

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

## EXPERIMENTY S TERMOSNÍMKY RUKOU

BAKALÁŘSKÁ PRÁCE

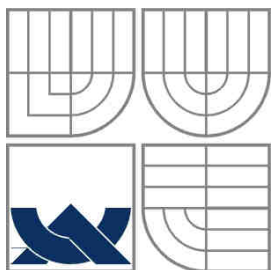
BACHELOR'S THESIS

AUTOR PRÁCE

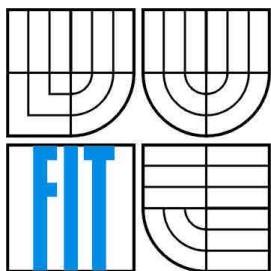
AUTHOR

RADEK FOLTÝN

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

## EXPERIMENTY S TERMOSNÍMKY RUKOU

EXPERIMENTS WITH THERMOSCANS OF HANDS

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

RADEK FOLTÝN

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. JAN VÁŇA

BRNO 2010

## **Abstrakt**

Jednou z biometrických metod pro identifikace člověka je identifikace pomocí otisku prstu nebo dlaně. K tomu, abychom mohli tuto identifikaci provést, je potřeba ruku nejdříve detekovat. Tato práce se zabývá dvěma problémy, prvním z nich je detekce ruky v termosnímku za pomoci hledání výrazných kontur (hrany dlaně, jednotlivé prsty). Druhým je separování užitečných dat z termosnímku, tedy takových, které souvisí s danou rukou (odstranění rušivých okolních částí).

## **Abstract**

This identification with fingerprints or palmprints is one of the biometric methods of man's identification. This thesis deals with two problems: The first one is the hand detection in thermoscans by means of outstanding contours search (palm edge, individual fingers). The second one is the depiction of useful data from thermoscans, i.e. those ones which are connected with the given hand (separation of interfering parts).

## **Klíčová slova**

Rozpoznávání ruky, termosnímký ruky, termosnímek, infračervená kamera, infračervený snímek, detekce hran, biometrie

## **Keywords**

Recognition of hands, thermoscans of hands, thermal image, infrared camera, infrared image, edge detection, biometrics

## **Citace**

Foltýn Radek: Experimenty s termosnímký rukou, bakalářská práce, Brno, FIT VUT v Brně, 2010

# Experimenty s termosnímkou rukou

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením:

Ing. Jana Váni

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Radek Foltýn  
19.5.2010

## Poděkování

Děkuji vedoucímu bakalářské práce Ing. Janu Váňovi za metodické vedení, pedagogickou a odbornou pomoc při zpracování mé bakalářské práce. Dále bych chtěl poděkovat všem dobrovolníkům, díky kterým se mi podařilo nasnímat používanou databázi.

© Radek Foltýn, 2010

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

# Obsah

Obsah .....	1
1 Úvod.....	3
2 Cíle práce .....	4
3 Databáze snímků a měření teplotního pole .....	5
3.1 Termografie.....	5
3.2 Jak pracuje termokamera.....	5
3.3 Emisivita.....	5
3.4 Tvorba databáze .....	6
3.5 Problémy při získávání termosnímků.....	6
3.6 Snímky s nevhodným kontrastem .....	7
3.7 Proč jsou snímky s nevhodným kontrastem nepoužitelné.....	8
3.8 Zpracování snímků dostupným softwarem .....	8
3.8.1 Problémy při zpracování snímků dostupným softwarem.....	10
3.9 Parametry termokamery .....	11
4 Zpracování snímků.....	12
4.1 Histogram.....	12
4.2 Metody prahování .....	13
4.2.1 Globální prahování.....	13
4.2.2 Adaptivní prahování.....	17
4.3 Metody použité pro detekci kontury ruky .....	18
4.3.1 Detekce ruky pomocí hledání výrazných kontur .....	18
4.3.2 Detekce hran pomocí Sobelova operátoru .....	18
4.3.3 Detekce hran pomocí Cannyho operátoru.....	18
4.3.4 Detekce hran pomocí kombinace optimálního prahu a Cannyho operátoru .....	20
5 Implementace .....	20
5.1 Vývojové prostředí a programovací jazyk .....	20
5.2 Objektový návrh .....	20
5.2.1 Třída Database .....	20
5.2.2 Třída ImageProcessing.....	21
5.3 Implementace metod za použití OpenCV .....	21
5.3.1 Použité metody pro detekci hran.....	21
5.3.2 Použité metody pro tvorbu masky .....	22
5.4 Překlad a ovládání programu.....	22
5.5 Činnost programu .....	23

5.6	Součásti programu.....	23
6	Testování.....	24
6.1	Způsob testování .....	24
6.2	Testování jednotlivých metod .....	24
6.3	Testování kombinací více metod.....	24
6.4	Výsledky testování .....	25
6.4.1	Testování metod hledajících optimální práh pro vytvoření masky .....	25
6.4.2	Metody pro detekci hran v obrázku .....	27
6.4.3	Testování kombinací metod.....	30
7	Závěr .....	38
	Literatura .....	39
	Seznam příloh .....	41
	Příloha A. Výstupy použitých metod.....	42
	Příloha B. Obsah CD .....	47

# 1 Úvod

Termosnímky jsou v dnešní době využívány v celé řadě vědních oborů, jako je například lékařství, stavebnictví, strojní průmysl (defektoskopie) apod. V této práci budou termosnímky použity jako zdroj biometrických dat.

Termosnímky mají oproti klasickým snímkům (fotografiím) řadu výhod, ale ovšem i několik nevýhod. Mezi jejich hlavní pozitiva patří nezávislost na intenzitě osvětlení, tedy je lze pořít i ve tmě. Další výhodou je jejich téměř nemožné zfalšování. Za hlavní nevýhodu bych považoval nestabilní teplotu ruky, tedy nemožnost udělat dva totožné a na sobě nezávislé snímky ruky jednoho člověka s větším časovým odstupem. Teplota ruky je ovlivněna teplotou v budově (místnosti), zdravotním stavem daného člověka nebo venkovní teplotou [6].

Využití termosnímků ruky v dnešní době z hlediska bezpečnosti nemá příliš velké uplatnění, jelikož je tato technologie poměrně nákladná a není možné pomocí ní přesně určit danou osobu (např. na rozdíl od otisku prstu pracují termosnímky s teplotou, a ta není v daných částech ruky vždy stejná). Pokud by se jednalo pouze o určení živosti, či analýzu toho, jestli se jedná o muže, ženu, či dítě, tak by nasazení těchto snímků mohlo být poměrně přesné, ovšem zde by mohlo docházet k omylům.

Jako řešení problému s jednoznačnou identifikací bych viděl kombinaci termografie s jinou technologií. Například při snímání otisku prstu by termosnímek sloužil právě jen k ověření živosti.

V kapitole Cíle práce jsou uvedeny jednotlivé cíle, kterými se tato práce zabývá. Teoretický rozbor a informace o termosnímku se nachází v kapitole Databáze snímků a měření teplotního pole. Vybraným metodám pro řešení jednotlivých problémů se věnuje kapitola Zpracování snímků. V předposlední kapitole je uveden způsob implementace aplikace. Kapitola Testování informuje o provedených testech a jejich vyhodnocení.

## 2 Cíle práce

Cílem práce je vytvoření programu, který bude sloužit k experimentům s termosnímkou ruky. Těmito experimenty budu hledat řešení následujících bodů :

- porovnat různé metody nebo jejich kombinace pro co nejlepší detekci kontury ruky v termosnímkou
- snažit se ze snímku získat data, která souvisí se pouze samotnou rukou a na ruce potom zvýraznit její jednotlivé části



## 3 Databáze snímků a měření teplotního pole

### 3.1 Termografie

Termografie je věda, která se zabývá analýzou rozložení teplotního pole na povrchu tělesa, a to bezkontaktním způsobem. Úkolem termografie je analýza infračervené energie vyzařované tělesem. Termografickým měřicím systémem lze zobrazit teplotní pole měřeného objektu, ovšem jen na jeho povrchu [7].

### 3.2 Jak pracuje termokamera

Princip práce termokamery je v podstatě totožný s prací pyrometru. Ten ovšem na rozdíl od termokamery ukazuje pouze teplotu bodu, na který je pyrometr zaměřen. Rozdíl je tedy v tom, že u termokamery dopadá infračervené záření na matici senzorů a měřená data jsou zobrazena jako 2D obraz na displeji kamery. Tento termogram znázorňuje rozložení teploty v oblasti a umožňuje vidět rozdíly teploty v různých oblastech. Tyto oblasti mohou být zobrazeny v různých barevných škálách. Pro naše měření postačila stupnice šedé, kde černá barva označuje chladnější část objektu [3, 4, 6].

### 3.3 Emisivita

Udává schopnost tělesa vyzařovat teplo a je určena jako poměr intenzity vyzařování reálného tělesa a tzv. *černého tělesa*<sup>1</sup>, majícího stejnou teplotu jako těleso reálné. Před samotným měřením je důležité správné nastavení emisivity, která může nabývat hodnot v intervalu  $<0; 1>$ . Hodnota 0 je potom pro těleso s absolutním odrazem, tedy není zde pohlceno žádné záření. Naopak hodnota 1 náleží právě již zmíněnému černému tělesu. Při snímání databáze termosnímků ruky byla použita emisivita náležící lidskému tělu, její hodnota je 0,98 [1].

---

<sup>1</sup> Černé těleso – maximálně pohlcuje i vyzařuje záření. Emisivita černého tělesa nabývá hodnoty 1.

