní polygon.

3.1.5 Morfologické operace - otevření

Morfologické operace jsou matematické transformace, které pomocí aplikace strukturního elementu upravují obraz. Strukturní element je malý obrazec, který je obvykle definován jako malá matice nebo jádro. Strukturní element může nabývat různých tvarů jako například čtverec, obdélník a kříž viz 3.8. Jeho velikost a tvar pak určuje, jaký druh transformace

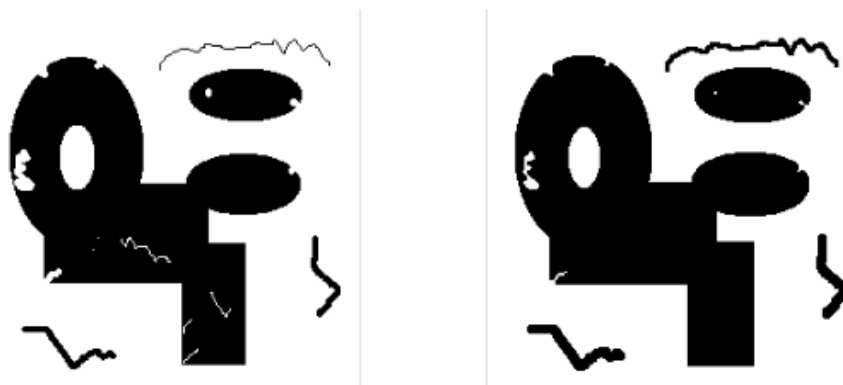
bude proveden na vstupní obraz. Při aplikaci se strukturální element posouvá přes vstupní obraz a určuje, jak se změní hodnoty pixelů výstupního obrazu.



Obrázek 3.8: Základní typy strukturálních elementů [18]

Morfologické operace pracují s binárním nebo šedotónovým obrazem a používají se například pro detekci hran, segmentaci obrazu, odstranění šumu (detailů) a ztenčování nebo rozšiřování objektů. V této práci se budou morfologické operace používat na binární obraz. Mezi základní morfologické operace patří eroze, dilatace, otevření a uzavření. Operace typu uzavření zde ovšem nebude vysvětlena, jelikož to není potřeba vzhledem k povaze práce.

Dilatace je druh operace, který se používá k vyplnění malých děr, rozšiřování objektů a tvoří základ pro složitější morfologické operace jako otevření a uzavření. Na následujícím obrázku bude ukázka aplikace dilatace 3.9.



Obrázek 3.9: Aplikace dilatace na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]

Eroze je operace, která je duální vůči dilataci. Používá se pro odstranění a ztenčení objektů v obraze nebo umožňuje získat obrys objektu. Stejně jako dilatace tvoří základ pro složitější morfologické operace. Na následujícím obrázku bude uvedena ukázka aplikace eroze 3.10.



Obrázek 3.10: Aplikace eroze na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]

Morfologické otevření je operace, která využívá kombinace eroze a dilatace. Díky její aplikaci je možné získat obraz, kde budou odděleny úzce spojené objekty a odstraněny malé detaily [14]. Tato operace byla vytvořena kvůli tomu, že eroze nejen odstraňuje malé detaily v obraze, ale také ztenčuje objekty v něm. Z tohoto důvodu se na obrázek aplikuje dilatace, která obnovuje původní šířku objektů a současně ponechává odstraněné detaily. Otevření je tedy ve výsledku eroze následovaná dilatací. Na následujícím obrázku bude ukázka aplikace morfologického otevření 3.11.

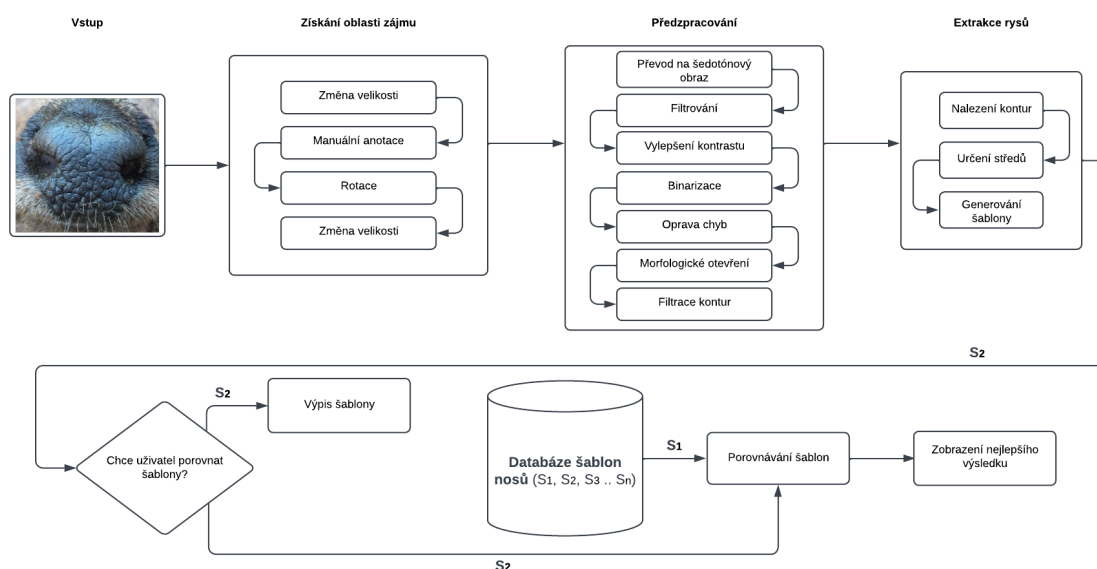


Obrázek 3.11: Aplikace morfologického otevření na vstupní obraz se strukturálním elementem o rozměrech 3x3 [18]

3.2 Návrh systému

Systém bude fungovat jako konzolová aplikace, kde uživatel zadá cestu ke vstupní fotografii a případně cestu ke složce s šablonami, s kterými chce fotografii porovnat. V případě, že uživatel nebude chtít fotografii porovnávat, vypíše se mu na standardní výstup textová reprezentace vstupní fotografie (šablona markantů). Celkově se systém bude skládat ze čtyř částí. Na začátku je potřeba z fotografie získat oblast zájmu. Ta se získá pomocí rotace a změny velikosti, díky kterým bude výsledná oblast zájmu ve stejném formátu pro další rozpoznávání. Následně je potřeba oblast zájmu předzpracovat. Toho se docílí pomocí dolnoproputního filtru zachovávající hrany. Následně se na obrázek aplikuje kontrastní vy-

lepšení, které následujícím algoritmům pomůže se správným rozpoznáním polygonů. Před samotným určením těchto polygonů je třeba obraz binarizovat a následně v něm opravit možné chyby. Toho se docílí pomocí opravy kontur. Následně se na obraz použijí morfologické operace, které způsobí oddělení kontur v místech, kde při binarizaci vznikl malý spoj. Následně se redukuje počet kontur. Toho se dosáhne pomocí řady filtrů, které byly navrženy tak aby z binarizovaného obrazu odstranily porušené kontury, které by mohly způsobit nepřesné určení markantů. Poté se nalezené markanty převedou do textové reprezentace (šablony) a vypíší na standartní výstup a nebo se provede porovnávání s ostatními šablonami. V případě, že si uživatel zvolil možnost s porovnáváním se postupně prochází složka s šablonami a postupně se porovnávají. Nakonec se na základě procentuální podobnosti šablon určí, který obraz nejlépe odpovídá tomu vstupnímu. Na následující obrázku je ukázka navrženého systému 3.12.



Obrázek 3.12: Navržený systém pro rozpoznávání zvěře na základě fotografie nosu

3.2.1 Získání fotografií nosů

Vstupem do systému jsou fotografie nosů zvěře. Ty jsou získány z experimentální databáze nosů zvěře, která byla vytvořena pro tento účel. Fotky v této databázi byly pořízeny v terénu dobrovolníky po zabití jedince zvěře. Pro každého takového jedince byla pořízena sada fotek. Způsob pořizování fotografií je prozatím neustálený z hlediska směru, osvětlení (používání blesku) a okolních vlivů. Příkladem těchto vlivů může být například vlhkost na čumáku, která při fotografii s bleskem způsobí, že je fotografie téměř nepoužitelná pro rozpoznávání. Dalším takovým vlivem mohou být nečistoty na čumáku jako třeba malé listy případně jiné objekty, které mohou na čumáku zůstat po odstřelu při nedostatečném očištění. Mezi další nejčastější problémy patří nedostatečné zaostření na oblast rhinaria a špatný úhel při pořizování fotografie. Prozatím je v této databázi přes 2000 fotografií a pro účel této práce bylo vybráno 171 fotek, které splňovali autorem určená kritéria. Hlavním kritériem při výběru fotek byla hlavně kvalitně zaostřená a dobře viditelná oblast rhinaria.

Jelikož bylo v zájmu vyzkoušet tento přístup na různé typy fotek, byla tato databáze doplněna o fotky, které nebyly úplně ideální a obsahovaly různé problémy, které byly popsány dříve. Nejčastěji byly fotografie byly pořizovány mobilním telefonem, aby nebylo nutné používat žádné zvláštní zařízení. Z důvodu možného pohoršení čtenáře budou na následujícím obrázku ukázány pouze ořezané fotky možných vstupních fotografií bez zbytku usmrcené zvěře [3.13](#).



Obrázek 3.13: Ořezané fotografie nosů zvěře

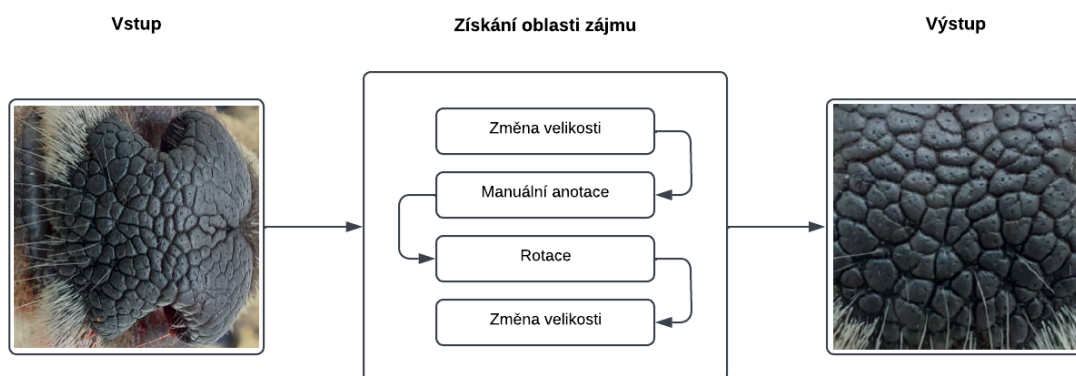
3.2.2 Získání oblasti zájmu

Jelikož jsou fotografie focené z větší vzdálenosti, obsahují kromě oblasti rhinaria i různé nevyžádané prvky. Navíc mají fotografie i různé velikosti, což je pro další postup nevhodné. Pro tento účel je potřeba získat tzv. oblast zájmu. **Oblast zájmu** je část původní fotografie, která obsahuje pouze rhinarium a je již rotačně a velikostně převedena do stavu, který je vhodný pro další postup. Zároveň oblast zájmu představuje i souřadnicový systém, ve kterém se budou nacházet markanty.

Na základě pozorování bylo zjištěno, že vyhovujícími body pro ohraničení této oblasti budou nosní dírky. První myšlenkou pro automatické zajištění těchto bodů byla jejich barva. Teoreticky by nosní dírky měly být nejtmaší. To se ovšem ukázalo jako nepravdivé, jelikož nestejný způsob osvětlení fotografií způsoboval, že nosní dírky nebyly nejtmaší, tudíž bylo od tohoto způsobu upuštěno.

