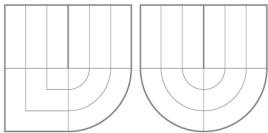
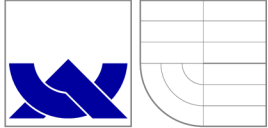
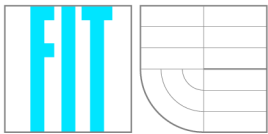


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ



FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

## 3D REKONSTRUKCE OBLIČEJE Z 2D SNÍMKU

3D FACE RECONSTRUCTION BASED ON 2D IMAGE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MARTIN KARHÁNEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETR ČÁSTEK

BRNO 2009

## **Abstrakt**

Tato práce se zabývá možnými přístupy k 3D rekonstrukci obličeje z 2D snímku. Jsou zde popsány způsoby analýzy vstupního snímku, jako lokalizace obličeje a obličejových rysů, ze kterých rekonstrukce vychází. Dále je čtenář detailněji seznámen s modifikačním modelem, který je základem většiny zmíněných metod, jeho vytvořením z databáze statistických dat a použitím při rekonstrukci. Obsahem je také popis implementace algoritmu využívajícího tento model.

## **Abstract**

This work deals with procedures that enables you reconstruct 3D face from a 2D picture. It describes ways to analyze an input picture, such as a face and facial features localization, on which this reconstruction builds. A reader is getting into a morphable model, its creation from static database data and and its usage on a reconstruction, more detailed later on, this morphable model is a building block of mentioned methods. This work contains also an algorithm implementation based on this model.

## **Klíčová slova**

rekonstrukce obličeje, modifikační model, rozpoznávání obličeje, obličej, analýza obličeje, obličejové rysy

## **Keywords**

face reconstruction, morphable model, face recognition, face, face analysis, facial features

## **Citace**

Martin Karhánek: 3D rekonstrukce obličeje z 2D snímku, bakalářská práce, Brno, FIT VUT v Brně, 2009

# 3D rekonstrukce obličeje z 2D snímku

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Petra Částka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Martin Karhánek

18. května 2009

## Poděkování

Tímto bych chtěl poděkovat Ing. Petru Částkovi za odborné vedení a konstruktivní připomínky, které mi při tvorbě této práce pomohly.

© Martin Karhánek, 2009.

*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

<b>1 Úvod</b>	<b>3</b>
<b>2 Analýza obličeje</b>	<b>4</b>
2.1 Detekce obličeje	4
2.1.1 Barevné modely	4
2.1.2 Waveletová transformace	5
2.1.3 Morfologické operace	5
2.1.4 Detekce význačných bodů	6
2.2 Active shape model (ASM)	6
2.3 Active appearance model (AAM)	7
2.4 Support vector machines (SVM)	7
2.5 Analýza hlavních komponent (PCA)	8
<b>3 Rekonstrukce obličeje</b>	<b>9</b>
3.1 3D modifikační model	9
3.1.1 Vytvoření databáze	9
3.1.2 Vytvoření modelu	10
3.1.3 Využití 3D modifikačního modelu při rekonstrukci	10
3.2 Rekonstrukce založená na vzorech	11
3.3 Rekonstrukce kombinací statistických a geometrických informací	13
3.4 Metoda využívající ALLE algoritmus	14
<b>4 Implementace</b>	<b>17</b>
4.1 Návrh řešení	17
4.2 Vytvoření modelu a databáze	18
4.2.1 Model obličeje	18
4.2.2 Databáze	19
4.3 Detekce	19
4.3.1 Hledání kandidátských oblastí	19
4.3.2 Validace obličeje	20
4.3.3 Hledání záchytných bodů	21
4.4 Rekonstrukce	21
4.4.1 Afinní transformace	21
4.4.2 Výpočet odchylky modelu od předlohy	21
4.4.3 Iterační cyklus rekonstrukce	22
4.5 Ovládání aplikace	23

<b>5</b>	<b>Výsledky</b>	<b>24</b>
5.1	Testování	24
5.1.1	Chybná detekce	24
5.1.2	Nedostatečné množství detekovaných bodů	25
5.1.3	Ovlivnění výsledků počtem záznamů v DB	25
5.2	Manuální označení záchytných bodů	26
<b>6</b>	<b>Závěr</b>	<b>28</b>
<b>A</b>	<b>Obsah CD</b>	<b>31</b>

# Kapitola 1

## Úvod

Rekonstrukce obličeje v 3D prostoru z jeho dvourozměrné projekce je jedním z problémů počítačového vidění a zpracování obrazu. Jde o obnovení informace o hloubce určité scény tak, aby co nejvíce odpovídala skutečnosti. K tomu je ovšem kromě 2D snímku získat další informace. Teoreticky může k rekonstrukci pomoci např. více snímků z různých úhlů. Další možností, kterou využívá většina současných metod, je vytvoření databáze statisticky získaných informací, pomocí které je odhadnut nejpravděpodobnější vzhled rekonstruovaného obličeje.

Pokud požadujeme jistou míru automatizace rekonstrukce, je nutné se zabývat i algoritmy pro analýzu vstupního snímku, detekci obličeje, apod. Proto se tato práce z části zabývá i těmito tématy.

Lidských obličejů existuje nespočet a liší se tvarem i barvou o změnách vlivem emocí nemluvě. Při rekonstrukci se vyskytuje řada problémů, které ji mohou značně ztížit. Ať už jde o nízkou kvalitu pořízeného snímku, či o situace, kdy je osoba nalíčená, případně používá nejrůznější módní doplňky. Pokud navíc chceme, aby byla aplikace co nejuniverzálnější, musíme vzít v úvahu i nejrůznější úhly natočení obličeje vůči kameře.