## 3.4 Tvorba databáze

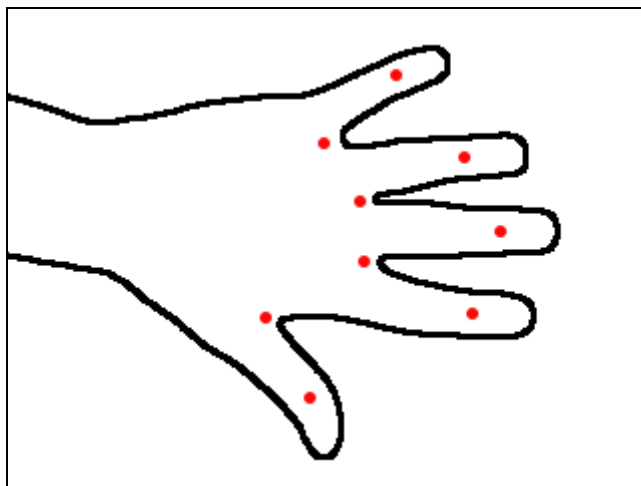
Databáze termosnímků byla vytvořena za účelem analýzy získaných dat, a ta spočívá v porovnání závislostí mezi jednotlivými termobary rukou. Tyto závislosti bude detekovat vytvořený program, který dokáže získat ze snímku požadovaná data.

Databáze snímků byla vytvořena termokamerou ThermoCAM E300 v biometrické laboratoři ÚITS (Ústav inteligentních systémů). Bylo důležité nasnímat databázi v jednom časovém období, abychom předešli různým nastavením pro různé skupiny lidí. To se ve velké míře povedlo. Proto jsou podmínky pro získání snímků nastaveny stejně. Celkem databáze obsahuje snímky od 32 lidí, z toho 4 ženy a 28 mužů. Jedná se o lidi ve věku od 17 do 24 let. Tyto poměry bohužel nejsou vyvážené, ale tato práce nemá za úkol porovnávat rozdíly mezi věkovými skupinami, proto lze tento nedostatek zanedbat.

U každého z 32 účastníků bylo získáno 8 snímků. Jednalo se o dlaně a hřbety obou rukou a každé z těchto natočení bylo snímáno dvakrát nezávisle na sobě – tedy tak, že účastník zvednul ruku z podložky vyrobené pro měření (obr. 3.1), a poté ji opět přiložil. Díky tomu bylo dosaženo toho, že žádné dva snímky nejsou totožné, protože účastník nepřiloží ruku nikdy naprosto stejně.

## 3.5 Problémy při získávání termosnímků

Na začátku celého fotografování bylo důležité uvědomit si v jaké poloze bude ruka natočena, tak aby byl pořízen dostatečně kvalitní snímek a zároveň aby se tato poloha dala opakovat vícekrát a nezávisle na sobě. Po konzultaci s panem Ing. Váňou jsme dospěli ke tvaru šablony vhodné pro toto použití. Ovšem tím problém nekončil, ba naopak. Původní materiál (dřevo), ze kterého měla být šablona vyrobena, měl tu nežádoucí vlastnost, že po zahřátí trvalo dlouho, než se opět vrátil na svoji původní teplotu. To mělo za následek, že tepelná stopa po jednom účastníkovi měření znemožňovala okamžité snímání dalších snímků. Proto bylo potřeba vyrobit buď šablonu, která rychle odvede získané teplo a opět se vrátí na svoji původní teplotu, nebo takovou, která se ani po delším zahřívání nezahřeje natolik, aby ovlivnila následující snímky. Z hlediska obtížnosti výroby takové šablony jsem zvolil snazší řešení. Podložka šablony byla nakonec vyrobena z 2mm papíru a bloky určující směr a vzdálenost mezi prsty z balsového dřeva, viz obrázek 3.1.



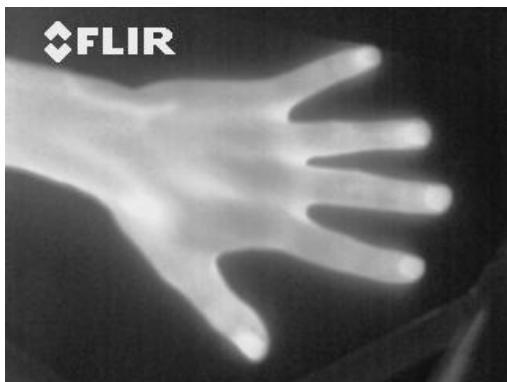
*Obrázek 3.1: Schéma použité šablony*

Dalším důležitým bodem bylo nastavení termokamery, které spočívalo ve vhodném nastavení emisivity a polohy pro snímání jednotlivých snímků. Nastavit emisivitu nebylo obtížné, neboť její hodnota se nachází buď v tabulkách nebo na internetových stránkách, které se touto problematikou zabývají [1]. Dále bylo potřeba nastavit teplotní interval, tedy rozmezí v jakém bude kamera pro daný snímek pracovat. Tento interval se mění dynamicky v závislosti na teplotě ruky. Bohužel nebylo možné nastavit obecný rozsah, proto byl použit automatický režim a snímky byly upraveny až za použití softwaru.

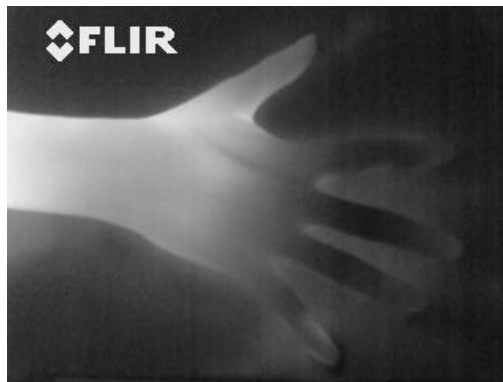
Posledním problémem bylo zaostření snímku, protože displej zobrazuje jen snímek o velikosti 320 x 240 pixelů. Navíc rozlišení displeje neposkytuje díky jeho kvalitě úplně ostrý obraz, přeci jen tato kamera není primárně používána na biometrická měření.

## **3.6 Snímky s nevhodným kontrastem**

V našem případě je za nevhodný snímek považován každý snímek, který nelze úspěšně zpracovat zvoleným softwarem, ať už tedy programem tvořeným v této práci nebo přímo dodanou aplikací od společnosti FLIR. Samozřejmě, i když je snímek správně zaostřen a má správně nastavenou emisivitu, nemusí být vhodný ke zpracování, a to hlavně díky nedostačujícímu kontrastu s pozadím (obr. 3.2). Naopak vhodným snímkem je snímek, kde je tento kontrast oproti pozadí co nejvyšší (obr. 3.1).



*Obrázek 3.1: Vhodný snímek*



*Obrázek 3.2: Nevhodný snímek*

### **3.7 Proč jsou snímky s nevhodným kontrastem nepoužitelné**

Jak jsem již zmínil výše, snímky mohou být nevhodné pro zpracování hned z několika důvodů. V této práci jsem se zabýval jedním z nich – teplotou ruky. Např. zatímco obrázek 3.1 zobrazuje ruku jako teplejší objekt ve srovnání s podložkou, na obrázku 3.2 je ruka buď stejně chladná, nebo chladnější než podklad. To mohlo být způsobeno špatným prokrvováním prstů, nebo příchodem osoby z chladnějšího prostředí.

Abychom co nejvíce eliminovali kolísání teplot rukou způsobené rozdílnou teplotou venkovní a laboratorní, byli účastníci měření ponecháni v pokojové teplotě alespoň 20 minut před začátkem měření. Ovšem ani potom nebylo možné vyloučit rozdílnou teplotu rukou v závislosti na teplotě laboratorní, a to z důvodu např. horšího prokrvování končetin.

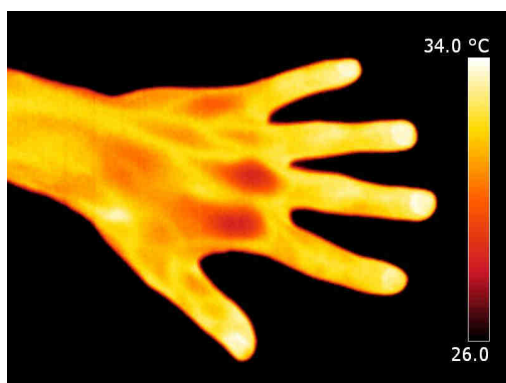
Na takto chladné ruce není možné zjistit, která místa mají přirozeně menší teplotu jako například klouby, a nelze jasně identifikovat ani místa teplejší, tedy místa s lepším prokrvením jako jsou dlaně a prsty.

### **3.8 Zpracování snímků dostupným softwarem**

Programů zabývajících se zpracováním termosnímků je dnes celá řada, bohužel většina z nich je komerčních. Proto jsem pro experimenty nejprve použil software dodávaný k termokameře. Jednalo se o program FLIR QuickReport v1.2 SP 1 s přidávanými částmi IRSDK a IRViewer. Tento software je rovněž zdarma ke stažení na stránkách společnosti FLIR.

QuickReport poskytuje celou řadu možností ke zpracování termosnímků, jako například přesný výpis podmínek, které byly na kameře nastaveny v době pořízení snímku, dále je zde možnost označení kteréhokoli místa na snímku a získání teplotního údaje o tomto místě. Ovšem asi nejpoužívanější funkcí v mém případě bylo manuální nastavení teplotního rozsahu pro každý snímek, protože volba jednoho konstantního intervalu by nevedla ke kontrastnímu vykreslení ruky na všech snímcích.

Níže jsou uvedeny jednotlivé režimy zobrazení, které dokáže software zobrazit. I když je snímek pořízen pouze v režimu šedé, tak je díky metadatům (EXIF<sup>2</sup>) možné vykreslit snímek i jinak než jen v této šedé.