Další variantou by bylo využít strojového/hlubokého učení, aby se tyto body ohraničující nosní dírky určily automaticky. Z důvodu nedostatku dat pro natrénování je tato možnost ovšem velice omezená a nemusela by správně fungovat.

Z těchto důvodů se nakonec zvolila uživatelsky nepohodlná ale zato spolehlivá metoda založená na manuální anotaci nosních dírek. V potaz při anotaci byl brán i horní ret, ovšem po zvolení způsobu ořezání oblasti zájmu se od toho opustilo, jelikož se pozice rtu nijak nevyužívalo. Ukázka návrhu získání oblasti zájmu na základě anotace [3.14](#).



Obrázek 3.14: Návrh získání oblasti zájmu

Ještě před samotnou anotací je potřeba vstupní fotografii zmenšit, aby nebyla příliš velká pro zobrazení uživateli. To se udělá pomocí změny velikosti obrazu v takovém poměru, aby výsledná fotografie měla šířku 600 pixelů a výšku určenou na základě poměru původní šířky vůči 600 pixelům. Díky tomu si vstupní fotografii zachová poměr stran a nebude nijak deformovaná.

Manuální anotace

Jedna nosní dírka se označí celkem třemi body, jelikož je to nejmenší počet bodů pro označení nějaké oblasti. Pozice prvního bodu je v případě levé nosní dírky její nejpravější bod. V případě pravé nosní dírky je to naopak nejlevější bod. Na základě těchto bodů bude možné určit, kde začíná rhinarium a kde končí nosní dírka. Díky tomu jsou tyto body vhodné pro určení šířky výsledné oblasti zájmu. Tyto dva body se budou dále nazývat jako **delt**, jelikož se pomocí nich bude určovat pozice vyextrahovaných rysů, což má jistou podobnost s deltami u otisků prstů. Navíc je možné tyto body nejjednoznačněji určit a tudíž se jeví jako nejvhodnější pro následný postup. Další dva body pro anotaci nosní dírky jsou její nejvrchnější bod a nejspodnější bod. Při anotování je důležité pořadí, v kterém jsou body označeny. Prvně se body označují levá dírka a až poté ta pravá. Bez zachování pořadí by nebylo možné určit, která nosní dírka je pravá a která levá, což by způsobilo špatnou rotaci obrazu. Na následujícím obrázku bude ukázáno, jak se fotografie anotuje [3.15](#).

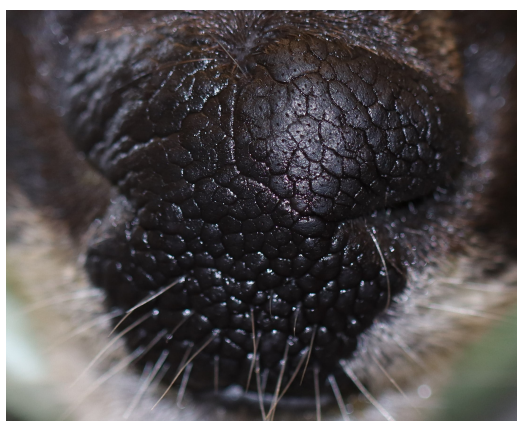


Obrázek 3.15: Anotace nosních dírek

Rotace

Původně bylo v plánu pomocí anotovaných bodů udělat okolo obou nosních dírek vepsané nebo opsané kružnice. Za pomoci středů těchto kružnic by se obrázek otočil, aby byl rotačně normalizován.

Dalším důvodem těchto bodů byla kromě vyříznutí oblasti zájmu transformace perspektivy, která by zajistila, že se fotografie pořízeny pod různými úhly transformují do stavu, který by odpovídal frontální fotografii. Od toho se nakonec upustilo, jelikož nosní dírky občas nebyly vidět celé a tudíž bylo těžké anotovat dírky na správných místech. Například v případě že je fotografie focena více ze strany, je velice obtížné anotovat nosní dírku vrchním a spodním bodem, jelikož nejde pořádně poznat, kde dírka končí [3.16](#).



Obrázek 3.16: Problémová fotografie pro anotaci

Z těchto důvodů se od kružnic opustilo. Rotace se dělá na základě pozic samotných delt. Díky tomu je oblast zájmu orotovaná na základě bodů, které byly potřebné pro její vyříznutí. Výsledek orotované fotografie lze vidět na následujícím obrázku [3.17](#).



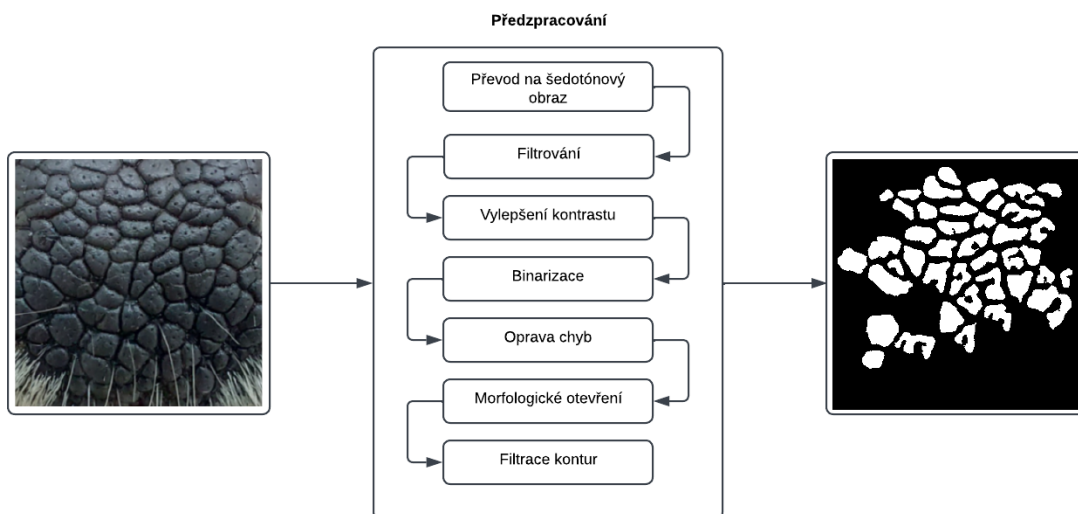
Obrázek 3.17: Orotovaná fotografie na základě delt

Získání oblasti zájmu

Cílem této části bylo získat z fotografie co největší část rhinaria a zároveň zajistit aby výsledná oblast neobsahovala ostatní části fotografie. Z těchto důvodů bylo vyzkoušeno několik útvarů s různým posunutím. Co se týče velikosti oblasti zájmu bylo na základě experimentování dospěno k závěru, že nejlepším útvarem je čtverec, který měl délku strany o vzdálenosti mezi deltami. Dále byl tento čtverec posunut o čtvrtinu délky strany dolů, jelikož pokud byly delty na středu, výsledná oblast neobsahovala pouze rhinarium ale i ostatní části fotografie. Výsledkem této části je oblast zájmu, která bude správně orotovaná a bude mít pevnou velikost 300x300 pixelů. Ukázka výsledné oblasti zájmu je vidět napravo na obrázku 3.14.

3.2.3 Předzpracování

Předzpracování fotografie je proces, který pracuje se získanou oblastí zájmu. Úkolem předzpracování je převedení oblasti do takové podoby, aby byla vhodná pro extrakci rysů. Prvním krokem předzpracování je převod obrazu na jeho šedotónovou variantu. Následně je vhodné obraz vyfiltrovat, aby se v něm redukoval šum. Po filtrování následuje vylepšení kontrastu rýh a polygonů. Vylepšený obraz poté poslouží jako vstup pro binarizaci. Tím vznikne dvouúrovňová reprezentace obrazu. Následně se v obrazu najdou kontury, u kterých dojde k jejich vylepšení. Poté se na obraz s konturami aplikují morfologické operace, aby se oddělily možné spojitosti v místech, kde by být neměly. Na obraz se nakonec aplikuje řada filtrů, které odstraní špatně rozpoznané kontury. Ukázka návrhu předzpracování obrazu se nachází na 3.18.



Obrázek 3.18: Návrh předzpracování oblasti zájmu

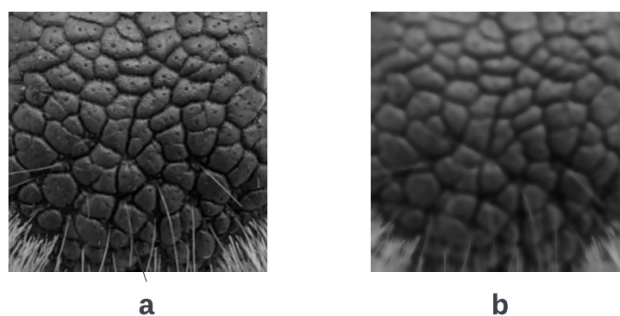
Převod na šedotónový obraz

Tento krok se provádí za účelem snížení výpočetní náročnosti aplikovaných algoritmů. Operace pracuje po pixelech a každý pixel složený ze tří komponent (modrá, zelená, červená) převede na jednu hodnotu intenzity dle vah jednotlivých komponent. Ukázka šedotónového obrazu se nachází na obrázku 3.19 společně s vyfiltrovaným obrazem.

V případě potřeby by bylo v tomto kroku možné určit, zda je obrázek vhodný pro další zpracování. Toho by se dalo dosáhnout pomocí analýzy histogramu daného obrazu. Na základě počtu světlých pixelů by bylo možné určit, jestli se v obraze nenachází příliš odlesku, což téměř znemožní další zpracování [17].

Filtrace obrazu

Pro filtrování je navrženo několik dolnoproputných filtrů jako Gaussův filtr, mediánový filtr, průměrový filtr a bilaterální filtr. Úkolem tohoto filtru bude kromě redukce šumu také co nejlepší zachování původních rýh a polygonů. Na základě těchto kritérií a řadě testování, byl pro tento krok zvolen bilaterální filtr 3.1.1. Ten je sice výpočetně pomalejší, ovšem ve srovnání s ostatními navrženými filtry vykazoval nejlepší výsledky. Na následujícím obrázku je uvedena ukázka aplikovaného filtru na šedotónový obraz 3.19.



Obrázek 3.19: Obrázek před a po filtraci. (a) šedotónový obraz, (b) obraz po aplikaci bilaterálního filtru

Vylepšení kontrastu

Po filtrování se v obrazu vylepší kontrast rýh a polygonů. Toho se dá dosáhnout pomocí histogramové ekvalizace [3.1.2](#), která způsobí, že intenzity pixelů se rozprostřou po celém spektru. Algoritmus zvolený pro tento krok se nazývá CLAHE a jedná se o vylepšenou variantu klasické histogramové ekvalizace. Tento přístup byl zvolen na základě práce o rozpoznávání psů na základě nosu [\[17\]](#), jelikož nosy srnek a jelenů mají barevně nos podobný tomu psímu. Na následujícím obrázku se nachází ukázka před a po vylepšení kontrastu včetně porovnání s klasickou histogramovou ekvalizací [3.20](#).