Své uplatnění může rekonstrukce obličeje najít např. v kriminalistice při identifikaci osob. Použití je ovšem možné i v komerční sféře, ať už při vytváření her, či ve filmovém průmyslu, kde se v poslední době rozvíjí trend „virtuálních“ herců. Zde by mohla tato problematika značně ulehčit práci při vytváření modelů.

Druhá kapitola popisuje metody pro analýzu obličeje. Je zde zmíněn postup detekce obličeje samotného i některých důležitých rysů, jako jsou rty a oči. Dále jsou stručně popsány statistické modely ASM a AAM, klasifikační algoritmus SVM a metoda strojového učení PCA, použitelné pro lokalizaci a získání dalších informací o obličejích.

V kapitole třetí jsou popsány možné postupy při rekonstrukci obličeje. Čtenář je obeznámen s modifikačním modelem, jeho vytvořením i použitím při rekonstrukci. Dále je zde uveden postup rekonstrukce hloubky obrazu pomocí statisticky získaných vzorů a rekonstrukce pomocí kombinace statistických a geometrických informací. V poslední části je popsáno využití ALLE algoritmu.

Kapitola čtvrtá popisuje implementaci zvolené metody, problémy, které při implementaci vznikaly a jejich řešení. Jsou zde zmíněna omezení použitých algoritmů.

Poslední kapitola shrnuje dosažené výsledky, popisuje chyby implementovaných algoritmů, jejich vznik a důsledky.

## Kapitola 2

# Analýza obličej

Jelikož je třeba před samotnou rekonstrukcí obličej analyzovat a zjistit o něm co nejvíce informací, aby byla rekonstrukce co nejpřesnější, zmíním se stručně o postupech, jakými lze detekovat v obraze obličej a jeho důležité rysy. Hned první částí je samotná lokalizace obličeje. Dále jsou zde popsány postupy pro detekci některých výrazných částí obličeje, statistické modely a klasifikační algoritmy.

### 2.1 Detekce obličej

Základní metodou zjištění oblasti, ve které se v obraze nachází obličej, je analýza podle barvy. Barva lidské kůže zaujímá v prostoru barev určitý ohraničený podprostor. Pokud budeme mít o tomto podprostoru informace, můžeme určit, zda je daný pixel součástí obličeje, či ne.

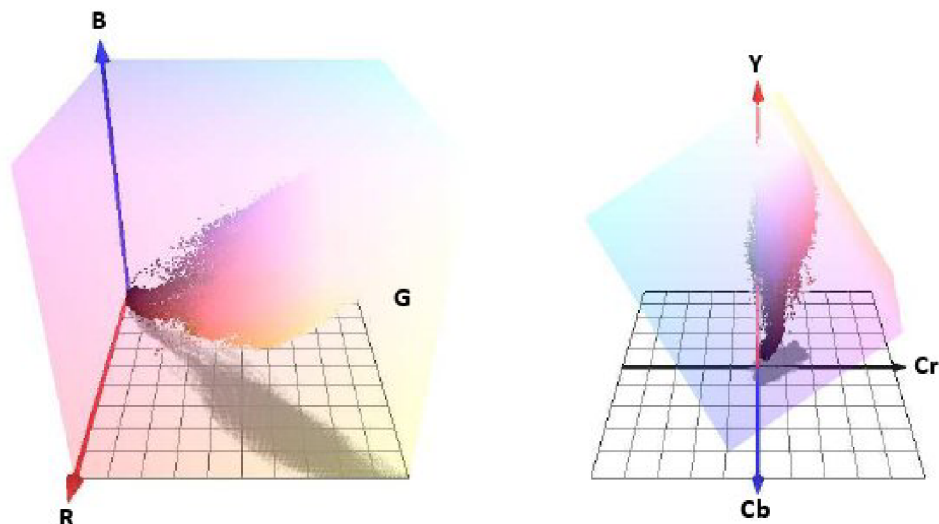
Výhodou této metody je rychlost a účinnost. Selhává ovšem v situaci, kdy se v pozadí nachází objekty se stejnou barvou, jakou má obličej. I přesto ji však lze použít u obrazů s velmi komplexním pozadím, pokud ji zkombinujeme s některou z dalších metod a zvýšíme tím její přesnost [9].

#### 2.1.1 Barevné modely

Prvním z kroků je výběr barevného modelu. Nejznámějším barevným modelem je RGB, který definuje barvu třemi složkami (červená, modrá zelená). Vhodnějším pro implementace je však model YCbCr, který odděluje barevné složky (Cb a Cr) a složku jasovou (Y). Převodní vztah mezi těmito modely vypadá následovně [9].

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65,481 & 128,553 & 24,966 \\ -37,797 & -74,203 & 112,000 \\ 112,000 & -93,786 & -18,214 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.1)$$

Dále je třeba nadefinovat podprostor barev, které odpovídají kůži. To lze explicitně vyjádřit na základě rozložení vzorových barev kůže. Modely rozložení barvy kůže mohou být neparametrické, tj. rozložení barvy je získáno z trénovací množiny bez vyjádření pomocí pravidel, nebo parametrické, které se snaží popsat rozložení barev matematickým modelem [9].



**Obrázek 2.1:** Podprostor barvy kůže v modelu RGB (vlevo) a v modelu YCbCr (vpravo). Převzato z [9].

### 2.1.2 Waveletová transformace

Po aplikování předchozího postupu se v pravděpodobnostní mapě mohou vyskytovat oblasti mimo obličejovou část. Zlepšit výsledky lze použitím waveletové transformace jasové složky obrazu.