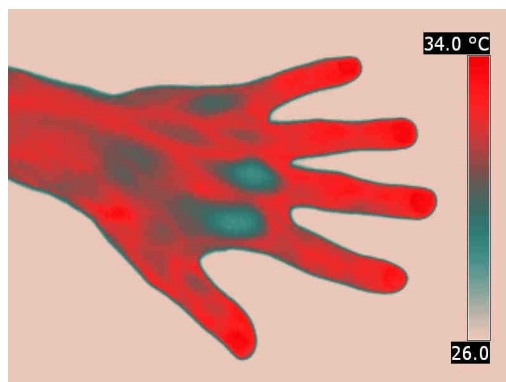


Obrázek 3.3: *GlowBow*

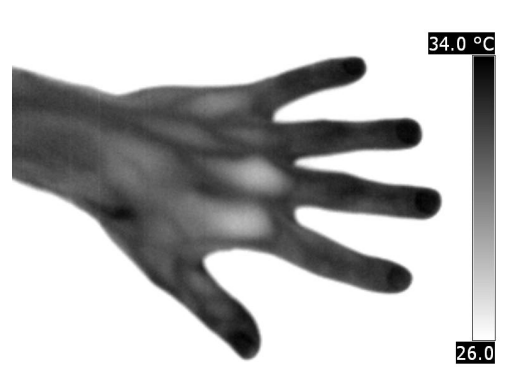


Obrázek 3.4: *Grey*

Pro další porovnání s implementovanými metodami jsem použil obrázek v zobrazení *Grey* (3.4)



Obrázek 3.5: *GreyRed*



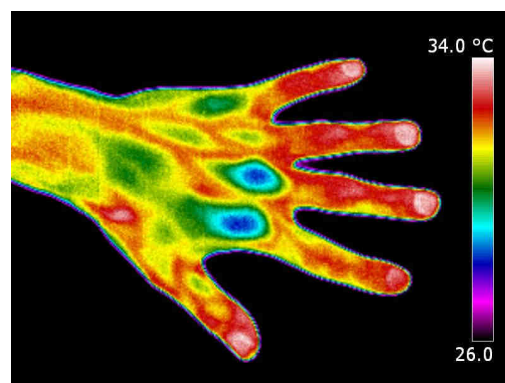
Obrázek 3.6: *Inverted Grey*

---

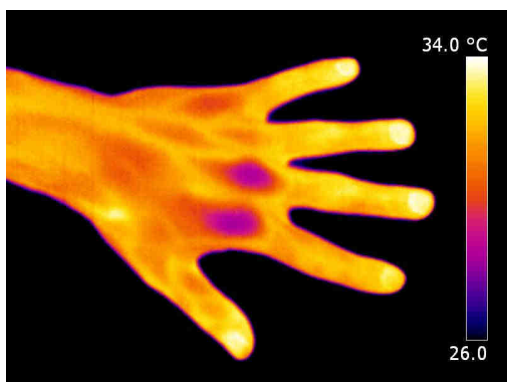
<sup>2</sup> EXIF (Exchangeable image file format) - je specifikace pro formát metadat, vkládaných do souborů digitálními fotoaparáty. Informace se vkládají do existujících souborových formátů.



Obrázek 3.7: Iron



Obrázek 3.8: Rain



Obrázek 3.9: Medical

### 3.8.1 Problémy při zpracování snímků dostupným softwarem

Jistá úskalí se objevila i při úpravě snímků, zejména šlo o to, že stupnice musela být nastavována manuálně, protože ji automatický režim nedokázal nastavit tak, aby byl kontrast obrysu ruky a okolního prostředí dostatečně velký. Toto manuální nastavování mě přivedlo na myšlenku tuto činnost zautomatizovat.

Dalším důležitým krokem bylo zachování metadat (EXIF) při úpravě snímku v některém jiném programu než v QuickReportu. V takovém případě zůstaly základní informace nepoškozeny, ovšem údaje spojené přímo s termosnímkem, jako rozsahy teplot, emisivita, vzdálenost apod. byly ztraceny bez možnosti je znovu doplnit.

Proto v případě, že bychom chtěli snímek později zpracovat QuickReportem, není vhodné jej upravovat v jiném softwaru, než v tom, který je k tomu určen. Jinou možností by bylo přistoupit k metadatům a zabránit jejich přepsání či poškození. Po několika neúspěšných pokusech jsem od této myšlenky upustil.

## 3.9 Parametry termokamery

Pro snímání termosnímků rukou byla použita termokamera řady E od společnosti FLIR. Její přesné označení je ThermaCAM E300. Tato kamera spadá do střední třídy termovizních kamer. Její primární využití není ve snímání biometrických dat, ale slouží ke kontrole kabelových svazků, transformátorů a el. vedení. Ovšem pro požadovaná měření byla dostačující [5].

*Tabulka 3.1: Základní parametry termovizní kamery ThermaCAM E300*

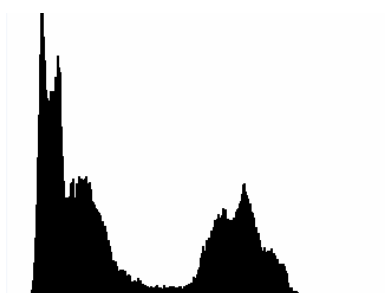
	<b>ThermaCAM E300</b>
<b>Detektor</b>	320 x 240 bodů
<b>Citlivost</b>	0,1°C (nebo 0,08°C)
<b>Frekvence</b>	50 Hz
<b>Rozsah</b>	-20°C až + 500°C (možno až + 1200°C)

## 4 Zpracování snímků

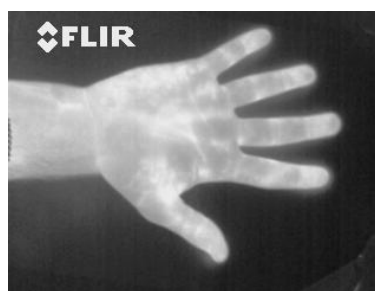
### 4.1 Histogram

Histogram je reprezentací rozložení barev v obrazu. Používá se v počítačové grafice a digitální fotografii. Vyjadřuje poměrné zastoupení počtu pixelů každého z daných barevných rozsahů.

V této práci je histogramem graf, na jehož ose x je vyneseno 256 bodů, odpovídajících 256 odstínům šedé (od černé po bílou). Na ose y je znázorněn počet pixelů patřících do daného rozsahu. Na obrázcích je zobrazen termosnímek (obr. 4.2) a k němu příslušný histogram (obr. 4.1) [13, 14].



Obrázek 4.1: Histogram



Obrázek 4.2: Snímek k histogramu

Při tvorbě histogramu u barevných snímků jsou zastoupeny všechny tři barevné složky (RGB), a histogram je tedy tvořen pro každou z nich zvlášť. Při analýze termosnímků nepotřebuji využívat všechny tři barevné složky, proto v této práci pracuji pouze s odstíny šedé. V tom případě bude každá z barevných složek zastoupena v obrázku rovnocenně.

Proč nás vlastně zajímá histogram? Jak bylo napsáno v kapitole 2, cílem práce je extrahovat ruku od pozadí pro pozdější úpravy. Takto extrahovaná ruka bude mít ve vytvořené masce bílou barvu, viz kapitola 6.4.1. Tato maska bude potom na původní snímek dále aplikována, čímž zobrazíme pouze užitečná data (veškeré okolí ruky bude vymaskováno – překryto černou barvou).



## 4.2 Metody prahování

Cílem je najít optimální práh, který bude sloužit k tvorbě masky, viz předcházející kapitola.

### 4.2.1 Globální prahování

Globální prahování je funkce, která obecně upravuje hodnoty vstupu podle předpisu:

$$G(x, y) = \begin{cases} 1 & \text{pro } I(x, y) \geq T \\ 0 & \text{pro } I(x, y) < T \end{cases} \quad (4.1)$$

kde  $I(x, y)$  je šedotónový obraz,  $G(x, y)$  je výstupním obrazem a  $T$  je zvolený práh.

Tento způsob prahování má jak svoje výhody, tak i značnou řadu nevýhod. Jako hlavní klady bych uvedl její rychlost a jednoduchost. Za nevýhodu lze považovat celkem složité nalezení hodnoty optimálního prahu. Další nevýhodou je, že pokud nemá obrázek velké rozdíly mezi intenzitou detekovaného objektu a pozadím, lze tento objekt špatně nebo vůbec detekovat, právě kvůli použití jen jednoho prahu pro celý obrázek.

#### 4.2.1.1 Otsu algoritmus na hledání optimálního prahu

Metoda spočívá v tom, že je histogram rozdělen na dvě části (třídy) –  $C_0$  a  $C_1$ , kde jedna reprezentuje objekt na obrázku a druhá pozadí. Tyto hodnoty jsou rozděleny prahem v hodnotě  $k$ , což znamená, že body patřící do třídy  $C_0$  mají hodnotu jasu v intervalu  $\langle 0; k-1 \rangle$  a body ze třídy  $C_1$  v intervalu  $\langle k; J-1 \rangle$ . Hodnota  $J$  odpovídá počtu jasových úrovní obrazu. Tedy pravděpodobnost výskytu bodu ze třídy  $C_0$  (označíme ji jako  $\omega_0$ ) je rovna

$$\omega_0 = \sum_{i=0}^{k-1} p_i = \omega(k) \quad (4.2)$$

a pravděpodobnosti výskytu ve třídě  $C_1$  (označíme jako  $\omega_1$ ) odpovídá vztah

$$\omega_1 = \sum_{i=k}^{J-1} p_i = 1 - \omega(k), \quad (4.3)$$

kde  $p_i$  je  $i$ -tá hodnota z relativního histogramu. Jelikož máme pouze dvě třídy, do nichž jsou všechny obrazové body rozděleny, platí práh

$$\omega_0 + \omega_1 = 1 \quad (4.4)$$

Nyní určíme střední hodnotu obou tříd, Pro třídu  $C_0$  platí

$$\mu_0 = \sum_{i=1}^{k-1} \frac{ip_i}{\omega_0} = \frac{\mu(k)}{\omega(k)}, \quad (4.5)$$

pro třídu  $C_1$  platí

$$\mu_1 = \sum_{i=k}^{J-1} \frac{ip_i}{\omega_1} = \frac{\mu_T - \mu(k)}{1 - \omega(k)}. \quad (4.6)$$

Dále zavedeme hodnoty

$$\omega(k) = \sum_{i=0}^{k-1} p_i \text{ a } \mu(k) = \sum_{i=0}^{k-1} ip_i \quad (4.7)$$

jako nultý a první centrální moment histogramu do jasu  $k-1$  a hodnotu

$$\mu_T = \sum_{i=0}^{J-1} ip_i \quad (4.8)$$

jako celkovou střední hodnotu celého originálního obrazu. Z těchto vztahů nyní můžeme snadno odvodit platnost dalšího vztahu

$$\omega_0 \mu_0 + \omega_1 \mu_1 = \mu_T \quad (4.9)$$

Dále se zde zavádí pojem variance (různost, kolísání) třídy. Variance je definována pro třídu  $C_0$  a  $C_1$  takto:

$$o_0^2 = \sum_{i=0}^{k-1} \frac{(i - \mu_0)^2}{\omega_0}, \quad o_1^2 = \sum_{i=k}^{J-1} \frac{(i - \mu_1)^2 p_i}{\omega_1}. \quad (4.10)$$