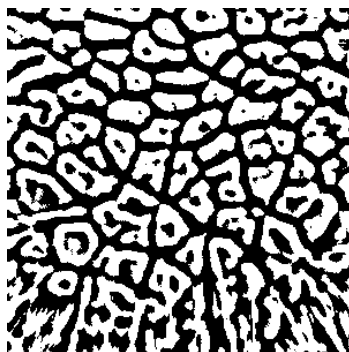


Obrázek 3.20: Aplikace histogramové ekvalizace na obraz. (a) původní obraz, (b) histogramová ekvalizace, (c) CLAHE

Binarizace

Po kontrastním vylepšení bude následovat binarizace. Ta převede šedotónový obraz do černobíle reprezentace. K tomu je vhodné využít tzv. adaptivního prahování [3.1.3](#), které se oproti klasickému prahování liší v tom, že hodnotu prahu určí na základě lokálního okolí pixelu. Toto se pro tento krok velmi hodí a to z důvodu různého osvětlení otisku. Kdyby bylo využito klasického prahování, nedalo by se z důvodu různého osvětlení určit jednu

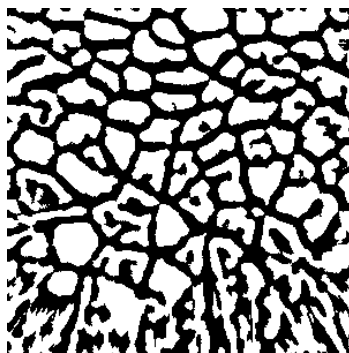
hodnotu prahu, která by byla vhodná pro celý obraz. Výsledkem tohoto bude černý obraz, který bude obsahovat spoustu menších bílých oblastí. Tyto ohraničené oblasti se nazývají kontury a zbytek této práce bude založen převážně na práci s nimi. Ukázka binarizovaného obrazu se nachází na 3.21. Kromě metody adaptivního prahování bylo vyzkoušeno i Otsuovo prahování. To ovšem nevykazovalo příliš vhodné výsledky a proto nebylo nakonec použito.



Obrázek 3.21: Binarizovaný obraz

Oprava chyb

Oprava chyb se zakládá na nalezení kontur v obraze a jejich struktuře uspořádání 3.1.4. V implementaci knihovny openCV je možné nalezené kontury vrátit ve stromové struktuře, v které je možné určit uspořádání kontur. Díky tomu se dá projít všechny kontury a v případě, že se kontura nachází uvnitř jiné se její plocha obarví. Tímto způsobem se opraví díry v konturách vzniklé při binarizaci. Ukázka obrázku po opravě je na obrázku 3.22.

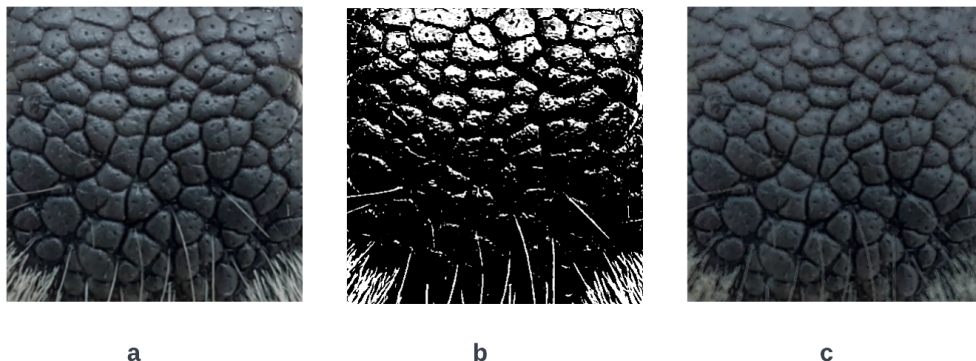


Obrázek 3.22: Opravený binarizovaný obraz

Dále bylo navrženo opravit kontury podle jejich Huulově konvexní aproximace. Od tohoto postupu byl ovšem odstoupeno, jelikož se tím naruší tvar původní kontury, což není vhodné pro finální extrakci pozic markantů.

Dalším navrženým způsobem opravy chyb bylo využití ostřícího jádra, které v obraze detekovalo oblasti s velkým kontrastem. Tímto se daly v některých fotografiích, najít pozice vousků a následně tyto vousky přebarvit na základě jejich okolí. Díky tomu se z původních fotografií odstranily vousky, které způsobovaly problémy při binarizaci. Tento přístup fungoval na některé obrázky, které neobsahovaly žádný odlesk ani žádné jiné vlivy, které by způsobovaly přílišný kontrast. Jelikož se ovšem odlesk v obrázcích vyskytuje poměrně

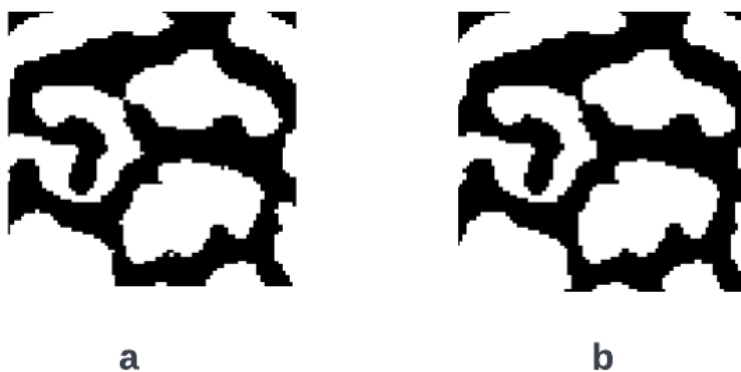
často, bylo od tohoto způsobu upuštěno. Ačkoli se tento postup neaplikuje na binarizovaný obraz, spadá do kategorie oprav a proto je uveden v této části. Na následujícím obrázku se nachází ukázka pokusu o odstranění vousků z obrazu 3.23.



Obrázek 3.23: Odstranění vousků z obrazu. (a) původní obraz, (b) oblasti detekované ostřícím jádrem, (c) obraz po odstranění oblastí

Morfologické otevření

Dále bylo navrženo na obraz aplikovat morfologické otevření 3.1.5 z důvodu oddělení kontur, které po binarizaci zůstaly spojeny krátkým úsekem (nejčastěji se jedná o pár pixelů v kterých se kontury dotýkají). Zároveň se díky této operaci zjemní kraje kontur, což zajistí menší paměťovou náročnost. Operaci je důležité aplikovat na obraz s malým jádrem a malým počtem iterací. V opačném případě se stane, že se deformují kontury, které byly původně v pořádku. Na následujícím obrázku je znázorněno, jak vypadají spojené kontury po aplikaci morfologického otevření 3.24.

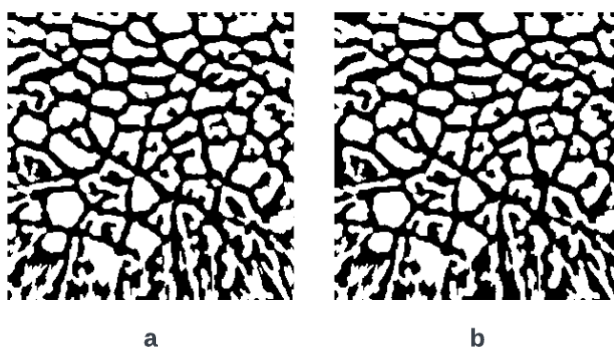


Obrázek 3.24: Rozpojení dvou kontur. (a) původní obraz, (b) obraz po aplikaci morfologického otevření

Filtrace oblastí

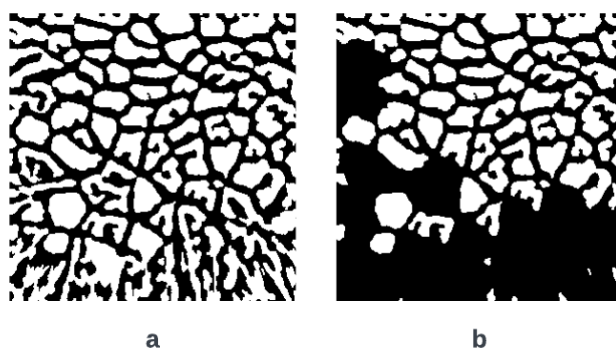
V této části se v obraze opět najdou kontury. Poté se iteruje přes všechny kontury a v případě, že kontura nevyhovuje nějakému ze tří kritérií dojde k jejímu smazání. Ukázka aplikace všech kritérií současně je znázorněna na obrázku 3.18.

Prvním kritériem je velikost plochy kontury. Pro tento účel byly zvoleny pevně dvě hranice, spodní hranice o hodnotě 100 a vrchní hranice 10 000. Tyto hranice byly vybrány na základě předešlé analýzy velikostí kontur v obraze. Kontury s obsahem plochy menší než je dolní hranice u testovaných obrázků nepředstavovaly hledané polygony, ale spíše ostatní objekty. Kontury s obsahem který je větší než horní hranice reprezentovaly špatně nalezené kontury, které se sloučily dohromady. Sloučení těchto kontur se dělo hlavně z důvodu, že se na původním obrázku nacházel například příliš velký odlesk nebo vousky. Ukázka odstranění těchto částí je na obrázku 3.25.



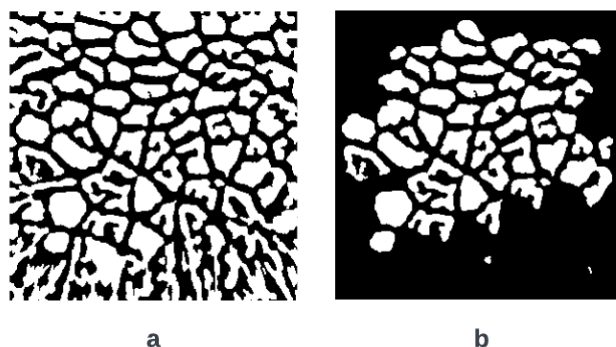
Obrázek 3.25: Odstranění příliš malých/velkých kontur. (a) původní obraz, (b) obraz po filtraci

Druhé kritérium filtrace využívá pevnosti kontury. Po testování byla zvolena hranice pevnosti na 60%. Tato hranice se zdála jako ideální, jelikož se v obraze odstranily příliš rozbité kontury a zároveň se zachovaly kontury, které se rozpoznaly dobře. Hranice by se určitě dala navýšit, jelikož ale kontury nebývají v převážné většině případů ideálně určené, není to v aktuálním stádiu vhodné. Na obrázku níže se nachází ukázka aplikace tohoto kritéria při filtrování 3.26.



Obrázek 3.26: Odstranění kontur na základě pevnosti. (a) původní obraz, (b) obraz po filtraci

Třetí kritérium se zabývá pozicí kontury v obraze. To využívá souřadnic bodů, které kontury ohraničují. V případě, že se kontura nachází na hranici obrazu se kontura smaže. To se dělá z důvodu, že kontury na kraji nejsou v obraze celé, což by mohlo způsobit, že se v těchto konturách určí nesprávně pozice markantů (středů). Na následujícím obrázku se nachází oblast redukovaná o hraniční kontury [3.27](#).



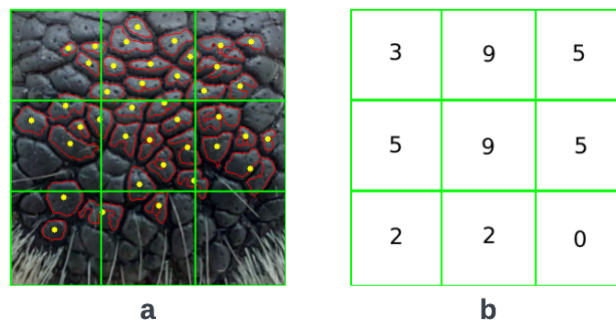
Obrázek 3.27: Odstranění kontur podle umístění. (a) původní obraz, (b) obraz po filtraci

Dalším uvažovaným kritériem pro filtraci byla velikost nalezených kontur. Myšlenkou této filtrace bylo na základě velikostí ploch kontur v obraze určit průměrnou plochu kontury a poté odstranit všechny kontury v obraze, které by se svou plochou lišily o více než zvolenou odchylku. Tento způsob se ovšem později ukázal jako neproveditelný, jelikož se rozložení velikostí ploch kontur v různých obrazech výrazně lišily a bylo obtížné tuto odchylku jednoznačně určit. Na základě zkoumání velikostí nalezených kontur při testování této myšlenky vzniklo první kritérium.