Předpokládáme, že v oblasti očí, rtů, apod. jsou ve vertikálním směru rozpoznatelné výrazné přechody jasu. Použijeme tedy detailní koeficienty waveletové transformace ve vertikálním směru prvního stupně rozkladu za použití Daubeschiesové vlny 2. typu. Za účelem pokrytí větší části obličeje použijeme dále dolní propust a důležité koeficienty roztáhneme tentokrát vertikálně, použijeme tedy speciální průměrovací funkci sinusoidního tvaru [11].

### 2.1.3 Morfologické operace

I po použití waveletové transformace mohou v mapě zůstat body, které nenáležejí oblasti s obličejem. Další možností jak tyto chyby odstranit je použití některých z morfologických operací. Těchto operací je celá řada. Vstupní obraz je transformován tak, že každý pixel je upraven podle hodnot pixelů v jeho okolí [11].

Pro odfiltrování osamocených bodů a spojení blízkých bodů je možné použít operace *otevření* (2.2) a *uzavření* (2.3). Otevření zjednodušuje strukturu objektu a rozděluje ho v úzkých místech, naopak uzavření spojuje blízké objekty a vyhlazuje jejich tvar.

$$X \bullet B = (X \oplus B) \ominus B, \quad (2.2)$$

$$X \circ B = (X \ominus B) \oplus B, \quad (2.3)$$

kde  $\oplus$  je operace *dilatace* (2.4) a  $\ominus$  je operace *eroze* (2.5) [11].

$$X \oplus B = \{p \in \varepsilon^2 : p = x + b, x \in X, b \in B\} \quad (2.4)$$

$$X \ominus B = \{p \in \varepsilon^2 : p + b \in X, \forall b \in B\} \quad (2.5)$$



### 2.1.4 Detekce význačných bodů

Výstupem metody popsané v části 2.1 může být několik kandidátských oblastí. Abychom mohli s větší jistotou říct, ve kterém případě se jedná skutečně o obličej, můžeme se pokusit dané oblasti verifikovat např. tím, že se pokusíme detekovat další významné body jako oči a ústa.

#### Rty

Pokud pomíneme nejrůznější nepřirozené úpravy líčením, vyznačují se ústa tím, že je jejich barva tvořena vysokou hodnotou červené a velmi nízkou hodnotou modré. V prvním kroku tedy provedeme barevnou transformaci *FLD* (2.6) [11]. Tato bohužel nepokryje všechny případy, z toho důvodu [11] zavádí navíc transformaci *g* (2.7), která bude *FLD* doplňovat.

$$FLD = [-0,289 \ 0,379 \ 0,038] \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.6)$$

$$g = \frac{G}{(R + G + B)} \quad (2.7)$$

Výsledkem těchto transformací jsou mapy, ve kterých dosahuje oblast úst nízkých hodnot. Kvůli eliminaci případů, kdy se červená barva objevuje v obličejí i na jiných místech, můžeme podobně jako v 2.1 použít waveletovou transformaci. Pokud obě mapy sloučíme pomocí vztahu (2.8) a použijeme algoritmus *mean shift*, získáme souřadnice oblasti, která s velkou pravděpodobností odpovídá rtům.

$$map = \frac{1}{4} \cdot (map1 + map2) + \frac{3}{4} \cdot map1 \cdot map2 \quad (2.8)$$

#### Oči

Při detekci očí můžeme postupovat podobně jako u rtů. Oblast očí je charakteristická barvou bělma a zornice, které se u většiny lidí neliší. Použijeme tedy transformaci (2.9) k vytvoření mapy, ve které dosahují oblasti očí vysokých hodnot. Tuto mapu opět upravíme waveletovou transformací a prahováním do binární podoby. Pokud je výsledkem více oblastí, např. byly detekovány zuby, vybereme ty nejpravděpodobnější podle jejich umístění vzhledem k obličejí.

$$map = \frac{1}{3}C_b^2 + (256 - C_r)^2 + \frac{C_b}{C_r} \quad (2.9)$$

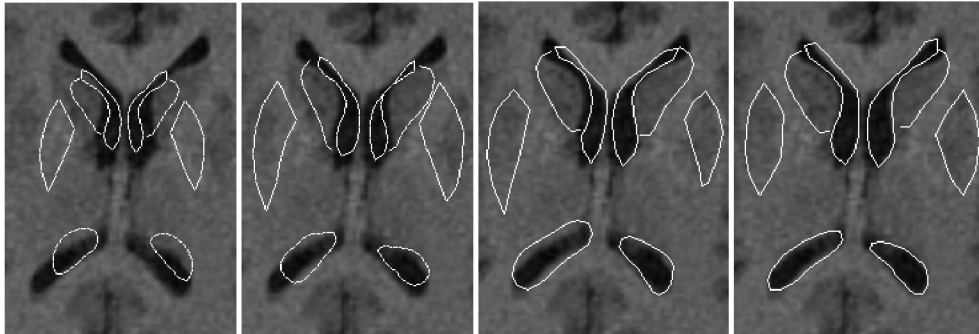
## 2.2 Active shape model (ASM)

Jde o statistický model popisující tvar objektu, který je postupně deformován tak, aby odpovídal objektu na obrázku. Tvar objektu je popsán množinou bodů a je omezen statistickým modelem PDM<sup>1</sup>, který zajišťuje, aby jeho tvar odpovídal objektům z trénovací

<sup>1</sup>point distribution model – model získaný statistickou analýzou několika vzorů určitého objektu

množiny. Algoritmus úpravy modelu v každém kroku analyzuje okolí bodů modelu a podle toho upravuje jejich pozici. Na obrázku 2.2 je vidět postupná změna modelu vzhledem k objektu na obrázku. Iterativní postup pro zlepšení pozice bodů vypadá následovně [6]:

1. Prozkoumej oblast obrázku v okolí bodu  $\mathbf{X}_i$  a najdi nejlepší blízkou shodu pro bod  $\mathbf{X}'_i$
2. Aktualizuj parametry  $(X_t, Y_t, s, \theta, \mathbf{b})$
3. Opakuj postup dokud konverguje



**Obrázek 2.2:** Příklad konvergence modelu k objektu na obrázku (zleva: výchozí stav, po 1 iteraci, po 6 iteracích, po 12 iteracích). Převzato z [5].