Nyní se dostáváme k jádru problému, definujeme totiž takzvanou mezitřídní varianci  $o_B^2$ , jejíž maximalizací získáme hodnotu prahu v histogramu. Platí pro ni následující vztah

$$o_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2, \quad (4.11)$$

který si ještě dále upravíme a zjednodušíme aplikací výše uvedených vzorců. Tuto úpravu si podrobně rozepíšeme. Nejprve dosadíme za  $\mu_T$  ze vztahu (4.10) a umocníme a roznásobíme výrazy v závorkách, čímž obdržíme tento vztah

$$\begin{aligned} o_B^2 = & \omega_0 \mu_0^2 - 2\mu_0^2 \omega_0^2 - 2\mu_0 \omega_0 \mu_1 \omega_1 + \omega_0^3 \mu_0^2 + 2\mu_0 \omega_0^2 \mu_1 \omega_1 + \omega_0 \omega_1^2 \mu_1^2 + \\ & + \omega_1 \mu_1^2 - 2\mu_0 \omega_0 \mu_1 \omega_0 - 2\mu_1^2 \omega_1^2 + \omega_0^2 \mu_0^2 \omega_1 + 2\mu_0 \omega_0 \mu_1 \omega_1^2 + \omega_1^3 \mu_1^2. \end{aligned} \quad (4.12)$$

Zaměříme se na šestý a poslední výraz a vytkneme z něj hodnotu  $\omega_1^2 \mu_1^2$ , z čehož nám vznikne výraz  $\omega_1^2 \mu_1^2 (\omega_0 + \omega_1)$ . Podle výše uvedeného vztahu (4.4) víme, že  $\omega_0 + \omega_1 = 1$ , a tedy získáme výsledný vztah pouze ve tvaru  $\omega_1^2 \mu_1^2$ . Podobným způsobem zjednodušíme i čtvrtý a desátý výraz na tvar  $\omega_0^2 \mu_0^2$ . Dále si vezmeme třetí, pátý, osmý a předposlední výraz a vytknutím z nich vytvoříme výraz  $2\mu_0 \omega_0 \mu_1 \omega_1 (-1 + \omega_0 - 1 + \omega_1)$ . Výraz v závorce se nám opět zredukuje a to na hodnotu -1.

Napíšeme-li si nyní celý původní výraz znovu, zjistíme, že již došlo ke značnému zjednodušení. Jeho tvar bude:

$$o_B^2 = \omega_0 \mu_0^2 - 2\mu_0^2 \omega_0^2 + 2\mu_0 \omega_0 \mu_1 \omega_1 + \omega_0^2 \mu_0^2 \omega_1 + \omega_1^2 \mu_1^2 + \omega_1 \mu_1^2 - 2\mu_1^2 \omega_1^2 \quad (4.13)$$

Sečteme druhý výraz se čtvrtým a pátý s posledním, vyjde nám tato rovnice

$$o_B^2 = \omega_0 \mu_0^2 - \mu_0^2 \omega_0^2 + 2\mu_0 \omega_0 \mu_1 \omega_1 + \omega_1 \mu_1^2 - \mu_1^2 \omega_1^2 \quad (4.14)$$

Vytknutím z jejich prvních dvou členů získáme výraz  $\omega_0 \mu_0^2 (1 - \omega_0)$ , což je vlastně  $\omega_0 \omega_1 \mu_1^2$ . Aplikací stejného postupu na poslední dva členy uvedené rovnice obdržíme výraz  $\omega_0 \omega_1 \mu_1^2$ . Opět si napíšeme celou rovnici, jak bude vypadat po této úpravě

$$o_B^2 = \omega_0 \omega_1 \mu_0^2 + \omega_0 \omega_1 \mu_1^2 - 2\mu_0 \omega_0 \mu_1 \omega_1 \quad (4.15)$$

Nyní již jen provedeme vytknutí  $\omega_0 \omega_1$  ze všech členů a ještě drobnou úpravu, kde zbylé členy přepíšeme na čtvercový tvar, a tím se dopracujeme k výslednému tvaru celé rovnice

$$o_B^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2 \quad (4.16)$$

Tento vztah můžeme přepsat dále pomocí vztahů (4.2), (4.3), (4.5), (4.6) na tvar

$$o_B^2(k) = \omega(k)(1 - \omega(k)) \left( \frac{\mu_T - \mu(k)}{1 - \omega(k)} - \frac{\mu(k)}{\omega(k)} \right)^2 \quad (4.17)$$

Zlomky rozšíříme a převedeme na společného jmenovatele, čehož obdržíme vztah

$$o_B^2(k) = \omega(k)(1 - \omega(k)) \left( \frac{\mu_T \omega(k) - \mu(k) \omega(k) - \mu(k) + \mu(k) \omega(k)}{\omega(k)(1 - \omega(k))} \right)^2. \quad (4.18)$$

Vidíme, že nyní můžeme vykrátit jmenovatele s výrazem před zlomkem a rovněž můžeme dopočítat čitatele ve zlomku. Výsledkem je vzorec tvaru

$$o_B^2(k) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \quad (4.19)$$

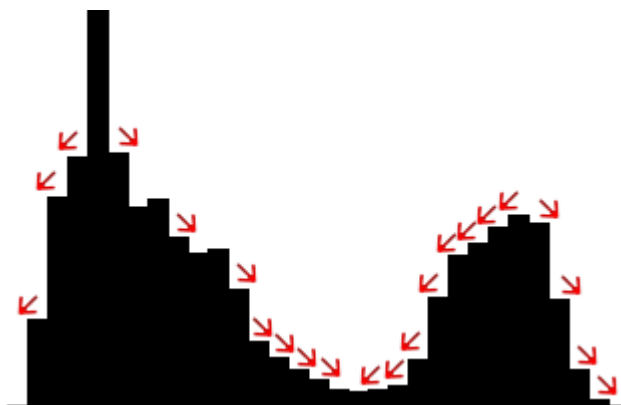
Hledanou prahovou hodnotu potom dostaneme vyhodnocením tohoto výrazu pro všechna  $k \in \langle 0; J-1 \rangle$  a nalezením takového  $k$ , pro které výraz nabude maximální hodnoty. Výsledné  $k^*$  tedy bude zapsáno takto

$$o_B^2(k^*) = \max_{0 \leq k < J-1} o_B^2(k) \quad (4.20)$$

Hodnota prahu bude v histogramu určena v místě, kde je vzdálenost středních hodnot obou dvou tříd maximální a dojde tak k optimální separaci pozadí a samotného objektu. [9, 18]

#### 4.2.1.2 Vlastní metoda pro hledání ideálního prahu

Moje metoda pro hledání hodnoty ideálního prahu vychází principiálně z Otsu algoritmu – jedná se o průchod histogramem a hledání prahové hodnoty, ale s tím rozdílem, že množina bodů v histogramu není rozdělena na dvě podmnožiny (třídy) a rovněž nehledám jejich průnik, který bych označil za nově nalezený práh, ale počítám počet sestupných a vzestupných hran v histogramu a počtem těchto hran udávám váhu (hodnotu) bodu na ose x, který je minimem pro danou část histogramu viz obrázek 4.3.



Obrázek 4.3: Hledání prahové hodnoty

Pro výpočet této hodnoty používám histogram pouze o 32 hodnotách na ose x, a to z důvodu menší členitosti, jelikož nepoužívám žádné vyhlazení histogramu ani jej neprokládám křivkou. Získaný výsledek násobím 8x, abych opět získal hodnotu v rozsahu  $\langle 0; 255 \rangle$ .

#### 4.2.2 Adaptivní prahování

V případě, že nelze určit jeden globální práh pro celý snímek, je potřeba vybrat jinou vhodnější metodu, jako například adaptivní prahování.

Tato metoda počítá práh pro každý jednotlivý pixel obrazu zvlášť a v případě, že je daný pixel menší než hodnota prahu, dostane pixel barvu pozadí bílé. V případě, že bude hodnota vyšší, pixel se zobrazí jako černý. Pro získání tohoto prahu existuje několik metod jako například metoda Chow a Kanenko nebo Otsu algoritmus viz níže [11].

#### 4.2.2.1 Metoda Chow a Kanenko

Metoda od Chow a Kanenka funguje na principu rozdělení obrazu do pole překrývajících se pod-obrazů a následném nalezení vhodné prahové hodnoty pro každý tento pod-obraz vyšetřením jeho histogramu. Prahová hodnota pro každý pixel je vypočtena interpolací hodnot jednotlivých pod-obrazů. Nevýhoda tohoto přístupu je výpočetní náročnost, a tak není vhodná pro real-time aplikace [9].

### 4.3 Metody použité pro detekci kontury ruky

#### 4.3.1 Detekce ruky pomocí hledání výrazných kontur

Kontura je ve snímku popsána jako množina bodů, které jsou postupně ukládány. Hledání této množiny probíhá tím způsobem, že na binárním snímku jsou nalezeny všechny body, které se v obrázku nacházejí na hranici dvou od sebe kontrastních oblastí. Po nalezení všech těchto bodů je výsledná množina proložena polygonální křivkou. Výsledkem tohoto proložení je spojitá nebo jen částečně spojitá kontura ruky [19].

#### 4.3.2 Detekce hran pomocí Sobelova operátoru

Sobelův operátor se využívá pro detekci vodorovných a svislých hran pomocí masek  $h_1$  a  $h_2$ .

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad h_2 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Pro detekci vodorovných i svislých hran v obrázku je tedy potřeba využít obě tyto masky. Dále je třeba udělat součet těchto dvou masek, abychom dostali do jednoho výsledného obrázku jak svislé, tak vodorovné hrany. Pro součet těchto hran je potřeba sečíst jejich absolutní hodnoty, protože kdybychom tak neučinili, získali bychom bílý obrázek. Výsledný obrázek je pro lepší znázornění ještě naprahován [12].