3.2.4 Extrakce markantů

Extrakce markantů je proces, který ze vstupu vygeneruje šablonu rysů. Tuto šablonu bude představovat dvoudimenzionální pole, kde v každá buňka bude reprezentovat malou oblast

z původního obrazu. Například pokud má původní obraz rozměry 300x300 pixelů a pole bude mít velikost 10x10, bude každá buňka pole reprezentovat oblast o velikosti 30x30 pixelů. Následně je potřeba v obraze zjistit pozice markantů. Toho se dosáhne určením pozice středů v jednotlivých konturách pomocí jejich momentů. Poté se tyto pozice převedou do předpřipraveného pole, přičemž pozice v poli bude odpovídat původním pozicím markantů v obraze. Dále je vhodné aby velikost pole nebyla příliš velká, jelikož je nutné aby, mírná změna při anotaci nezpůsobila, že se pozice markantů připíše do jiných bloků. Na následujícím obrázku se nachází ukázka vizualizace tohoto kroku 3.28.



Obrázek 3.28: Převod nalezených markantů na reprezentaci polem. (a) původní obraz s nalezenými markanty, (b) vizualizace pole s počtem markantů v jednotlivých buňkách

Během vývoje bylo navrženo při extrakci využít pouze pár nejvýznamnějších kontur. Toho by se dosáhlo například pomocí vybrání počtu největších kontur, z kterých by se provedla extrakce. Od této varianty bylo ale nakonec odstoupeno, jelikož se po testování ukázalo, že největší kontury většinou nebývají nejlépe určené. Na tomto nápadu se ale dále pracovalo a vzniklo z něj navržené filtrování kontur, jehož cílem je zanechat pouze ty nejlépe určené kontury.

Návrhem pro vylepšení této metody by bylo kromě pozic markantů také extrahovat jejich orientaci jako tomu bylo v práci [30]. Tímto způsobem by se měl snížit počet milných shod při porovnání, ovšem musel se změnit formát prvků v poli tak, aby kromě hodnot obsahoval i úhly korespondující k daným bodům. Jelikož se zatím jedná o první pokus o identifikaci, bude prozatím testována pouze základní varianta tohoto způsobu.

3.2.5 Nalezení nejlepší shody

Tento krok je nepovinný a záleží na uživateli, zda jej bude vyžadovat a nebo pouze skončí u výpisu extrahované šablony. Pro nalezení nejlepší shody je potřeba, aby uživatel zadal složku, v které se nachází ostatní šablony společně s fotografiemi ke kterým se šablony vážou. Navržený algoritmus poté ve vybrané složce projde všechny nalezené šablony a porovnává je s aktuálně vyextrahovanou šablonou. Po projití celé databáze se už pouze vybere šablona z databáze, s kterou se vstupní otisk shodoval nejvíce a fotografie, která k dané šabloně patří se zobrazí uživateli.

Kapitola 4

Implementace

Tato kapitola se věnuje implementaci návrhu, který byl prezentován v předchozí kapitole. Implementované algoritmy jsou rozděleny do dvou typů. Prvním typem jsou algoritmy, které byly převzaty z knihovny openCV. U těchto algoritmů bude provedeno porovnání konkrétních výstupů ve formě obrazů, které vznikly v závislosti na zvolených parametrech. Druhým typem jsou algoritmy, které bylo nutné navrhnout, aby bylo možné dosáhnout výsledků, které byly stanoveny v návrhu implementace. Příkladem takových algoritmů jsou rotace, změna velikosti a určení markantů. V této části budou podrobně popsány postupy, které byly použity při implementaci těchto algoritmů.

4.1 Použité technologie

Výsledný program byl implementován v jazyce Python verze 3.10.11 s pomocí několika knihoven. Stěžejní knihovnou využitou pro implementaci bylo openCV verze 4.7.0. Mezi další využití knihovny patří argparse, math a numpy. Celý program se nachází v jednom souboru a je členěn do jednotlivých funkcí. Program je možné spustit s dvěma argumenty. První argument je povinný a reprezentuje cestu ke vstupní fotografii (parametr `-i/-input`). Druhý nepovinný argument určuje, zda se bude vstupní fotografie porovnávat a zároveň určuje cestu ke složce v níž se nachází šablony s kterými chce uživatel vstupní fotografii porovnat (parametr `-c/-cmpFile`).

OpenCV

OpenCV je multiplatformní volně dostupná otevřená knihovna, která umožňuje přístup k algoritmům spojeným s počítačovým viděním. Původně byla tato knihovna vyvíjena společností intel za účelem zrychlení pokroku v oblasti rozpoznávání obrazu. Díky jeho licenci je totiž možné knihovnu využít i v komerčních aplikacích, čímž by postupně vzrostl zájem o rychlejší procesory. V dnešní době se vývoj hodně odklonil od původního intelu a do otevřeného kódu spíše přispívají jedinci, kteří knihovnu využívají. Navíc také slouží jako základní stavební kámen pro projekty spojené s počítačovým viděním a díky optimalizovanosti algoritmů není nutné vymýšlet základní operace a je možné na nich pouze stavět. Z tohoto důvodu je výsledný kód lépe čitelný a přenosný. Původně byla tato knihovna napsána v jazycích C / C++, avšak v dnešní době již podporuje i spoustu jiných jazyků jako Python a Java. Knihovna je strukturovaná do několika modulů, jako jsou například zpracovávání obrazu, detekce objektů, analýza videa a další viz [31]. V aktuální verzi openCV 4.7.0 je k

dispozici přes 2500 algoritmů. Pro tuto práci bude důležitá hlavně verze openCV-Python, která slouží jako API původní knihovny openCV pro python [8].

4.2 Načítání fotografie

Pro načtení fotografie do programu je potřeba zadat správnou cestu k fotografii. Pokud cesta k fotografii není správná, program skončí s chybou. Načítání se provádí v hlavní části programu pomocí funkce `imread()`.

Následně se fotografii ve funkci `rescaleImage()` změní velikost. Vstupem do této funkce je parametr představující hodnotu nové šířky, pomocí kterého se vypočítá hodnota měřítka. Toto měřítko se poté použije pro transformaci rozměrů fotografie. Ukázka výpočtu tohoto měřítka se nachází v následující rovnici 4.1.

$$\text{měřítko} = \frac{\text{šířka}_{\text{nová}}}{\text{šířka}_{\text{původní}}} \quad (4.1)$$

Nové rozměry fotografie se poté vypočítají následovně:

$$\text{šířka} = \lfloor \text{šířka}_{\text{původní}} * \text{měřítko} \rfloor \quad (4.2)$$

$$\text{výška} = \lfloor \text{výška}_{\text{původní}} * \text{měřítko} \rfloor \quad (4.3)$$

Výsledkem toho je fotografie, která má šířku o velikosti zadané parametrem a výšku o velikosti úměrné vůči změně původní šířky fotografie. V aktuální implementaci se používá hodnota parametru 600, jelikož se jevila jako ideální velikost pro následné zobrazení uživateli.

4.2.1 Anotace fotografie

Následně se uživateli zobrazí zvolená fotografie. Děje se tomu tak díky zavolání funkce `makeAnnotations()`. Funkce následně čeká na anotaci bodu uživatelem. Ten tak učiní pomocí dvojitého kliknutí na pixel na obrázku. Reakcí na tento klik je zavolání funkce `drawCircle()`, která na fotografii vykreslí kruh, kterým uživatele informuje o pozici výběru bodu. Následně má uživatel možnost tento bod potvrdit pomocí stisku klávesy **a**. V případě, že uživatel není spokojen s výběrem bodu, může pouze označit jiný bod a poté opět potvrdit výběr. Pokud by uživatel chtěl znovu vybrat již potvrzený bod, může tak učinit pomocí stisku klávesy **r**. Tímto celý proces začne od začátku. Po anotaci potřebných bodů už stačí pouze potvrdit výběr pomocí klávesy **d**. Pozice vybraných bodů jsou následně uloženy pro další postup. Důležité je při anotování brát v úvahu pořadí označení bodů. Prvně se označuje levá a poté pravá dírka.

4.2.2 Rotace

Poté je potřeba obraz převést do vhodného formátu. Toho se učiní pomocí zavolání funkce `rotateImage()`. Tato funkce má za úkol rotačně normalizovat obraz a zároveň vrací rotační matici, která se použije pro transformaci pozic anotovaných bodů.

Pro vytvoření matice rotace je potřeba zjistit úhel o který se má obrázek orotovat a bod okolo kterého se rotace provede. Souřadnice rotačního bodu jsou vypočítány jako průměr souřadnic delt. Pro výpočet úhlu bylo využito funkce `atan2()`, která je definována následovně:

$$\text{atan2}(y, x) = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{je-li } (x > 0) \wedge (y \geq 0), \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{je-li } (x < 0), \\ \arctan\left(\frac{y}{x}\right) + 2\pi & \text{je-li } (x > 0) \wedge (y < 0), \\ \frac{\pi}{2} & \text{je-li } (x = 0) \wedge (y > 0), \\ \frac{3}{2}\pi & \text{je-li } (x = 0) \wedge (y < 0) \end{cases} \quad (4.4)$$

Funkce řeší dva problémy, které mohou nastat při použití klasických funkcí tangens a arkustangens. Tyto problémy zahrnují dělení nulou, což může vést k hodnotám ± 180 , a také návratovou hodnotu, která se nachází v intervalu $(0, 360)$, zatímco klasické funkce tangens a arkustangens vrací hodnoty v intervalu $(-90, 90)$.

Pro výpočet úhlu je potřeba brát v úvahu vzájemné pozice delt, jelikož se výpočet úhlu liší v závislosti na jejich souřadnicích. Výpočet se tedy rozděluje na následující 4 případy (delta je reprezentována souřadnicemi $[x_n, y_n]$ a x, y reprezentují vzálenost mezi deltami po x-ové a y-ové ose):

$$\text{úhel} = \begin{cases} \text{atan2}(x, y) + \pi & \text{je-li } (x_1 > x_2) \wedge (y_1 > y_2), \\ -\text{atan2}(x, y) + \pi & \text{je-li } (x_1 > x_2) \wedge (y_1 < y_2), \\ -\text{atan2}(x, y) & \text{je-li } (x_1 < x_2) \wedge (y_1 > y_2), \\ \text{atan2}(x, y) & \text{je-li } (x_1 < x_2) \wedge (y_1 < y_2) \end{cases} \quad (4.5)$$

Po zjištění bodu rotace a úhlu stačí pomocí funkce `getRotationMatrix2D()` vytvořit rotační matici. Tato matice se následně aplikuje na vstupní obraz, čímž se obraz orotuje vůči pozicím delt.

4.2.3 Získání oblasti zájmu

Po orotování obrazu je potřeba z obrazu získat oblast zájmu. K tomu je zapotřebí získat nové souřadnice delt po rotaci pomocí funkce `getNewCoordinates()`. Tato funkce za pomoci matice použité při rotaci a funkce `transform()` určí nové souřadnice delt.

Následně se z obrazu získá oblast zájmu. To se děje ve funkci `cropRectangle()`, která pomocí vzdálenosti mezi deltami vyřízne z obrázku čtverec představující oblast zájmu.

Nakonec už se pouze pomocí funkce `rescaleImage()` změní velikost obrazu na 300x300 pixelů. Díky těmto krokům budou všechny fotografie ve stejném formátu pro následné zpracování.