V praxi např. předpokládáme, že hranice modelu odpovídají hranám v obrázku. Můžeme tedy jednoduše lokalizovat nejsilnější hranu podél profilu k získání nové pozice bodu [5].

### 2.3 Active appearance model (AAM)

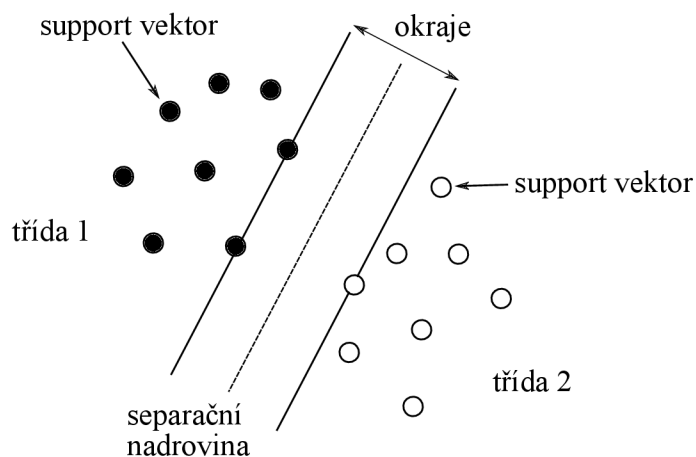
Stejně jako v předchozím případě jde o statistický model. Na rozdíl od Active Shape modelu, který manipuluje pouze s tvarem modelu, využívá i další informaci získatelnou ze zdrojového obrázku, texturu. Pro nalezení nejlepší shody mezi modelem a obrázkem je třeba minimalizovat velikost rozdílového vektoru  $|\Delta\mathbf{I}|^2$  změnami parametrů modelu. Rozdílový vektor  $\delta\mathbf{I}$  může být definován následovně:

$$\delta\mathbf{I} = \mathbf{I}_i - \mathbf{I}_m, \quad (2.10)$$

kde  $\mathbf{I}_i$  je vektor hodnot úrovní šedi v obrázku a  $\mathbf{I}_m$  je vektor hodnot pro současné parametry modelu[4].

### 2.4 Support vector machines (SVM)

Tato metoda provádí klasifikaci dat vytvořením separační N-dimenzionální nadroviny. Tato nadrovina optimálně rozděluje data do dvou kategorií. K výpočtu okraje mezi nimi jsou zkonstruovány dvě paralelní nadroviny, na každé straně separační nadroviny jedna. Čím větší je potom vzdálenost každé z nadrovin od bodů z opačné kategorie, tím menší je chyba klasifikace [13]. SVM modely jsou příbuzné s vícevrstevnými perceptronovými neuronovými sítěmi.



**Obrázek 2.3:** Příklad SVM klasifikace v 2D prostoru

## 2.5 Analýza hlavních komponent (PCA)

PCA je metoda strojového učení založená na lineárních vzhledových klasifikátorech. Lze ji využít k rychlé a jednoduché aproximaci složitějších objektů, případně pro aproximaci hodnoty velikosti objektů, jelikož jejím účelem je především nalezení obálky obklopující skupinu bodů [9].

Existuje několik způsobů vytvoření obálky kolem bodů ve mnohorozměrném prostoru. Jedním z nich je metoda AABB (Axis Align Bounding Box), která využívá vektory kolmé na osy souřadného systému. Je ovšem použitelná pouze u objektů, které jsou soustředěny kolem jednoho bodu. Metoda OBB využívá vektory, které nemusí být nutně kolmé na osy. Lze tedy dosáhnout lepší aproximace objektu [9].

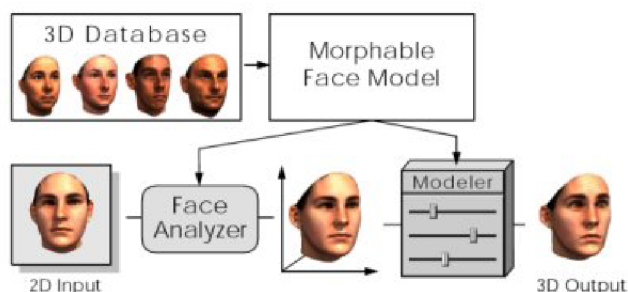
## Kapitola 3

# Rekonstrukce obličeje

V této kapitole se zaměřím na samotný postup rekonstrukce. Většina dále popsaných algoritmů vychází z *3D modifikačního modelu*, který je podrobněji popsán hned v první části, včetně postupu při jeho vytváření.

### 3.1 3D modifikační model

Volker Blanz a Thomas Vetter představili v [1] metodu, která umožňuje uživateli modelovat velké množství různých obličejů jednoduchou změnou několika koeficientů. Navíc tento způsob umožňuje ve spojení s analýzou 2D snímku téměř plně automatizovanou rekonstrukci obličeje. Modifikační model vychází z databáze 3D scanů reálných obličejů, přičemž nové jsou vytvářeny jako jejich lineární kombinace. To dává k dispozici velké množství kombinací, které se přitom nevymykají přirozenému vzhledu.