#### 4.3.3 Detekce hran pomocí Cannyho operátoru

Cannyho hranový detektor je algoritmus zahrnující několik kroků pro získání co nejlepšího výsledku při detekci hran v dvourozměrném diskretním obraze [15].

Jednotlivé kroky použité k detekci hran pomocí Cannyho operátoru:

- **Eliminace šumu Gaussovým filtrem**

Dvourozměrná varianta Gaussova normálního rozdělení je dána vztahem

$$G(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}},$$

kde  $x, y$  jsou souřadnice pixelu v obraze a sigma je standardní odchylka tohoto rozdělení. Výpočet je vhodné realizovat pomocí konvoluce. Vzorcem je tedy vypočtena pouze konvoluční maska, která se pak aplikuje na celý obraz.

- **Velikost a směr gradientu**

Hrany v obrázku se mohou vyskytovat v různých směrech, proto Cannyho operátor používá čtyři filtrů pro detekci horizontálních, vertikálních a dvou diagonálních hran v tomto obrázku. Pro detekci hran využívá např. Sobelův operátor, který vrátí hodnotu pro první derivát, v horizontálním směru ( $G_y$ ) a vertikálním směru ( $G_x$ ). Hranový gradient může být popsán takto:

$$G = \sqrt{G_x^2 + G_y^2} \quad (4.2)$$

- **Ztenčení**

Úkolem této funkce je vybrat z hodnot gradientů jen lokální maxima. Respektive odebrat body, které nejsou maximem. Tím zajistíme, že hrana bude detekována v místě největšího gradientu.

- **Prahování**

Nyní máme detekovány hrany, ovšem ne všechny mají stejné lokální maximum, proto není vhodné zvolit jeden globální práh, ale zvolíme dvojici prahů  $T_1$  a  $T_2$ , mezi kterými se budeme pohybovat. Pokud hodnota gradientu daného pixelu leží nad větším prahem  $T_2$ , je přímo označen jako hranový. Pokud posuzujeme bod, jehož hodnota leží mezi  $T_1$  a  $T_2$ , pak je „uznán“ jedině pokud sousedí s bodem, který byl jako hrana označen už dříve.

### 4.3.4 Detekce hran pomocí kombinace optimálního prahu a Cannyho operátoru

Obrázek je nejdříve naprahován prahem  $T$ , který jsem vypočítal vlastní metodou pro hledání optimálního prahu, viz kapitola 4.2.1.2. Po tomto naprahování získáme v ideálním případě bílou ruku na černém pozadí.

Pro získání kontury ruky je na takto upravený snímek použit Cannyho hranový operátor. Použitím tohoto operátoru získáme spojitou konturu ruky.

## 5 Implementace

### 5.1 Vývojové prostředí a programovací jazyk

Program pro detekci ruky v termosnímku byl napsán v jazyce C++ s využití knihovny pro zpracování obrazu OpenCV (**O**pen **S**ource **C**omputer **V**ision), která je zaměřena na počítačové vidění a manipulaci s obrazem v reálném čase. Jako editor byl použit nástroj Eclipse.

### 5.2 Objektový návrh

Dokončená aplikace umožňuje využívat všechny metody zmíněné v kapitole 6. Celý program (jeho třídy) je navržen tak, aby byl co nejnázorněji rozšiřitelný o další funkčnost.

Dále budou popsány jednotlivé abstraktní třídy<sup>3</sup> zajišťující načtení a zpracování snímků.

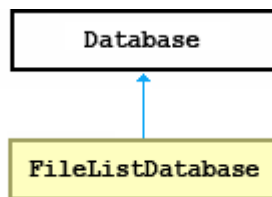
#### 5.2.1 Třída Database

Třída `Database` zajišťuje načítání obrázku ze souboru. Od této abstraktní třídy je odvozena třída `FileListDatabase` (viz obr. 5.1), která zajišťuje inicializaci (`init`), načítání dalšího obrázku (`nextImage`) a jeho návrat na první záznam (`reset`). Vlastní implementací třídy `Database` lze program libovolně modifikovat pro vlastní potřeby.

---

<sup>3</sup> Abstraktní třída má v C++ alespoň jednu prázdnou virtuální metodu

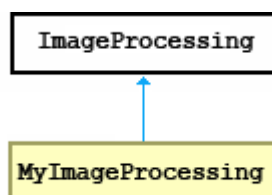




Obrázek 5.1: Definice třídy Database

## 5.2.2 Třída ImageProcessing

Abstraktní třída `ImageProcessing` má za úkol zpracování snímku, a proto se nestará o jeho načtení, ale pouze o práci s ním. Třída `ImageProcessing` obsahuje virtuální metody pro zpracování obrázku (`processImage`), předávání parametrů (`prepare`) a vypisování nápovědy (`help`). Z této třídy je odvozena třída `MyImageProcessing` viz obr. 5.2.



Obrázek 5.2: Definice třídy ImageProcessing

## 5.3 Implementace metod za použití OpenCV

### 5.3.1 Použité metody pro detekci hran

Metody použité pro detekci hran v termosnímku ruky:

- **Detekce ruky pomocí `cvFindContours`**

Pro detekci a vykreslení kontur je použita kombinace několika vestavěných funkcí, které OpenCV nabízí. Konkrétně se jedná o tyto: `cvFindContours()`, `cvApproxPoly()` a `cvDrawContours()`. Činnost těchto funkcí spočívá v tom, že detekují a vykreslí nalezenou konturu. Tuto konturu je možné roztáhnout pomocí funkce `cvDilate()`.

- **Cannyho operátor**

Cannyho operátor je implementován pomocí vestavěné funkce `cvCanny()`, u které je potřeba nastavit vstupní obrázek, výstupní obrázek a dvě hodnoty prahu.

- **Sobelův operátor**

Sobelův operátor je rovněž součástí knihovny OpenCV a je implementován funkcí `cvSobel()`, kde je potřeba nastavit následující parametry: vstupní obrázek, výstupní obrázek, volbu, zda bude operátor aplikován vertikálně nebo horizontálně a velikost okna, ze kterého bude počítána výsledná derivace.

- **Detekce hran pomocí kombinace optimálního prahu a Cannyho operátoru**

V této metodě je použita vlastní metoda pro získání optimální hodnoty prahu a funkce `cvCanny()`, díky které získáme z naprahovaného obrázku spojitou konturu.

## 5.3.2 Použité metody pro tvorbu masky

- **Globální prahování**

- **Otsu algoritmus**

Možnost nalezení optimálního prahu pomocí Otsu algoritmu je v OpenCV implementováno jako parametr ve funkci `cvThreshold()`.

Ukázka toho, jak vypadá volání této funkce: `cvThreshold(this->image, otsu, 128, 255, CV_THRESH_BINARY | CV_THRESH_OTSU);`

- **Vlastní metoda pro hledání optimálního prahu**

Moje vlastní metoda, která se pro toto použití jevila jako nejvhodnější, vychází z prohledávání histogramu a hledání optimálního prahu. Histogram je zde reprezentován jako pole hodnot, který tento algoritmus prochází a přiděluje jednotlivým odstínům váhu, jak je popsáno v kapitole 4.2.1.2.

- **Adaptivní prahování**

Adaptivní prahování je v práci zastoupeno vestavěnou funkcí `cvAdaptiveThreshold()`

## 5.4 Překlad a ovládání programu

Program lze přeložit pomocí přidaného makefile souboru a standardního minGW, ovšem za předpokladu, že daný uživatel bude mít nainstalovanou a zkompilevanou knihovnu OpenCV.

Pro ovládání programu je zde vytvořena krátká nápověda (program se spustí s parametrem `-h`), ve které je uveden následující seznam přepínačů, kterými může uživatel spustit příslušnou metodu.

Seznam používaných přepínačů:

- [-1] -> Originalni obrazek
- [-2] -> Detekce hran pomoci funkce cvCanny
- [-3] -> Detekce hran pomoci funkce cvFindContours
- [-4] -> Detekce hran pomoci funkce cvSobel
- [-5] -> Detekce hran pomoci prahovani a funkce cvSobel
- [-6] -> Detekce hran pomoci nalezeni optimalniho prahu
- [-7] -> Adaptivni prahovani
- [-8] -> Metoda Otsu pro ziskani idealniho prahu
- [-9] -> Zobrazeni pouzite masky
- [-10] -> Zobrazeni histogramu
- [-11] -> Upraveny termosnimek
- [-12] -> Ziskani vysledku pomoci kombinace vice metod

## 5.5 Činnost programu

Činnost programu lze rozdělit do dvou částí, první z nich má za úkol detekovat obrys ruky v termosnímky a druhá získat z termosnímků co nejlepší obrázek celé ruky, tedy zobrazit jen ty informace, které se ruky týkají.

Program čte z textového souboru, kde je seznam termosnímků. Snímky ze souboru prochází po jednom, vždy na základě stisknutí klávesy. Na jednotlivé snímky poté program aplikuje metodu podle zvoleného přepínače. Tuto metodu není možné za chodu programu měnit. Program je ukončen buď poté, co projde všechny snímky ze seznamu, nebo kdykoli pomocí CTR+C. Program je funkční pouze pod operačním systémem Windows.

## 5.6 Součásti programu

Součástí programu jsou všechny zdrojové kódy, databáze snímků, makefile a php skript pro generování textového souboru. Tento skript má za úkol projít složku s obrázky, přejmenovat je a vytvořit textový soubor s jejich názvy, podle kterého je bude program načítat.

## 6 Testování

Celá databáze snímků obsahuje 283 termosnímků od 32 lidí. Po dobu implementace bylo používáno pouze 99 snímků. Ostatní snímky byly do programu přidány až po jeho dokončení. Na těchto snímcích byla ověřena funkčnost programu. Testy uvedené v této práci využívají kompletní databázi, tedy všech 283 termosnímků.

### 6.1 Způsob testování

Jednotlivé metody byly testovány samostatně i v kombinaci s jinými metodami na všech 283 snímcích. Výsledky byly průběžně ukládány a porovnávány. Přehled metod a jejich kombinací je uveden následující kapitole.