4.3 Předzpracování a extrakce markantů

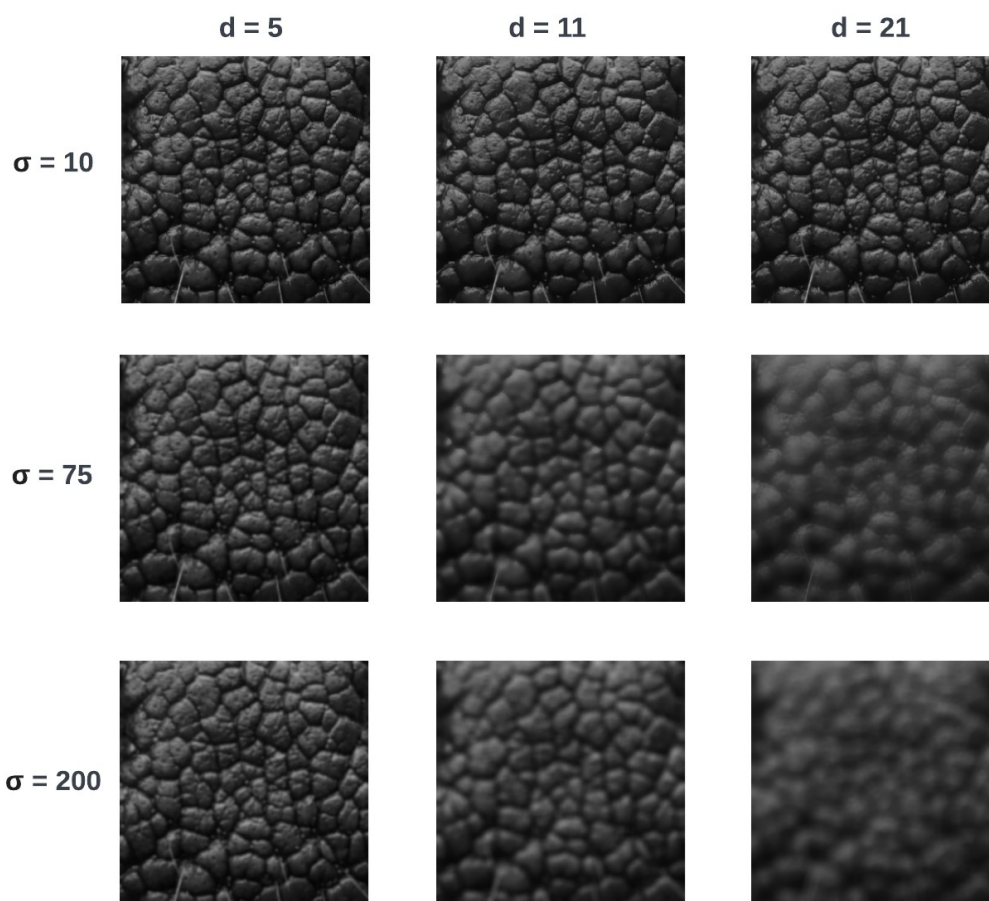
Tato podkapitola se bude věnovat aplikacím algoritmů z návrhu aplikace. Jelikož se většinou jedná o algoritmy, které byly již popsány dříve budou se v této části zkoumat různé parametry navržených algoritmů. V případě, že tyto algoritmy ještě nebyly pořádně vysvětleny jako třeba oprava chyb a filtrace kontur, budou náležitě popsány.

4.3.1 Filtrování

Pro filtrování obrazu se využívá bilaterálního filtrování. Tento filtr se aplikuje pomocí funkce `bilateralFilter()`. Funkce má možnost nastavení tří parametrů. První parametr představuje diametr (průměr) okolí pixelu, které se bude uvažovat pro výpočet nové hodnoty. Druhým parametrem je `sigmaColor`. Tento parametr reprezentuje odchylku barev, které budou

brány v potaz. V případě šedotónového obrazu to jsou rozdíly intenzit. Třetím parametrem je `sigmaSpace`, která reprezentuje vzdálenost bodů, které budou použity pro filtrování. Na následujícím obrázku je možné vidět rozdíly mezi nastavením těchto parametrů 4.1.

- d = uvažované okolí pixelu
- σ = `sigmaColor` a `sigmaSpace`



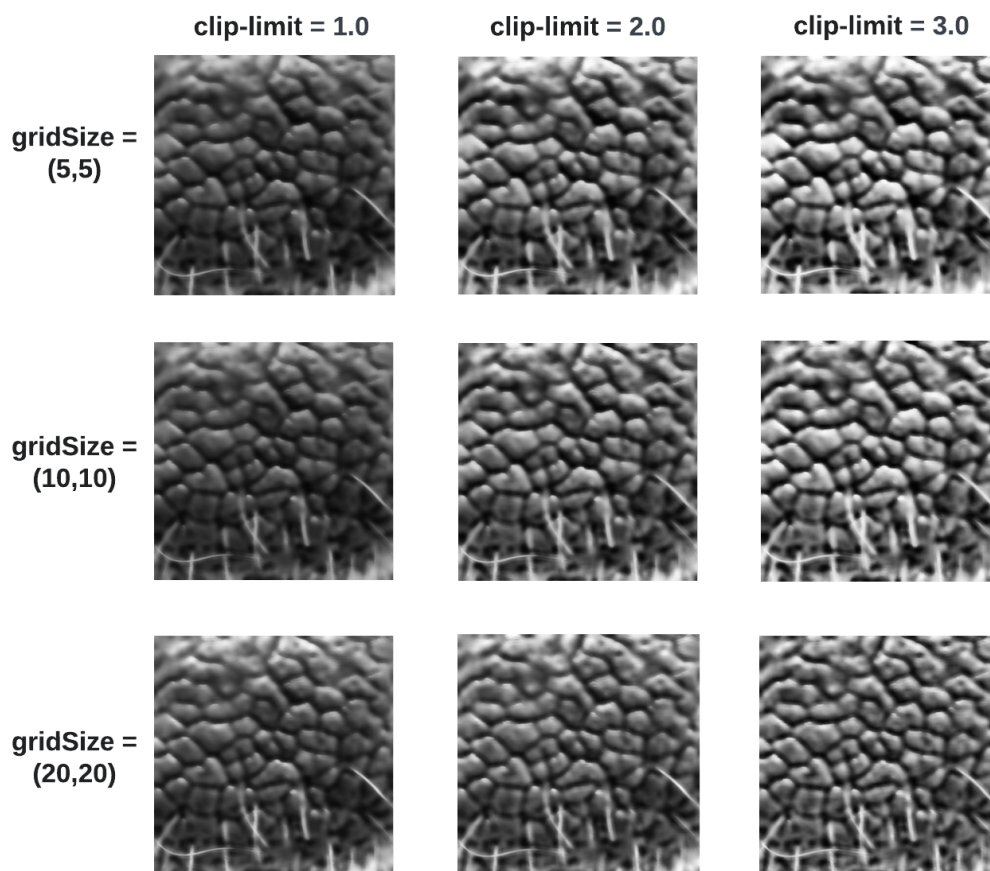
Obrázek 4.1: Použití různých parametrů bilaterálního filtru

Z obrázku lze pozorovat, že v případě nastavení malé hodnoty diametru nebo malé hodnoty σ , se obraz prakticky nezmění. Naopak v případě, že je diametr příliš velký, je výsledkem příliš rozmazaný obraz, v kterém je obtížné rozlišit polygon a rýhy. Na základě pozorování aplikace filtru na obrázky v databázi bylo určeno, že ideální hodnota diametru je 11 a hodnota σ je 75.

4.3.2 Vylepšení kontrastu

Pro vylepšení kontrastu obrazu se používá funkce `makeClahe()`. V této funkci se pomocí dvou parametrů nastavuje proces histogramové ekvalizace. Prvním parametrem je `clip-limit`, kterému se již věnoval text v navržených algoritmech 3.2. Druhým parametrem je `gridSize`,

který určuje, do kolika částí se obraz rozdělí. Poté se pro každou část udělá vyrovnání histogramu. Na následujícím obrázku bude ukázán vliv těchto parametrů na šedotónový obraz 4.2.

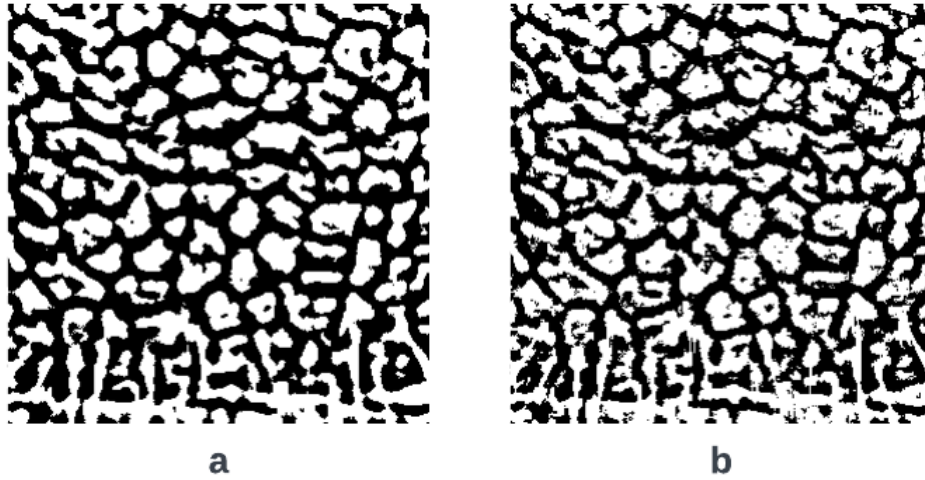


Obrázek 4.2: Použití různých parametrů CLAHE

Z obrázku lze vidět, že čím vyšší hodnotu clip-limit má, tím více se zvýrazňují světlé části obrazu. Nejlépe si po vizuálním testování více obrazů vedla hodnota clip-limit 2. Tato hodnota byla nejlepší, jelikož hodnota 1 měla minimální efekt na výsledný obraz a hodnota 3 až moc zvýrazňovala světlá místa na obrazu. Hodnota gridSize neměla příliš velký vliv na výsledek ekvalizace a následkem testování byla jeho hodnota určena na (10,10).

4.3.3 Binarizace

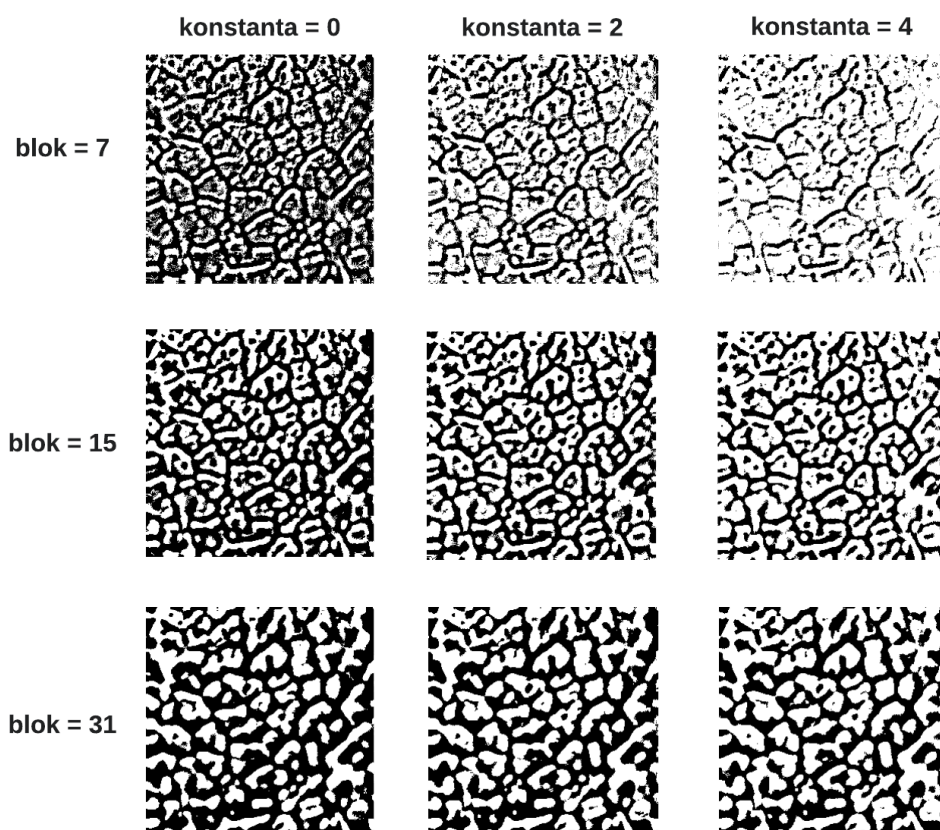
K binarizaci obrazu se používá funkce `adaptiveThreshold()`. Tato funkce má tři zajímavé parametry, které lze nastavit. Jedním z těchto parametrů je způsob pro výpočet hodnoty práhu pro daný blok. V aktuální implementaci openCV jsou dostupné dvě možnosti, průměr hodnot v bloku nebo hodnota určená váženým průměrem s Gaussovým jádrem. Pro srovnání těchto dvou způsobů bude na následujícím obrázku ukázán jejich rozdíl 4.3.