**Obrázek 3.1:** Schéma využití modifikačního modelu pro rekonstrukci obličeje z 2D snímku [1].

#### 3.1.1 Vytvoření databáze

K vytvoření databáze je využito několik desítek až stovek 3D scanů hlav dospělých osob obou pohlaví. Všechny osoby mají obličej prostý jakýchkoliv doplňků, vousů, či makeupu. Dalším krokem je vertikální (zadní část hlavy) a horizontální (ramena) ořezání a normalizace, která zajistí, že každý obličej bude mít stejnou orientaci a pozici v prostoru. Výsledné obličeje jsou reprezentovány polygonálním modelem s přibližně 70000 vrcholy a stejným počtem barevných hodnot. Společně s tvarem je tedy uložena i informace o barvě [1].

### 3.1.2 Vytvoření modelu

Modifikační model je založen na sadě trojrozměrných modelů. Aby bylo možné vytvářet jejich lineární kombinace, je třeba, aby spolu tyto modely svou strukturou plně korespondovaly. Geometrie tváře je reprezentována vektorem  $S = (x_1, y_1, z_1, x_2, \dots, x_n, y_n, z_n)^T \in \mathcal{R}^{3n}$ , který popisuje souřadnice  $x, y$  a  $z$  každého z  $n$  vrcholů. Pro jednoduchost je počet barev stejný jako počet vrcholů (údaje o barvě jsou uchovávány pro každý vrchol). Textura obličeje je tedy reprezentována vektorem  $T = (R_1, G_1, B_1, R_2, \dots, R_n, G_n, B_n)^T \in \mathcal{R}^{3n}$ , nesoucím informaci o RGB hodnotě každého vrcholu [1].

Nový model tvaru ( $S_{model}$ ) a textury ( $T_{model}$ ) obličeje dostaneme lineární kombinací vzorů z databáze:

$$S_{model} = \sum_{i=1}^m a_i S_i, \quad T_{model} = \sum_{i=1}^m b_i S_i, \quad (3.1)$$

kde

$$\sum_{i=1}^m a_i = \sum_{i=1}^m b_i = 1. \quad (3.2)$$

Modifikační model je tedy definován jako množina obličejů ( $S_{model}(\vec{a}), T_{model}(\vec{b})$ ) parametrizovaných koeficienty  $\vec{a} = (a_1, a_2, \dots, a_m)^T$  a  $\vec{b} = (b_1, b_2, \dots, b_m)^T$ . Nové obličeje jsou generovány změnou těchto parametrů, které ovlivňují tvar a texturu modelu [1].

Variabilitu modelu lze zvýšit rozdělením obličeje do několika částí (např. oči, nos, ústa a jejich okolí), které budou transformovány nezávisle (obr. 3.2). Tato segmentace je ekvivalentní rozdělení vektorového prostoru obličejů na nezávislé podprostory [1].

### 3.1.3 Využití 3D modifikačního modelu při rekonstrukci

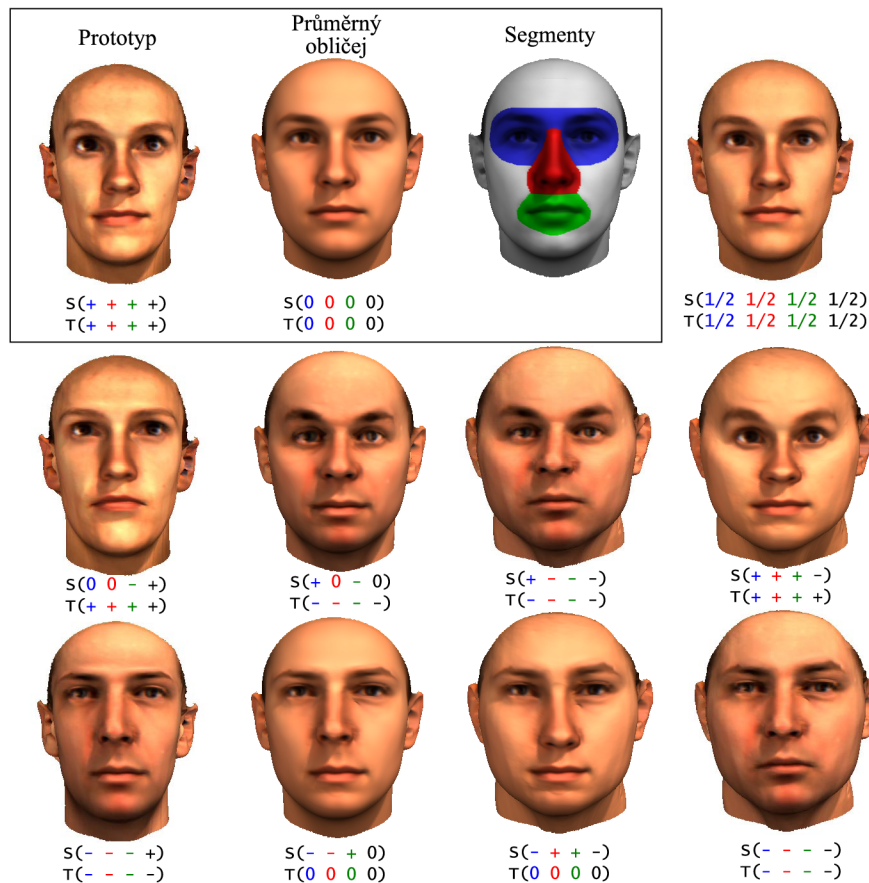
Cílem rekonstrukce je vytvořit změnou koeficientů modelu novou tvář, která bude svým tvarem odpovídat obličejí na vstupním snímku. Současně jsou stanoveny další parametry scény, jako natočení a pozice obličeje, nasvícení scény, či ohnisková vzdálenost kamery. Algoritmus v cyklu hledá takové parametry modelu a scény, aby byl nový obraz, který je renderován grafickým algoritmem, co nejvíce podobný vstupnímu. Iterativní proces začíná s průměrným obličejem a standardními podmínkami pro renderování scény (pohled a nasvícení zepředu) [2].