### 6.2 Testování jednotlivých metod

**Metody hledající optimální masku:**

- Adaptivní prahování
- Otsu algoritmus pro hledání optimálního prahu
- Vlastní metoda pro hledání optimálního prahu

**Metody pro detekci hran v obrázku**

- Sobelův operátor pro detekci hran – součet hodnot, prahování
- Sobelův operátor pro detekci hran 2 – prahování každého směru zvlášť a následné sčítání získaných hodnot
- Cannyho operátor pro detekci hran
- cvFindContours()
- Vlastní metoda pro hledání optimálního prahu – prahování, Cannyho operátor

### 6.3 Testování kombinací více metod

V kombinaci metod bylo použito všech pěti implementovaných metod pro detekci hran v obrázku. Postupně byla testována shoda jednotlivých bodů na jednotlivých obrázcích, od 25% shody (shoda hodnoty pixelu minimálně ve dvou obrázcích) až po 100% shody (shoda hodnoty pixelu ve všech čtyřech získaných obrázcích).

## 6.4 Výsledky testování

Všechny jednotlivé metody jsou demonstrovány na obrázcích 6.1 a 6.2. Metoda použitá pro tvorbu masky je v závěru testování porovnána s výstupem QuickReportu.



*Obrázek 6.1: Vhodný snímek*



*Obrázek 6.2: Nevhodný snímek*

### 6.4.1 Testování metod hledajících optimální práh pro vytvoření masky

- **Adaptivní prahování**

Metoda, která nepřinesla žádné vhodné výsledky. Výstupem této metody byl obrázek, kde sice došlo k částečnému získání požadované informace, ale okolní šum byl zastoupen v takové míře, že nebylo možné obrázek dále využít.



*Obrázek 6.3: Vhodný snímek*



*Obrázek 6.4: Nevhodný snímek*

- **Otsu algoritmus pro hledání optimálního prahu**

Známý algoritmus, rozděluje histogram na dvě třídy, a poté hledající optimální práh. Tato metoda se ukázala jako celkem přesná, například ve srovnání s mojí vlastní metodou vykazuje ve většině případů srovnatelné nebo lehce horší výsledky, ovšem v několika případech dával Otsu algoritmus jednoznačně lepší výstup. Vysvětlení viz následující text.



*Obrázek 6.5: Vhodný snímek*



*Obrázek 6.6: Nevhodný snímek*

- **Vlastní metoda pro hledání optimálního prahu**

Moje metoda vychází z Otsu algoritmu, proto jsem očekával podobné výsledky. Metoda byla ve většině případů přesnější než Otsu algoritmus, ale na nevhodných snímcích nebyla schopná nic detekovat na rozdíl od Otsu algoritmu. Tyto chyby jsou s největší pravděpodobností způsobeny tím, že při výpočtu prahu neuvažuji hned všech 255 hodnot jako Otsu, ale jen 32, které potom násobím 8x, čímž nemohu vždy dosáhnout stejného prahu. Při spuštění programu můžeme porovnat velikosti obou prahových hodnot a uvidíme, že rozdíl je většinou opravdu minimální, bohužel tomu tak není vždy. I přesto byla tato metoda byla nakonec použita pro vytvoření masky.



*Obrázek 6.7: Vhodný snímek*



*Obrázek 6.8: Nevhodný snímek*

- **Vymaskování snímku**

Porovnání metody využívající hledání optimálního prahu a výstupu softwaru dodávaného k termokameře.



Obrázek 6.9: Snímek upravený vlastní metodou



Obrázek 6.10: FLIR

## 6.4.2 Metody pro detekci hran v obrázku

- **Sobelův operátor pro detekci hran**

Sobelův operátor přinesl jedny z nejlepších výsledků a dokázal i z nevhodných snímků získat hodnotná data. Velmi často se však stávalo, že ačkoli dokázal detekovat velké množství hran, nebyla detekovaná ruka spojitá, tedy obrys ruky nebyl tvořen souvislou křivkou, ale několika na sebe nenavazujícími (obr. 6.11). To znemožňovalo jednoznačně oddělit vnitřní část ruky například jinou barvu, a tím ji od pozadí úplně odlišit.



Obrázek 6.10: Vhodný snímek



Obrázek 6.11: Nevhodný snímek

- **Sobelův operátor pro detekci hran 2**

Obdobné výsledky jako Sobelův operátor, ale s tím rozdílem, že díky prahování, které probíhá před součtem hodnot, jsou některé informace ztraceny a není dosaženo tak přesných výsledků jako u klasického použití Sobelova operátoru. Na druhou stranu díky této ztrátě nejsou zobrazovány drobné chyby vyskytující se ve snímku.

Tato metoda nemá samostatně lepší využití než Sobelův operátor, a proto je spíše vhodná do kombinace s jinou metodou, kde by mohla přispět k separaci chyb.



*Obrázek 6.12: Vhodný snímek*



*Obrázek 6.13: Nevhodný snímek*

- **Cannyho operátor pro detekci hran**

Cannyho operátor dokázal u testovaných snímků nalézt největší množství hran, což by na jednu stranu bylo považováno za úspěch, ale na stranu druhou je to problém, protože byly detekovány i hrany, které lze obecně prohlásit za nežádoucí.



*Obrázek 6.14: Vhodný snímek*



*Obrázek 6.1: Nevhodný snímek*

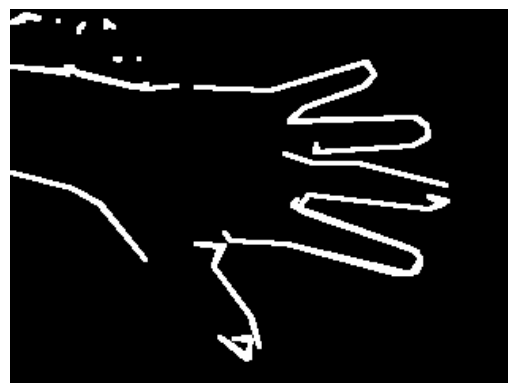


- **Detekce hran pomocí hledání výrazných kontur**

Kombinace vestavěných funkcí pro detekci hran dokázala detekovat obrys ruky pravděpodobně nejspojitěji ze všech uvedených vestavěných funkcí, bohužel již samotné proložení není nejlepší, a to díky velkému počtu jednotlivých kontur, díky kterým se stává obrázek „hranatějším“. Další nevýhodou je poměrně velké zvýraznění hran kolem obrázku. Tyto hrany jsou považovány za rušivé a nežádoucí.



*Obrázek 6.16: Vhodný snímek*



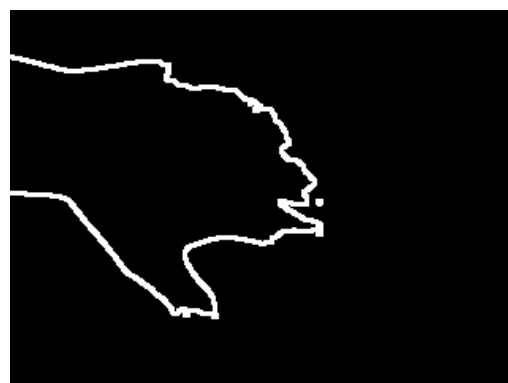
*Obrázek 6.17: Nevhodný snímek*

- **Detekce hran pomocí kombinace optimálního prahu a Cannyho operátoru**

Tato metoda vychází z metody pro nalezení optimálního prahu. U vhodných snímků je zde detekována spojitá kontury ruky. Velkou výhodou této metody je to, že při jejím použití jsou téměř vždy odstraněny nežádoucí hrany z okolí ruky, protože při hledání kontury ruky je použit již naprahovaný obrázek, použitý pro masku viz kapitola 6.4.1.



*Obrázek 6.18: Vhodný snímek*



*Obrázek 6.19: Nevhodný snímek*

### 6.4.3 Testování kombinací metod

V následující kapitole budou popsány a testovány jednotlivé kombinace metod a budou zde shrnuty výsledky těchto testů. Testy kombinací těchto metod proběhly na kompletní databázi a u každé kombinace jsou uvedeny vždy dva statistické výsledky. První z nich uvádí, v kolika případech byl vykreslen spojitý obrys ruky. Druhý potom, kdy byly separovány všechny vnější rušivé body.

Odevzdaný program je přeložen s nastavením na testování všech pěti metod a při použití 62% shody v bodě. Tato hodnota byla spočítána jako optimální poměr spojitosti kontury ruky a chybovosti jejího okolí, viz kapitola 6.4.3.6. Hodnota udávající shodu je implementována jako konstanta THRESH, tato hodnota není uživatelsky měnitelná bez zásahu do zdrojového kódu a následného překladu. Stejně jako není možné měnit počet testovaných metod, to jde opět jen úpravou kódu a novým překladem.

#### 6.4.3.1 2 metody se 100% shodou v daném bodě

Tato kombinace popisuje porovnání dvou metod se 100% shodou v daném bodě, tedy bod je prohlášen za platný pouze v případě, že bude na obou snímcích. Tento způsob testování má primárně za úkol oddělit části kontury, které dokáže vykreslit každá metoda, od částí, které mohou být považovány za nežádoucí body. To, že obrys ruky není spojitý, je v tomto testu zanedbáno. Jako jednotlivé dvojice jsem zvolil tyto: viz tabulka 6.1.

Tabulka 6.1: Výsledky 2 metod se 100% shodou v bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Contour	13%	1%
Cannyho operátor / Sobelův operátor	39%	3%
Cannyho operátor / Sobelův operátor 2	69%	3%
Cannyho operátor / Optimální práh	97%	42%
Contour / Sobelův operátor	42%	30%
Contour / Sobelův operátor 2	72%	1%

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Contour / Optimální práh	97%	31%
Sobelův operátor / Sobelův operátor 2	65%	2%
Sobelův operátor / Optimální práh	97%	26%
Sobelův operátor 2 / Optimální práh	100%	2%

#### 6.4.3.2 2 metody s 0% shodou v daném bodě

Jedná se o sjednocení dvou snímků. Požadovaným výsledkem tohoto testu je vykreslení spojitě kontury celé ruky. Okolí obrázku bude zanedbáno.