Obrázek 4.3: Použití různých způsobů pro určení hodnoty prahu, (a) průměr, (b) Gaussovo jádro

Na obrázku lze vidět, že určení prahu na základě průměru hodnot v bloku produkuje značně čistější binarizovaný obraz. Tento jev se ukazuje téměř na všech obrázcích v databázi a právě z tohoto důvodu byl způsob binarizace pomocí průměru hodnot zvolen jako ten lepší.

Dalšími dvěma nastavitelnými hodnotami při binarizaci jsou velikost bloku a také konstantní hodnota, která se odečítá od průměru nebo váženého průměru bloku. Na následujícím obrázku lze vidět rozdíly mezi použitím odlišných hodnot [4.4](#).



Obrázek 4.4: Použití různých parametrů adaptivního prahování

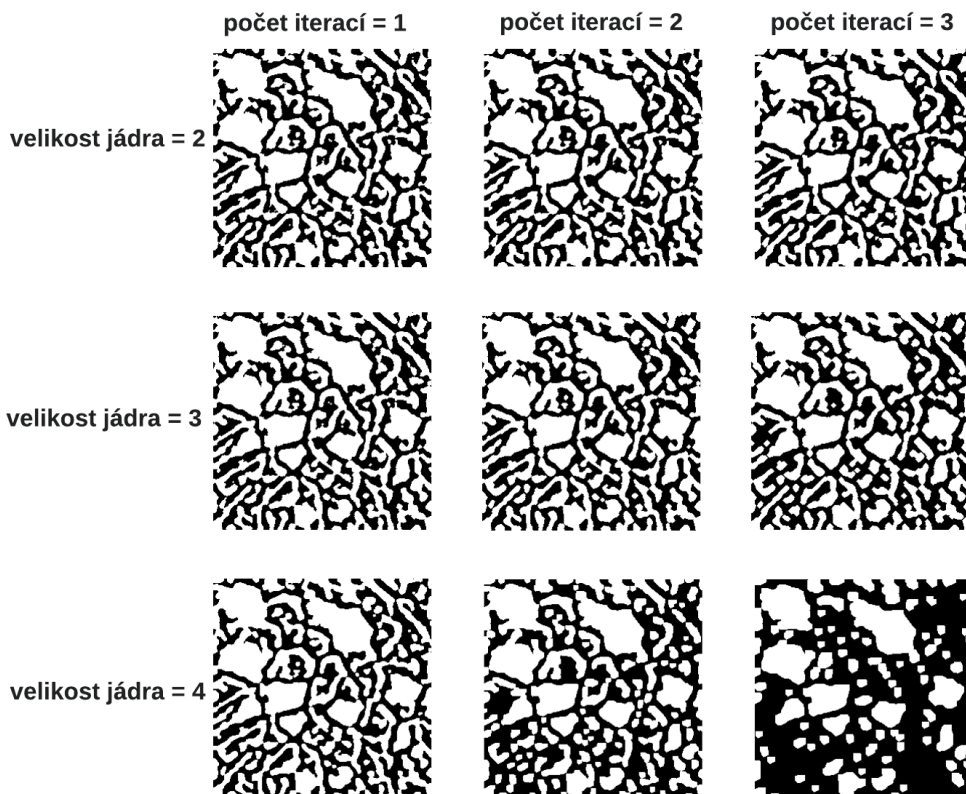
Z obrázku lze vidět, že čím větší konstanta se odečítá, tím více se pixely obarví. S rostoucí velikostí bloku klesá přesnost určení původních bloků, pokud je ale okolí příliš malé, výsledný obraz je téměř nepoužitelný. Na základě testování různých hodnot byla pro finální implementaci zvolena hodnota velikosti bloku 15 a hodnota konstanty 2.

4.3.4 Oprava chyb

Oprava chyb formou vyplnění děravých kontur se nachází ve funkci `getFilled()`. Tato funkce si jako vstup bere binarizovaný obraz. Následně se pomocí funkce `findContours()` vrací kontury ve stromové struktuře. Poté se tato struktura prochází a u každé kontury se rozhoduje, zda se jedná o externí nebo interní konturu. Toho se dosáhne pomocí přístoupení k hierarchii na indexu 3. V případě, že je hodnota tohoto indexu záporná se jedná o venkovní konturu a tudíž se použije funkce `fillPolly()`, která v binarizovaném obraze vyplní konturu bílými pixely. Občas se stane, že tato oprava vybarví značně větší oblasti, než je vyžadováno, jelikož jsou kontury ohraničeny vousky nebo obraz není dostatečně zaostřený. Ovšem pro většinu obrazů, které jsou dobře zaostřeny a mají pěkně rozpoznatelné rozdíly mezi polygony a rýhami je tento způsob vhodný.

4.3.5 Morfologické otevření

Morfologické otevření se provádí ve funkci `getErodedDilated()`. Ve funkci se prvně vytvoří jádro pomocí funkce `getStructuringElement()` a následně se pomocí funkcí `erode()` a `dilate()` provede morfologické otevření. Mezi nastavitelné parametry v tomto procesu patří tvar strukturního elementu (jádra), jeho velikost a počet iterací aplikace dilatace a eroze. Na následujícím obrázku budou ukázány rozdíly mezi různými nastaveními počtu iterací a velikosti jádra 4.5. Při ukázce bylo využito kruhového strukturního elementu.



Obrázek 4.5: Použití různých parametrů morfologického otevření

Z obrázku lze vidět, že počet iterací má velký vliv na výsledný obraz, z tohoto důvodu se vyloučilo využívání více jak 2 iterací. Malá velikost jádra při nízkém počtu iterací nepřináší požadované změny do obrazu a proto také nebyla zvolena jako vhodná varianta. Dostatečně dobře se vedla velikost jádra 4 s jednou iterací, nebo také velikost jádra 2 s dvěma iteracemi. Po delším pozorování se ukázalo, že si s rušením malých spojů a zároveň zachováním původních tvarů vedla varianta s velikostí jádra 4 a jednou iterací.

Co se týče tvaru strukturního elementu byl zvolen právě výše uvedený kruh. Při použití jiných útvarů byly vzhledem k malému počtu iterací a malé velikosti jádra pouze zanedbatelné rozdíly.

4.3.6 Filtrace kontur

Filtrace kontur se nachází ve funkci `getFiltered()`. Tato funkce vezme vstupní obrázek a poté pomocí funkce `findContours()` vyhledá kontury, které se uloží ve formě seznamu.

Poté se tímto seznamem postupně prochází kontury a testuje se, zda kontura vyhovuje kritériím.

První kritérium velikosti využívá funkce `contourArea()` a kontroluje, zda tato funkce vrací hodnotu mezi 100 a 10 000.

Druhé kritérium týkající se Huulovy aproximace využívá funkce `convexHuul()` pro převedení kontury do její konvexní podoby. Poté se pomocí funkce `contourArea()` ověří, že obsah plochy původní kontury není menší než 60% obsahu plochy konvexní kontury.

Třetí kritérium využívá funkce `boundingRect()`, která kolem kontury udělá ohraničující čtyřúhelník. Tato funkce vrací x-ovou a y-ovou souřadnici společně s výškou a šířkou ohraničujícího okna. Poté se pouze kontroluje zda se některý z vrcholů čtyřúhelníku nenachází na okraji obrazu.

4.3.7 Extrakce markantů

Extrakce markantů se nachází ve funkci `getCentroids()`. Ve funkci se opět najdou kontury, které se postupně prochází. Pro každou konturu se funkcí `moments()` získají momenty kontury. Následně se pomocí vzorce 3.4 získá střed kontury, který se poté uloží do předpřipraveného pole. Kromě ukládání pozic středů kontur (markantů) se zde také pomocí funkcí `drawContours()` a `circle()` vykreslují pozice markantů a kontur do původní fotografie. Výstupem funkce je poté obraz se znázorněnými konturami a markanty a také pole, v kterém jsou uloženy pozice markantů.

Jelikož jsou tyto pozice reprezentovány x-ovými a y-ovými souřadnicemi, nejsou vhodné pro další postup. Z tohoto důvodu se zavolá funkce `getPositionArray()`, která převede původní list na upravený list, který byl uveden v návrhu. Upravený list použitý v implementaci má rozměry 30x30.

4.4 Porovnávání šablon a export informací

V případě, že uživatel zvolil možnost s porovnáváním se zavolá funkce `findBestMatch()`. V této funkci se poté otevře složka zadaná v argumentu při spouštění. V této složce se poté prochází všechny soubory a v případě, že soubor je v textovém formátu (jedná se o šablonu) se obsah tohoto souboru uloží do listu. Tento list se poté převede do stejného formátu jako je list s markanty nalezenými při extrakci. Poté se zavolá funkce `comapreArray()`, která tyto dvě pole porovnává.

Porovnávání je implementováno pomocí průchodu přes každou buňku obou listů. V každé buňce se do proměnné uchováující celkový počet markantů přičte maximum z hodnot listů v aktuální buňce. Kromě toho se zde také do druhé proměnné představující shodný počet markantů v aktuální buňce přičítá minimum z hodnot.

Tyto hodnoty se poté vrací do původní funkce, kde se vypočítá procentuální shodnost dvou šablon. Pokud se jedná o dosavadně nejvyšší hodnotu, uloží se společně s názvem souboru, který dosáhl nejlepší shody.

Po dokončení procházení všech souborů funkce vrátí cestu k fotografii s nejlepší shodou společně s procentuální shodností. Nakonec se v hlavní části programu pomocí funkce `imshow()` zobrazí fotografie, s kterou se vstupní nejvíce shodovala společně se vstupní fotografií. V popisu zobrazené fotografie se nachází vypočítaná shodnost a název této fotografie.

V případě, že si uživatel nezvolil možnost s porovnáváním se pouze vypíše vyextrahovaná šablona v textovém formátu na standardní výstup.

Kapitola 5

Testování a vyhodnocení

Tato kapitola se zaměřuje na posouzení implementovaného postupu pro extrakci markantů. V její úvodní části je uveden obecný popis databáze a způsob jejího vytvoření. Dále je zde popsán postup, pomocí kterého se určovala správnost extrakce markantů, včetně nejčastějších problémů, které se při extrakci vyskytovaly. V této části jsou rovněž prezentovány grafy, které vizualizují přesnost implementovaného přístupu. Nakonec je zde uveden stručný text, v němž je popsána problematika implementovaného algoritmu pro nalezení nejlepší shody (rozpoznávání).

5.1 Databáze

Databáze je tvořena množinou dvojic souborů. Tyto dvojice jsou od sebe odlišitelné názvem a jsou tvořeny fotografií nosu a šablonou s markanty v textové podobě. Při vytváření databáze bylo využito tří separátních programů z důvodu lepšího hromadného zpracování. Jeden z těchto programů sloužil k hromadné anotaci fotek a fungoval stejně jako funkce pro anotaci ve finální verzi programu s výjimkou toho, že označené body se neukládaly do proměnné ale do souboru. Následně byly tyto body načteny druhým programem a ten na základě zadaných bodů vyřízl oblast zájmu. Třetí program poté aplikoval stejnou posloupnost operací jako byly popsány v implementaci. Výsledky extrakce byly poté uloženy do šablony se stejným názvem jako původní fotografie. Celkově se v této databázi nachází 171 fotografií.