Pro inicializaci potřebuje systém znát pozici několika výchozích bodů v obličejí, např. koutky očí, úst, špička nosu. Tyto body jsou zadávány ručně tak, že uživatel označí příslušný vrchol modifikačního modelu a následně odpovídající bod na vstupním snímku. Tyto body mohou být tedy pro každý obličej různé. Záleží na tom, které části obličeje jsou na vstupním snímku viditelné [2].

#### Algoritmus pro renderování scény

Výpočet syntetizované scény lze rozdělit na dvě části. První řeší pozici a natočení modelu (vrcholů modelu) a jeho perspektivní projekci. Souřadnice každého vrcholu  $k$ ,  $\mathbf{x}_k = (x_k, y_k, z_k)^T$  jsou transformovány do souřadnic relativních vůči kameře:

$$(w_{x,k}, w_{y,k}, w_{z,k})^T = \mathbf{R}_\gamma \mathbf{R}_\theta \mathbf{R}_\phi \mathbf{x}_k + \mathbf{t}_w, \quad (3.3)$$



**Obrázek 3.2:** Ovlivnění vzhledu obličej v závislosti na koeficientech  $\vec{a}$  a  $\vec{b}$ . (+) a (-) značí zvýšení resp. snížení váhy daného segmentu [1].

kde úhly  $\theta$  a  $\phi$  definují rotaci kolem vertikální a horizontální osy a úhel  $\gamma$  rotaci kolem osy kamery.  $\mathbf{t}_w$  je posunutí v prostoru [2].

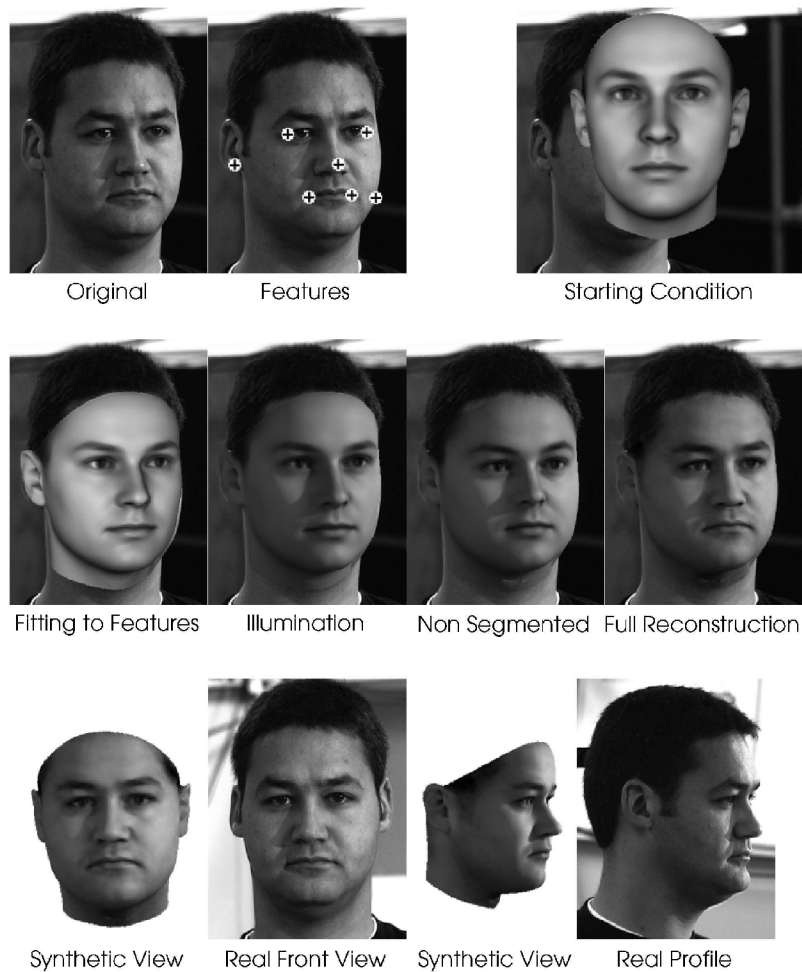
Perspektivní projekce potom transformuje souřadnice vrcholů do 2D prostoru v závislosti na ohniskové vzdálenosti ( $f$ ) [2].

$$p_{x,k} = P_x + f \frac{w_{x,k}}{w_{z,k}}, \quad p_{y,k} = P_y + f \frac{w_{y,k}}{w_{z,k}} \quad (3.4)$$

Druhá část algoritmu řeší barvu a nasvětlení modelu. Stínování modelu závisí na směru normálových vektorů trojúhelníků, které tvoří model, a parametrech světel ve scéně. Výpočet osvětlení je řešen pomocí *Phongova osvětlovacího modelu*. Tento model poskytuje dostatečně přirozené výsledky a je relativně rychlý. Viditelnost vrcholů a stíny jsou řešeny pomocí z-bufferu.