Tabulka 6.2: Výsledky 2 metod s 0% shodou v bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Contour	5%	65%
Cannyho operátor / Sobelův operátor	3%	70%
Cannyho operátor / Sobelův operátor 2	4%	66%
Cannyho operátor / Optimální práh	16%	58%
Contour / Sobelův operátor	9%	66%
Contour / Sobelův operátor 2	11%	68%
Contour / Optimální práh	5%	44%
Sobelův operátor / Sobelův operátor 2	19%	21%
Sobelův operátor / Optimální práh	26%	70%
Sobelův operátor 2 / Optimální práh	65%	72%

#### 6.4.3.3 3 metody se 100% shodou v daném bodě

Kombinace obdobná jako v předcházejícím případě, avšak jen s tím rozdílem, že jsou porovnávány 3 metody. Hledaným výsledkem je počet snímků, které neobsahují žádné rušivé části.

Tabulka 6.3: Výsledky 3 metod se 100% shodou v bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Contour / Cannyho operátor / Sobelův operátor	42%	18%
Contour / Cannyho operátor / Sobelův operátor 2	70%	2%
Contour / Cannyho operátor / Optimální práh	97%	29%
Contour / Sobelův operátor / Sobelův operátor 2	68%	1%
Contour / Sobelův operátor / Optimální práh	97%	17%
Contour / Sobelův operátor 2 / Optimální práh	100%	1%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2	67%	1%
Cannyho operátor / Sobelův operátor / Optimální práh	98%	17%
Cannyho operátor / Sobelův operátor 2 / Optimální práh	100%	2%
Sobelův operátor / Sobelův operátor 2 / Optimální práh	100%	2%

Tato kombinace se ukázala jako výhodná zejména pro separaci rušivých částí, zejména při kombinacích, kde je použita metoda s využitím optimálního prahu. Ovšem vzhledem k detekci spojitého obrysu dosahuje velmi špatných výsledků.

#### 6.4.3.4 3 metody s 66% shodou v daném bodě

Kombinace uvažující shodu alespoň 2 ze 3 daných bodů. Tedy v případě, že bude shoda pixelu alespoň ve dvou případech, je tomuto pixelu přiřazena daná barva. Tento test nepodal jednoznačné výsledky o metodách, které je vhodné při tomto nastavení shody použít při hledání spojitosti, ani o metodách, které budeme aplikovat, budeme-li hledat chyby v okolí ruky.

Tabulka 6.4: Výsledky 3 metod s 66% shodou v bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Contour / Cannyho operátor / Sobelův operátor	10%	61%
Contour / Cannyho operátor / Sobelův operátor 2	12%	57%
Contour / Cannyho operátor / Optimální práh	10%	43%
Contour / Sobelův operátor / Sobelův operátor 2	42%	31%
Contour / Sobelův operátor / Optimální práh	42%	61%
Contour / Sobelův operátor 2 / Optimální práh	68%	57%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2	34%	25%
Cannyho operátor / Sobelův operátor / Optimální práh	41%	59%
Cannyho operátor / Sobelův operátor 2 / Optimální práh	67%	50%
Sobelův operátor / Sobelův operátor 2 / Optimální práh	70%	23%

#### 6.4.3.5 3 metody s 0% shodou v daném bodě

Tabulka 6.5: Výsledky 3 metod s 0% shodou v bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Contour / Cannyho operátor / Sobelův operátor	2%	63%
Contour / Cannyho operátor / Sobelův operátor 2	3%	61%
Contour / Cannyho operátor / Optimální práh	2%	74%
Contour / Sobelův operátor / Sobelův operátor 2	8%	64%
Contour / Sobelův operátor / Optimální práh	9%	72%

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Contour / Sobelův operátor 2 / Optimální práh	10%	73%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2	2%	60%
Cannyho operátor / Sobelův operátor / Optimální práh	3%	72%
Cannyho operátor / Sobelův operátor 2 / Optimální práh	5%	69%
Sobelův operátor / Sobelův operátor 2 / Optimální práh	27%	30%

Podle tabulky výsledků je zřejmé, že shoda ve 33% se má při použití kombinace 3 metod kladný dopad na spojitost kontury, oproti tomu se na snímku vyskytuje mnoho rušivých částí.

#### 6.4.3.6 4 metody se 100% shodou v daném bodě

Tabulka 6.6: Výsledky 4 metod se 100% shodou v daném bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2	68%	2%
Cannyho operátor / Sobelův operátor / Contour / Optimální práh	2%	15%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2 / Optimální práh	0%	2%
Contour / Sobelův operátor / Sobelův operátor 2 / Optimální práh	0%	3%

#### 6.4.3.7 4 metody se 75% shodou v daném bodě

Tabulka 6.7: Výsledky 4 metod s 75% shodou v daném bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2	42%	25%

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Optimální práh	45%	63%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2 / Optimální práh	44%	58%
Contour / Sobelův operátor / Sobelův operátor 2 / Optimální práh	60%	26%

#### 6.4.3.8 4 metody se 50% shodou v daném bodě

Tabulka 6.8: Výsledky 4 metod se 50% shodou v daném bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2	11%	62%
Cannyho operátor / Sobelův operátor / Contour / Optimální práh	13%	63%
Cannyho operátor / Sobelův operátor / Sobelův operátor 2 / Optimální práh	39%	61%
Contour / Sobelův operátor / Sobelův operátor 2 / Optimální práh	42%	62%

Hodnota 50% shody v bodě nebyla jednoznačná ani pro spojitost nebo chybovost, proto lze předpokládat, že se hodnota shody zvolená pro vyvážený poměr spojitosti a chybovosti bude pohybovat právě kolem hodnoty 50%.

#### 6.4.3.9 4 metody se 0% shodou v daném bodě

Tabulka 6.9: Výsledky 4 metod se 0% shodou v daném bodě

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2	4%	65%
Cannyho operátor / Sobelův operátor / Contour / Optimální práh	3%	78%

	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Sobelův operátor 2 / Optimální práh	3%	72%
Contour / Sobelův operátor / Sobelův operátor 2 / Optimální práh	9%	70%

Při nastavení této shody došlo ke sjednocení všech bodů z každého obrázku, výsledný obrázek byl tedy zaměřen zejména na spojitost.

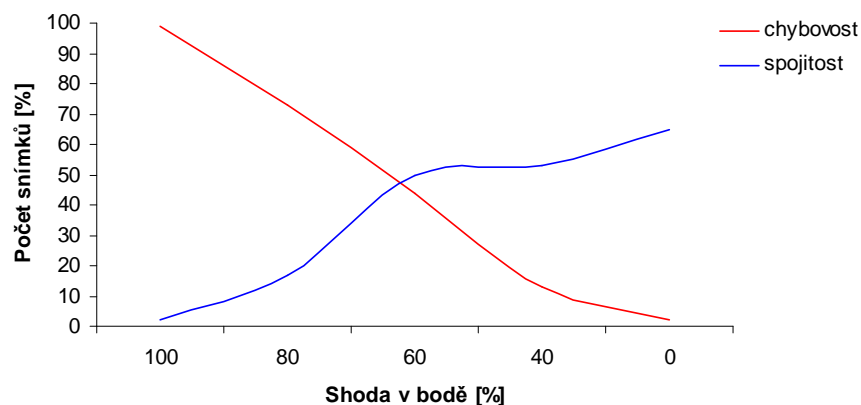
#### 6.4.3.10 5 metod s danou shodou

Tabulka 6.10: Výsledky 5 metod s danou shodou

	shoda	Počet snímků, které neobsahují žádné vnější rušivé části	Počet snímků, kde byl detekován spojitý obrys ruky
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2 / Optimální práh	100%	99%	2%
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2 / Optimální práh	80%	73%	17%
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2 / Optimální práh	60%	44%	50%
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2 / Optimální práh	40%	13%	53%
Cannyho operátor / Sobelův operátor / Contour / Sobelův operátor 2 / Optimální práh	0%	2%	63%

Pro poslední test byla zvolena kombinace všech pěti metod. Požadovaným výsledkem bylo shromáždit taková data, aby bylo možné na základě těchto dat zjistit ideální hodnotu shody v bodě, pro optimální poměr spojitosti a chybovosti při použití všech pěti metod.





*Graf 6.1: Závislost chybovosti a spojitosti*

Na základě získaných dat je zřejmé, že mezi spojitostí a chybovostí platí nepřímá úměrnost, tedy čím více bude kontura ruky spojitá, tím větší bude rušení v pozadí (kolem ruky) a naopak.

Za pomoci těchto získaných dat byl vytvořen graf (6.1). Z tohoto grafu je odečtena hodnota udávající optimální poměr mezi chybovostí a spojitostí. V našem případě je hodnota stanovena na 62% shodu v daném bodě.

## 7 Závěr

Využití termosnímků ruky je z hlediska biometrie poměrně novou a ne příliš používanou metodou. Tato práce se snaží shrnout toto téma co nejobecněji. Vyzkoušet různé metody nebo jejich kombinace pro detekci ruky v termosnímku a porovnat jednotlivé výsledky.

V první části se práce zabývá hledáním hodnoty optimálního prahu, který je potřeba určit. Tento nalezený práh nám pomůže při tvorbě masky, kterou bude originální obrázek překryt, a tím z něj oddělíme okolní části, které nejsou na snímku důležité. Při hledání tohoto prahu byly použity jak běžné metody jako globální prahování, adaptivní prahování aj., tak i jedna vlastní, která byla vyvinuta na základě známých metod a pro tuto aplikaci poskytovala nejlepší výsledky, Proto byla tato metoda nakonec použita pro tvorbu masky.

V další část práce se zabývá detekcí ruky, respektive jejího obrysu – kontury. Ovšem kromě nalezení této kontury bylo důležité odstranit okolní nežádoucí hrany. Tyto hrany jsou buď detekovány jednotlivými metodami, nebo jejich kombinacemi. Při použití jednotlivých metod ovšem nebylo dosaženo tak kvalitních výsledků jako při použití kombinací. Cílem těchto kombinací metod bylo nalézt optimální hodnotu shody v daném pixelu, tak aby byla zachována spojitost kontury a přitom rušení kolem ruky bylo přijatelné. Pro nalezení této hodnoty byla použita kombinace všech pěti implementovaných metod. Zvolená shoda byla potom určena na základě provedených měření, její hodnota byla stanovena na 62%.