5.2 Přesnost extrakce markantů

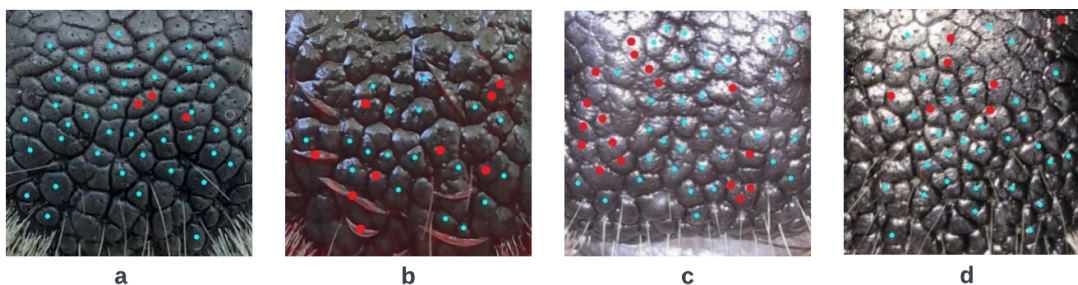
Ještě před samotným určením přesnosti extrakce je potřeba vědět, jakým způsobem se tato přesnost určovala.

Přesnost byla definována jako počet správně určených markantů vůči celkovému počtu extrahovaných markantů. Aby bylo možné určit, co je špatný a co dobrý výsledek určení markantu, bylo potřeba projít všechny výsledky extrakce a v těch ručně vyznačit, zda se jedná o správně určený markant či nikoli.

Správnost tohoto určení se dělala na základě vizuální kontroly. V případě že pozice markantu odpovídala středu polygonu, bylo určení považováno za správné. Pokud nebylo z fotografie zcela zřejmé, zda se markant správně identifikoval, rozhodnutí bylo nakloněno k výsledku, který byl stanoven pomocí aplikovaného algoritmu.

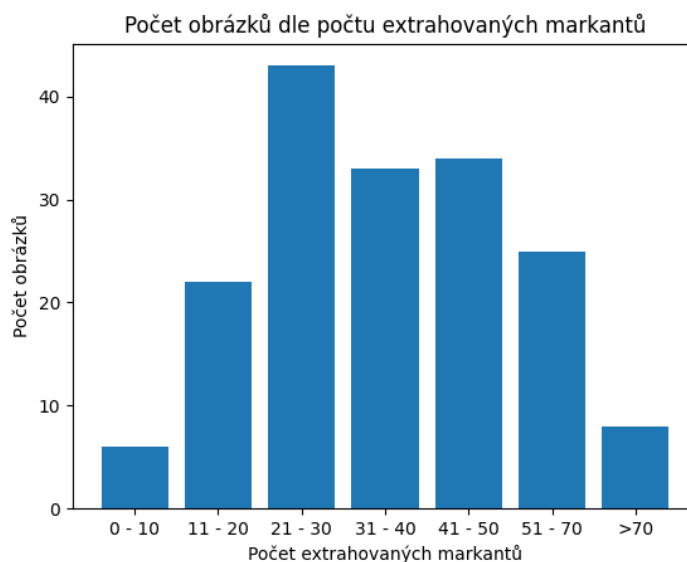
Mezi nejčastějšími problémy při určování markantů se vyskytovaly následující:

- Rozdělení kontury do dvou, což způsobilo určení dvou markantů místo jednoho viz 5.1 (a)
- Určení markantů v místech, kde se nacházelo ochlupení viz 5.1 (b)
- Určení pozice markantu ve středu oblasti odlesku místo středu polygonu viz 5.1 (c)
- Určení markantu v konturách na okraji obrazu (tyto kontury by měly být odstraněny pomocí implementovaného filtru) viz 5.1 (d)



Obrázek 5.1: Nejčastější problémy při určování markantů. Červeně jsou označeny špatně určené markanty. Modře jsou označeny správně určené markanty.

Na grafu 5.2 je možné vidět fotografie rozdělené do skupin podle počtu extrahovaných markantů. Z tohoto grafu vyčíst, že průměrný počet extrahovaných markantů činí zhruba 40 markantů.



Obrázek 5.2: Graf reprezentující určení markantů

Fotografie byly poté rozděleny do skupin podle počtu extrahovaných markantů jako je znázorněno na grafu výše. Pro tyto skupiny byla následně určena úspěšnost extrakce, která

se vypočítala jako celkový počet úspěšně určených markantů ve skupině fotografií podělena celkovým počtem markantů ve skupině.

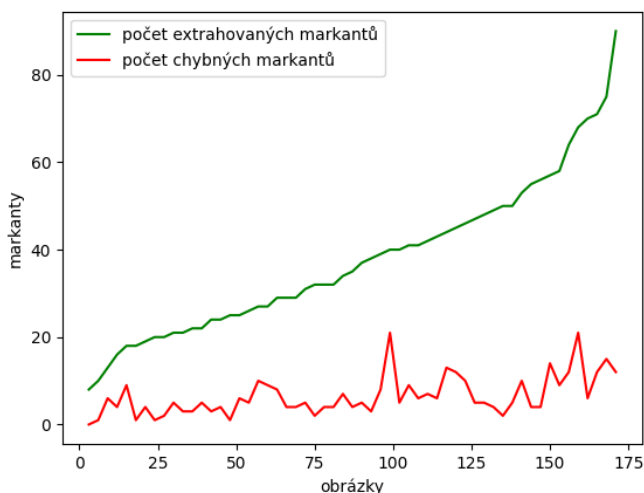
Výsledky pro jednotlivé skupiny byly následující:

$$\text{Přesnost} = \begin{cases} 85.1\% & \text{počet markantů} \in (0, 10) \\ 78.96\% & \text{počet markantů} \in (11, 20) \\ 79.42\% & \text{počet markantů} \in (21, 30) \\ 85.45\% & \text{počet markantů} \in (31, 40) \\ 82.35\% & \text{počet markantů} \in (41, 50) \\ 82.32\% & \text{počet markantů} \in (51, 70) \\ 87.33\% & \text{počet markantů} \in (70, \infty) \end{cases} \quad (5.1)$$

Celková přesnost extrakce byla poté vypočítána stejně jako úspěšnost extrakce jednotlivých skupin a činila 82,71%.

Souvislost chybovosti na počtu extrahovaných markantů

Pro další testování byly vstupní fotografie seřazeny podle počtu extrahovaných markantů. Následně byl celkový objem fotografií nevzorkován, aby byl výsledný graf lépe čitelný.



Obrázek 5.3: Graf reprezentující souvislost chybovosti na počtu extrahovaných markantů

Z grafu 5.3 lze vyčíst, že počet chyb v obrázku nijak nesouvisí s počtem extrahovaných markantů. Chyby se vyskytují náhodně a jsou způsobeny ostatními vlivy zmíněnými dříve.

5.3 Úspěšnost rozpoznávání

Pokud jde o úspěšnost rozpoznávání, stanovení konkrétního procentuálního úspěchu je velmi obtížné, protože zarovnání šablony závisí na deltách, které jsou určeny manuální anotací uživatelem. To znamená, že se nalezené markanty mohou nacházet na různých místech. Z tohoto důvodu by bylo vhodné pro porovnávání dvou šablon s markanty prozkoumat další metody, které by mohly vést k lepším výsledkům.

Kapitola 6

Závěr

Tato bakalářská práce měla za úkol vytvořit aplikaci pro zpracování obrazu a extrakci vlastností za účelem vyhledávání markantů pro identifikaci zvěře. Vzhledem k tomu, že se jednalo o první práci tohoto druhu, byla tato problematika zkoumána pouze na nosech zvěře typu jelen a srnka.

Pro vytvoření zmíněné aplikace bylo nutné se seznámit s oborem biometrie a oblastí zaměřenou na rozpoznávání lidí na základě otisků prstů. Studované práce měly hlavní přínos v podobě rozpoznávání na základě markantů, avšak tento způsob bylo nutné upravit, aby odpovídal odlišné biometrické charakteristice, konkrétně vzoru nosu. Z tohoto důvodu bylo nutné seznámit se s touto charakteristikou a s různými problémy, které s ní souvisejí. Na základě podobnosti se strukturou otisku prstu bylo určeno, že vhodnou pozicí pro markanty budou středy uzavřených oblastí na vzoru nosu. Nicméně, dalším vhodným způsobem pro určení markantů by mohly být útvary rýh, které tyto oblasti ohraničují. K navržení postupu získání markantů bylo potřeba prostudovat i další práce, které se zabývají identifikací zvěře, konkrétně dobytka a psů. Hlavním přínosem těchto prací byla myšlenka získání oblasti zájmu, která z původní fotografie odstranila nepotřebné části. Jelikož tyto práce využívaly obecných přístupů pro vyhledání identifikačních bodů na fotografiích nosu, byl z těchto prací vytvořen souhrn, který by mohl posloužit jako základ pro navazující práce.

Ve finální verzi tohoto přístupu se využilo již zmíněného získání oblasti zájmu. Na oblast se poté aplikovala řada algoritmů, které měly za cíl převést obraz do podoby vhodné pro extrakci rysů. K tomu bylo potřeba obraz převést na šedotónovou variantu, na kterou se následně aplikoval bilaterální filtr, který z obrazu odstranil nežádoucí šum a zároveň zanechal kontrast mezi přechody rýh a uzavřených oblastí. Poté bylo na obraz aplikováno CLAHE, které zvýšilo kontrast mezi zmíněnými přechody. Následně se obraz binarizoval a pomocí oprav, morfologického otevření a filtrace kontur se převedl do požadované podoby. Markanty se vyhledaly pomocí určení středů zbývajících kontur. Následně v závislosti na vstupu od uživatele bylo možné tyto pozice převést do šablon a vypsát, nebo provést porovnávání.

Na konci bylo provedeno testování s cílem určit přesnost extrakce markantů. Při využití databáze o velikosti 171 fotografií, které byly pořízeny za různých podmínek, bylo dosaženo přesnosti 82,71%.

Z výše uvedených bodů je zřejmé, že byly splněny všechny požadavky stanovené v zadání. Navíc bylo v této práci implementováno rozšíření, které se zaměřuje na porovnání šablon a nalezení nejpodobnější fotografie. Dalším rozšířením tohoto přístupu by bylo extrahování orientace markantů, což by vedlo k ještě přesnějšímu porovnávání.