## 3.2 Rekonstrukce založená na vzorech

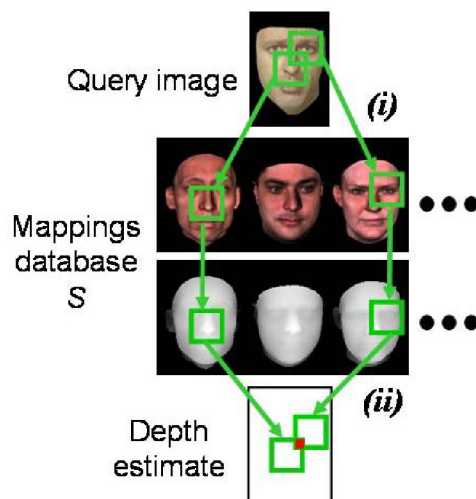
Touto metodou je možné rekonstruovat hloubku snímků nejrůznějších objektů. Vychází z předpokladu, že objekty stejné *třídy* (např. lidské obličej) mají podobný tvar. Je vybrána



**Obrázek 3.3:** Vstupní snímek a umístění záchytných bodů (vlevo nahoře). Výchozí pozice modelu (vpravo nahoře). Zarovnání vzhledem k záchytným bodům, rekonstrukce osvětlení a tvaru (uprostřed). Výsledky rekonstrukce (dole) [2].

sada 3D objektů, které představují *vzory* pro takovou třídu. Tyto objekty jsou použity pro vytvoření databáze  $S = \{M_i\}_{i=1}^n = \{(I_i, D_i)\}_{i=1}^n$ , kde  $I_i$  je snímek a  $D_i$  jeho hloubková mapa. Spolu se snímky jsou tedy uloženy i informace o hloubce. To umožňuje odhadnout tvar objektu ze vstupního snímku jeho porovnáním se záznamy v databázi [7].

Databáze bude ovšem málokdy obsahovat informaci o objektu totožném s tím na vstupním snímku. Je tedy potřeba vytvořit algoritmus, který bude z údajů v databázi schopen vytvářet nové tvary. Ten pracuje tak, že pro každou část  $p$  vstupního obrazu  $I$  vybere oblast velikosti  $k \times k$  v okolí  $p$ . Pro každou takovou oblast jsou v databázi vyhledány k ní nejlépe odpovídající a tím je získána informace o její pravděpodobné hloubce. Sloučením získaných informací vznikne hloubková mapa celého snímku [7].



**Obrázek 3.4:** Ukázka práce hloubkové rekonstrukce. Části vstupního snímku jsou porovnávány s údaji v databázi. Hloubkové mapy vyhovujících záznamů jsou pak použity pro rekonstrukci [7].

### 3.3 Rekonstrukce kombinací statistických a geometrických informací

Jde o algoritmus založený na modelu získaném ze sady 3D modelů obličejů. Na rozdíl od předchozí metody, která je založena na statistickém 3D modelu, využívá geometrických omezení odvozených ze vzdálenosti některých významných bodů v obličejí a Hausdorffovy vzdálenosti mezi obrysy vypočtenými z obrázku obličeje a projekce přibližného modelu hlavy. Tento algoritmus je efektivnější z hlediska výpočtu, jelikož je minimalizována závislost na osvětlení [12].

Podobně jako u modifikačního modelu je nejprve vytvořen ze statisticky získaných informací (3D scany obličejů) univerzální model. Stejně tak je prvním krokem rekonstrukce manuální zadání významných bodů, které umožní odvození parametrů velikosti, rotace a posunutí modelu. Tím je získána výchozí poloha modelu.

Po přibližném umístění modelu jsou automaticky nalezeny další shody, které umožňují lepší zarovnání modelu se snímek. Model je nejprve transformován do cylindrických souřadnic, které umožňují snadnější hledání [12].

Algoritmus pro rekonstrukci se sestává ze dvou fází. Nejprve je odhadnuta přibližná podoba 3D modelu z rysů vstupního obličeje. Následuje zpřesňování tvaru modelu kombinací testování shod obličejových rysů a kontur. Rysy, z kterých algoritmus vychází, jsou např. pozice koutků úst, očí, krajní body obočí, spodní okraj horního a dolní okraj dolního rtu. Informace o pozici obličeje je možné získat minimalizací funkce (3.5) [12].

$$E = \sum_{i=1}^n \|u_i - (sR\bar{x}_i + t)\| \quad (3.5)$$

kde  $u_i$  je  $i$ -tý bod na 2D snímku a  $x_i$  představuje  $i$ -tý bod na hledaném obličejí.  $s, R, t$  jsou vektory pro změnu velikosti, rotaci a posunutí. K získání výchozího modelu ze shod bodů můžeme tento problém vyjádřit jako lineární systém (3.6) [12]:



$$\begin{bmatrix} \bar{x}_1 & v_{1,1x} & \cdots & v_{1,ex} & 1 & 0 \\ \bar{y}_1 & v_{1,1y} & \cdots & v_{1,ey} & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \bar{y}_{fn} & v_{fn,1x} & \cdots & v_{fn,ey} & 0 & 1 \end{bmatrix}_{2fn \times (3+e)} \begin{bmatrix} s \\ \alpha_1 \\ \vdots \\ \alpha_e \\ t_x \\ t_y \end{bmatrix}_{(3+e) \times 1} = R^{-1} \begin{bmatrix} u_{1,x} \\ u_{1,y} \\ u_{2,x} \\ u_{2,y} \\ \vdots \\ u_{fn,x} \\ u_{fn,y} \end{bmatrix}_{(2fn) \times 1}, \quad (3.6)$$

kde  $R$  je konkatenace matice odhadované rotace,  $s$  je změna měřítka,  $v_{i,j}$  označuje  $i$ -tý prvek  $j$ -tého vektoru báze a  $u_i$  je souřadnice  $i$ -tého bodu obličejového rysu [12].