V nasnímané databázi se ovšem vyskytuje i několik sad snímků, kde je kontura ruky kvůli její teplotě špatně detekovatelná, nebo není detekovatelná vůbec. Tento problém by šel řešit již při samotném získávání snímku a to tak, že by snímky byly snímány na výrazně chladnějším podkladu, čímž by byl zaručen vyšší kontrast ruky na snímku. Pokud bychom chtěli tento problém řešit programově, tak by bylo možné použít například metodu ASM (Active Shape Model), která je založena na porovnávání získané části obrazu se sadou modelů reprezentujících lidskou ruku v různých natočeních. Pomocí této metody by bylo možné doplnit chybějící části kontury.

Zadání práce obsahovalo 4 dílčí části. První dvě položky, jako nastudování a tvorbu databáze, jsem splnil řádně v zimním semestru jako součást semestrálního projektu. Další dva body, které se týkaly samotných experimentů a shrnutí výsledků, jsem dokončil během letního semestru. Nad rámec zadání byla vytvořena aplikace, pomocí které je možné porovnat jednotlivé metody.

# Literatura

- [1] *Emissivity* [online], last modified on 27 December 2009 at 09:38 [cit. 13. 12. 2009], Wikipedia. Dostupné z WWW: < <http://en.wikipedia.org/wiki/Emissivity>>
- [2] SOCOLINSKY, D. A., SELINGER, A.: *Thermal Face Recognition Over Time*, In: Proceedings of ICPR'04, IEEE CS, 2004, pp. 187 – 190, ISBN 0-7695-2128-2.
- [3] *Pyrometer* [online], last modified on 24 September 2009 at 18:34 [cit. 19. 12. 2009], Wikipedie. Dostupné z WWW: < <http://en.wikipedia.org/wiki/Pyrometer>>
- [4] *Pyrometr* [online],[cit. 19. 12. 2009], Dostupné z WWW: < <http://www.technotrend.cz/stranky/teorie/teorie3.htm> >
- [5] *Termokamery FLIR řady E* [online], [cit. 27. 12. 2009], Dostupné z WWW: < <http://www.tmvss.cz/Aplikace/Termovize/Energetika/rada-e.html>>
- [6] *Teorie termovizního měření* [online], [cit. 27. 12. 2009], Dostupné z WWW: < <http://www.termovize.com/princip-termovize/>>
- [7] VÁŇA, J: *Face recognition by thermal image*, Dokument dostupný na URL <http://www.feec.vutbr.cz/EEICT/2008/sbornik/02-Magisterske%20projekty/10-Intelligentni%20systemy/08-xvanaj00.pdf> (březen 2009)
- [8] *Thermography* [online], last modified on 15 January 2010 at 13:37 [cit. 13. 12. 2009], Wikipedie. Dostupné z WWW: < <http://en.wikipedia.org/wiki/Thermography>>
- [9] GENČÚR, M. *Nástroj na zpracování fotografovaného textu*, bakalářská práce, Brno, FIT VUT v Brně, 2007
- [10] *Thresholding* [online], 2003, [cit. 27.4.2010] Dostupné z WWW: <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/threshld.htm>>
- [11] *Adaptive Thresholding* [online], 2003, [cit. 27.4.2010] Dostupné z WWW: <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/adpthrsh.htm>>

- [12] *Sobel Edge Detector* [online], 2003, [cit. 21.4.2010] Dostupné z WWW:  
<<http://homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm>>
- [13] *Smoothing a Histogram* [online], 2003, [cit. 21.4.2010] Dostupné z WWW:  
<<http://www.mathworks.com/products/splines/demos.html?file=/products/demos/shipping/splines/histodem.html>>
- [14] *Histogram* [online], 2003, [cit. 21.4.2010] Dostupné z WWW:  
<<http://www.digineff.cz/cojeto/ruzne/histogram.html> >
- [15] *Canny edge detector* [online], last modified on 7 March 2010 at 13:14. [cit. 27. 4. 2010], Wikipedie. Dostupné z WWW: <[http://en.wikipedia.org/wiki/Canny\\_edge\\_detector](http://en.wikipedia.org/wiki/Canny_edge_detector)>
- [16] *Introduction to programming with OpenCV* [online], January 27, 2006 [cit. 3.4.2010], Dostupné z WWW: <<http://www.cs.iit.edu/~agam/cs512/lect-notes/opencv-intro/opencv-intro.html>>
- [17] *Image Processing and Analysis Reference* [online], [cit. 4.4.2010], Dostupné z WWW: <[http://www710.univ-lyon1.fr/~bouakaz/OpenCV-0.9.5/docs/ref/OpenCVRef\\_ImageProcessing.htm](http://www710.univ-lyon1.fr/~bouakaz/OpenCV-0.9.5/docs/ref/OpenCVRef_ImageProcessing.htm)>
- [18] ZEMČÍK, P.: *Vybrané slajdy k předmětu POV*, VUT v Brně, Fakulta informačních technologií, ÚPGM, (2008)
- [19] BRADSKI, G., KAEBLER, A.: *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, Incorporated, Cambridge, 2008. ISBN 978-0-596-51613-0

# Seznam příloh

- A. Výstupy použitých metod
- B. Obsah CD

# Příloha A. Výstupy použitých metod

V této příloze jsou ukázány výstupy některých metod, které jsou v programu použity.

## Metody určené pro vykreslení kontury ruky

### cvFindContours()



Obrázek A.1



Obrázek A.2



Obrázek A.3



Obrázek A.4

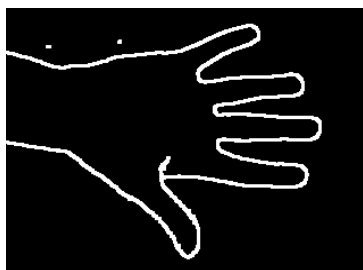


Obrázek A.5



Obrázek A.6

## cvCanny()



Obrázek A.7



Obrázek A.8



Obrázek A.9



Obrázek A.10



Obrázek A.11



Obrázek A.12

## cvSobel()



Obrázek A.13



Obrázek A.14



Obrázek A.15



Obrázek A.16

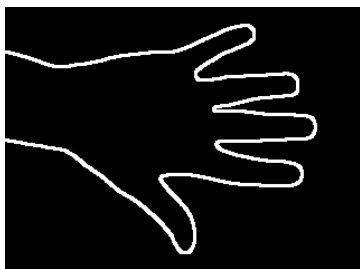


Obrázek A.17

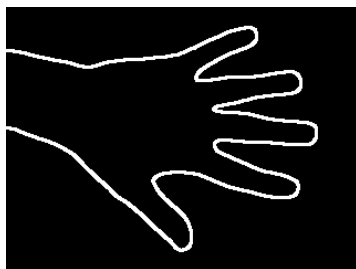


Obrázek A.18

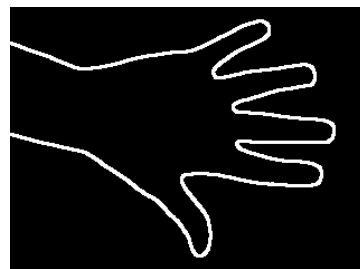
## Vlastní metoda pro detekci kontury za použití optimálního prahu



Obrázek A.19



Obrázek A.20



Obrázek A.21



Obrázek A.22



Obrázek A.23



Obrázek A.24

## Metody určené pro tvorbu masky

### Vlastní metoda pro hledání optimálního prahu



Obrázek A.25



Obrázek A.26



Obrázek A.27



Obrázek A.28



Obrázek A.29



Obrázek A.30



## Adaptivní prahování



Obrázek A.31



Obrázek A.32



Obrázek A.33



Obrázek A.34



Obrázek A.35



Obrázek A.36

## Otsu algoritmus



Obrázek A.37



Obrázek A.38



Obrázek A.39



Obrázek A.40



Obrázek A.41

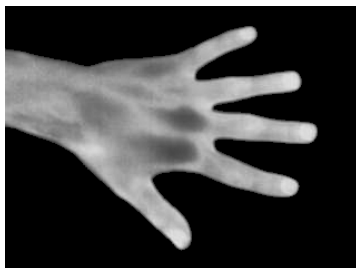


Obrázek A.42

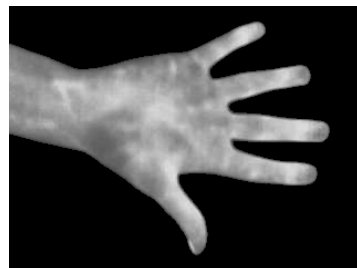
## Aplikace vytvořené masky



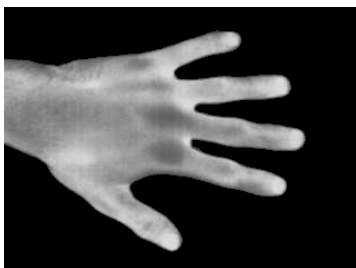
Obrázek A.43



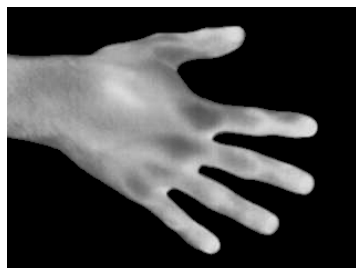
Obrázek A.44



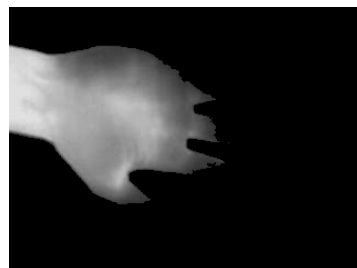
Obrázek A.45



Obrázek A.46



Obrázek A.47



Obrázek A.48

## Příloha B. Obsah CD

- Zdrojové kódy k projektu
- Skript pro generování fileListu
- Databáze snímků
- Seznam použitých snímků – fileList.txt
- Programová dokumentace
- Technická zpráva ve formátu pdf
- Přeložený a spustitelný program včetně dávkových souborů
- Makefile