Literatura

- [1] *Deer Nose Guide (how a deer's sense of smell works)* [online]. [cit. 2023-03-10]. Dostupné z: <https://worlddeer.org/deer-nose/>.
- [2] *Histogram equalization* [online]. [cit. 2023-04-08]. Dostupné z: https://en.wikipedia.org/wiki/Histogram_equalization.
- [3] *Označování zvířat* [online]. [cit. 2023-01-27]. Dostupné z: https://cs.wikipedia.org/wiki/Ozna%C4%8Dov%C3%A1n%C3%AD_zv%C3%AD%C5%99at.
- [4] AHMED, S., GABER, T., THARWAT, A., HASSANIEN, A. E. a SNÁEL, V. Muzzle-Based Cattle Identification Using Speed up Robust Feature Approach. In: *2015 International Conference on Intelligent Networking and Collaborative Systems*. 2015, s. 99–104. DOI: 10.1109/INCoS.2015.60.
- [5] ALI, M. M., MAHALE, V. H., YANNAWAR, P. a GAIKWAD, A. T. Overview of fingerprint recognition system. In: *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. 2016, s. 1334–1338. DOI: 10.1109/ICEEOT.2016.7754900.
- [6] AWAD, A. I. a HASSABALLAH, M. Bag-of-Visual-Words for Cattle Identification from Muzzle Print Images. *Applied Sciences*. 2019, sv. 9, č. 22. DOI: 10.3390/app9224914. ISSN 2076-3417. Dostupné z: <https://www.mdpi.com/2076-3417/9/22/4914>.
- [7] AWAD, A. I., ZAWBAA, H. M., MAHMOUD, H. A., NABI, E. H. H. A., FAYED, R. H. et al. A robust cattle identification scheme using muzzle print images. In: *2013 Federated Conference on Computer Science and Information Systems*. 2013, s. 529–534.
- [8] BOESCH, G. *What is OpenCV? The Complete Guide (2023)* [online]. [cit. 2023-03-05]. Dostupné z: <https://viso.ai/computer-vision/opencv/>.
- [9] CHOI, H. I., LEE, Y., SHIN, H., LEE, S., CHOI, S. S. et al. The Formation and Invariance of Canine Nose Pattern of Beagle Dogs from Early Puppy to Young Adult Periods. *Animals*. 2021, sv. 11, č. 9. DOI: 10.3390/ani11092664. ISSN 2076-2615. Dostupné z: <https://www.mdpi.com/2076-2615/11/9/2664>.
- [10] EDITORS, B. O. *Rhinarium* [online]. [cit. 2023-01-26]. Dostupné z: <https://www.biologyonline.com/dictionary/rhinarium>.
- [11] EL HENAWY, I., EL BAKRY, H. a EL HADAD, H. Muzzle Classification Using Neural Networks. *International Arab Journal of Information Technology (IAJIT)*. 2017, sv. 14, č. 4.

- [12] FOLENTA, J. *Extrakce podrobných informací z otisku dlaně*. Brno, CZ, 2022. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/25007/>.
- [13] GABER, T., THARWAT, A., HASSANIEN, A. E. a SNASEL, V. Biometric cattle identification approach based on Weber's Local Descriptor and AdaBoost classifier. *Computers and Electronics in Agriculture*. 2016, sv. 122, s. 55–66. DOI: <https://doi.org/10.1016/j.compag.2015.12.022>. ISSN 0168-1699. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S016816991600003X>.
- [14] HORÁK, K. *Matematická morfologie* [online]. [cit. 2023-04-08]. Dostupné z: http://vision.uamt.feec.vutbr.cz/ZVS/lectures/11_Matematicka_morfologie.pdf.
- [15] HOSSAIN, M. E., KABIR, M. A., ZHENG, L., SWAIN, D. L., MCGRATH, S. et al. A systematic review of machine learning techniques for cattle identification: Datasets, methods and future directions. *Artificial Intelligence in Agriculture*. 2022, sv. 6, s. 138–155. DOI: <https://doi.org/10.1016/j.aiia.2022.09.002>. ISSN 2589-7217. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S2589721722000125>.
- [16] JAIN, A. K., PRABHAKAR, S. a HONG, L. A multichannel approach to fingerprint classification. *IEEE transactions on pattern analysis and machine intelligence*. IEEE. 1999, sv. 21, č. 4, s. 348–359.
- [17] JANG, D.-H., KWON, K.-S., KIM, J.-K., YANG, K.-Y. a KIM, J.-B. Dog Identification Method Based on Muzzle Pattern Image. *Applied Sciences*. 2020, sv. 10, č. 24. DOI: [10.3390/app10248994](https://doi.org/10.3390/app10248994). ISSN 2076-3417. Dostupné z: <https://www.mdpi.com/2076-3417/10/24/8994>.
- [18] KOLOUCHOVÁ, M. *Morfologické operace ve zpracování obrazu* [online]. Brno, 2008. [cit. 2023-04-08]. Diplomová práce. Vysoké učení technické v Brně. Fakulta informačních technologií. Ústav počítačové grafiky a multimédií. Dostupné z: <http://hdl.handle.net/11012/53247>.
- [19] KRŠIČKA, T. *Extrakce podrobných informací z plastických otisků prstů*. Brno, CZ, 2022. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/25077/>.
- [20] KUMAR, S. a SINGH, S. Cattle Recognition: A New Frontier in Visual Animal Biometrics Research. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*. Květen 2019, sv. 90. DOI: [10.1007/s40010-019-00610-x](https://doi.org/10.1007/s40010-019-00610-x).
- [21] KUMAR, S. a SINGH, S. K. Automatic identification of cattle using muzzle point pattern: a hybrid feature extraction and classification paradigm. *Multimedia Tools and Applications*. Dec 2017, sv. 76, č. 24, s. 26551–26580. DOI: [10.1007/s11042-016-4181-9](https://doi.org/10.1007/s11042-016-4181-9). ISSN 1573-7721. Dostupné z: <https://doi.org/10.1007/s11042-016-4181-9>.
- [22] KUMAR, S., SINGH, S. K., ABIDI, A. I., DATTA, D. a SANGAIAH, A. K. Group Sparse Representation Approach for Recognition of Cattle on Muzzle Point Images. *International Journal of Parallel Programming*. Oct 2018, sv. 46, č. 5, s. 812–837.

DOI: 10.1007/s10766-017-0550-x. ISSN 1573-7640. Dostupné z:
<https://doi.org/10.1007/s10766-017-0550-x>.

- [23] KUMAR, S., SINGH, S. K. a SINGH, A. K. Muzzle point pattern based techniques for individual cattle identification. *IET Image Processing*. 2017, sv. 11, č. 10, s. 805–814. DOI: <https://doi.org/10.1049/iet-ipr.2016.0799>. Dostupné z:
<https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-ipr.2016.0799>.
- [24] KUMAR, S., SINGH, S. K., SINGH, R. S., SINGH, A. K. a TIWARI, S. Real-time recognition of cattle using animal biometrics. *Journal of Real-Time Image Processing*. Sep 2017, sv. 13, č. 3, s. 505–526. DOI: 10.1007/s11554-016-0645-4. ISSN 1861-8219. Dostupné z: <https://doi.org/10.1007/s11554-016-0645-4>.
- [25] KUSAKUNNIRAN, W. a CHAIVIROONJAROEN, T. Automatic Cattle Identification based on Multi-Channel LBP on Muzzle Images. In: *2018 International Conference on Sustainable Information Engineering and Technology (SIET)*. 2018, s. 1–5. DOI: 10.1109/SIET.2018.8693161.
- [26] KUSAKUNNIRAN, W., WIRATSUDAKUL, A., CHUACHAN, U., KANCHANAPREECHAKORN, S., IMAROMKUL, T. et al. Biometric for Cattle Identification using Muzzle Patterns. *International Journal of Pattern Recognition and Artificial Intelligence*. 2020, sv. 34, č. 12, s. 2056007. DOI: 10.1142/S0218001420560078. Dostupné z:
<https://doi.org/10.1142/S0218001420560078>.
- [27] MAHMOUD, H. A. a HADAD, H. M. R. E. Automatic cattle muzzle print classification system using multiclass support vector machine. *International Journal of Image Mining*. 2015, sv. 1, č. 1, s. 126–140. DOI: 10.1504/IJIM.2015.070022. Dostupné z:
<https://www.inderscienceonline.com/doi/abs/10.1504/IJIM.2015.070022>.
- [28] MINAGAWA, H., FUJIMURA, T., ICHIYANAGI, M., TANAKA, K. et al. Identification of beef cattle by analyzing images of their muzzle patterns lifted on paper. In: *China Agricultural Sciencetech Press. AFITA 2002: Asian agricultural information technology & management. Proceedings of the Third Asian Conference for Information Technology in Agriculture, Beijing, China, 26-28 October, 2002*. 2002, s. 596–600.
- [29] NOVIYANTO, A. a ARYMURTHY, A. M. Automatic cattle identification based on muzzle photo using speed-up robust features approach. In: *Proceedings of the 3rd European conference of computer science, ECCS*. 2012, sv. 110, s. 114.
- [30] NOVIYANTO, A. a ARYMURTHY, A. M. Beef cattle identification based on muzzle pattern using a matching refinement technique in the SIFT method. *Computers and Electronics in Agriculture*. 2013, sv. 99, s. 77–84. DOI: <https://doi.org/10.1016/j.compag.2013.09.002>. ISSN 0168-1699. Dostupné z:
<https://www.sciencedirect.com/science/article/pii/S0168169913002093>.
- [31] OPENCV. *The OpenCV Reference Manual* [online]. [cit. 2023-04-08]. Dostupné z:
<https://docs.opencv.org/>.
- [32] PARIS, S. *A Gentle Introduction to Bilateral Filtering and its Applications* [online]. [cit. 2023-04-08]. Dostupné z:
http://people.csail.mit.edu/sparis/bf_course/slides/03_definition_bf.pdf.

- [33] PETERSEN, W. The Identification of the Bovine by Means of Nose-Prints¹. *Journal of Dairy Science*. 1922, sv. 5, č. 3, s. 249–258. DOI: [https://doi.org/10.3168/jds.S0022-0302\(22\)94150-5](https://doi.org/10.3168/jds.S0022-0302(22)94150-5). ISSN 0022-0302. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0022030222941505>.
- [34] SMEJKAL, J. *Myslivost v etnologické perspektivě*. Brno, CZ, 2019. Diplomová práce. Masarykova univerzita, Filozofická fakulta. Dostupné z: <https://is.muni.cz/th/pl1dp/>.
- [35] THARWAT, A., GABER, T. a HASSANIEN, A. E. Cattle Identification Based on Muzzle Images Using Gabor Features and SVM Classifier. In: HASSANIEN, A. E., TOLBA, M. F. a TAHER AZAR, A., ed. *Advanced Machine Learning Technologies and Applications*. Cham: Springer International Publishing, 2014, s. 236–247. ISBN 978-3-319-13461-1.
- [36] THARWAT, A., GABER, T., HASSANIEN, A. E., HASSANIEN, H. A. a TOLBA, M. F. Cattle Identification Using Muzzle Print Images Based on Texture Features Approach. In: KÖMER, P., ABRAHAM, A. a SNÁŠEL, V., ed. *Proceedings of the Fifth International Conference on Innovations in Bio-Inspired Computing and Applications IBICA 2014*. Cham: Springer International Publishing, 2014, s. 217–227. ISBN 978-3-319-08156-4.
- [37] TUMINAITE, I. a KRÖGER, R. H. H. Rhinarium skin structure and epidermal innervation in selected mammals. *Journal of Morphology*. 2021, sv. 282, č. 3, s. 419–426. DOI: <https://doi.org/10.1002/jmor.21313>. Dostupné z: <https://onlinelibrary.wiley.com/doi/abs/10.1002/jmor.21313>.
- [38] UMRI, B. K., UTAMI, E. a KURNIAWAN, M. P. Comparative Analysis of CLAHE and AHE on Application of CNN Algorithm in the Detection of Covid-19 Patients. In: *2021 4th International Conference on Information and Communications Technology (ICOIACT)*. 2021, s. 203–208. DOI: 10.1109/ICOIACT53268.2021.9563980.
- [39] ČEŠKA, P. *Extrakce podrobných informací z otisku prstu*. Brno, CZ, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/24033/>.
- [40] VSWITCHS. *Adaptive histogram equalization* [online]. [cit. 2023-04-08]. Dostupné z: https://en.wikipedia.org/wiki/Adaptive_histogram_equalization#Contrast_Limited_AHE.
- [41] ZAORÁLEK, L., PRILEPOK, M. a SNÁŠEL, V. Cattle Identification Using Muzzle Images. In: ABRAHAM, A., WEGRZYN WOLSKA, K., HASSANIEN, A. E., SNASEL, V. a ALIMI, A. M., ed. *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015*. Cham: Springer International Publishing, 2016, s. 105–115. ISBN 978-3-319-29504-6.
- [42] ZHAO, F. a TANG, X. Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction. *Pattern Recognition*. 2007, sv. 40, č. 4, s. 1270–1281. DOI: <https://doi.org/10.1016/j.patcog.2006.09.008>. ISSN 0031-3203. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0031320306003797>.