Dalším krokem je rekonstrukce detailů, ta je prováděna pomocí kombinace informací o rysech a konturách obličeje. K detekci kontur je aplikována metoda pro extrakci kontur založená na křivkách. Algoritmus pro extrakci kontur z 3D modelu vypadá následovně [12]:

1. Odhad směru normály každého vrcholu modelu vypočtením normálových vektorů vyhlazeného povrchu 3D obličeje.
2. Vrcholy modelu, jejichž normály jsou kolmé ke směru pohledu, jsou vybrány jako kandidáti pro kontury.
3. Prostor kandidátů je rozdělen do několika skupin a z každé je vybrán jeden s nejvyšší hodnotou  $z$  jako kontrolní bod 3D kontury.
4. Z kontrolních bodů je vytvořena křivka (3D kontura), jejíž body jsou promítnuty na křivku v 2D snímku.

Ke zjištění odchylky mezi konturami z obrázku a konturami promítnutými z 3D modelu je použita upravená Hausdorffova vzdálenost:

$$H(A, B) = K_{a \in A}^{th} \min_{b \in B} \|a - b\|, \quad (3.7)$$

kde  $k$ -tá největší hodnota minimální vzdálenosti od pozice množiny bodů  $A$  k množině bodů  $B$  je použita jako vzdálenost mezi dvěma množinami bodů [12].

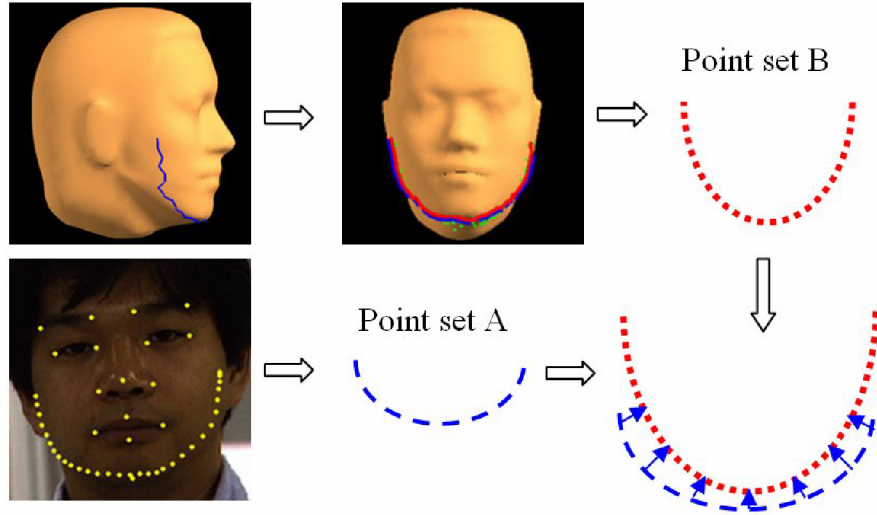
Rekonstrukce je tedy řešena minimalizací funkce

$$E = \sum_{i=1}^n \left\| u_i - \left( sR(\bar{x}_i + \sum_{j=1}^e \alpha_j v_{j,i}) + t \right) \right\| + H(C_1, C_{pj}(M)), \quad (3.8)$$

kde  $C_1$  je množina bodů kontury extrahované z 2D snímku a  $C_{pj}$  je množina bodů kontury získané projekcí 3D modelu  $M$  [12].

### 3.4 Metoda využívající ALLE algoritmus

Jde o vysoce automatizovanou metodu rekonstrukce. Při analýze je použita metoda *Adaptive manifold learning* pro snížení počtu rozměrů prostoru vzorů 3D obličejů. Tato metoda je efektivnější než metoda PCA (2.5).



**Obrázek 3.5:** Částečně jednocestná Hausdorffova vzdálenost: Množina  $A$  obsahuje body kontury na 2D snímku, množina  $B$  body získané projekcí kontury z 3D modelu [12].

Množinu obličejových rysů je možné vyjádřit vektorem [8]:

$$F_p = [X_1, Y_1, X_2, Y_2, \dots, X_m, Y_m]^T \in \mathcal{R}^{2m}, \quad (3.9)$$

kde  $m$  je počet obličejových rysů. Proces rekonstrukce lze rozdělit do tří kroků. V prvním je odhadnuta přibližná poloha modelu pomocí AAM (2.3). Po afinních transformacích je vektor  $F_p$  transformován do prostoru vzorových obličejů. Takto transformovaný vektor označíme  $F'_p$ . Z  $F'_p$  získáme souřadnice  $m$  obličejových rysů všech vzorových obličejů. Tyto rysy představují podprostor původních vzorů [8]:

$$S_f = \begin{bmatrix} X_{1,1} & \dots & X_{N,1} \\ Y_{1,1} & \dots & Y_{N,1} \\ \vdots & \ddots & \vdots \\ X_{1,m} & \dots & X_{N,m} \\ Y_{1,m} & \dots & Y_{N,m} \end{bmatrix} \in \mathcal{R}^{2m \times N}. \quad (3.10)$$

Poté je použit algoritmus pro adaptivní výběr sousedů k vybrání  $k$  vzorků z  $S_f$ , které se nejvíce blíží  $F'_p$ . Tyto vzorky jsou označeny jako  $S_i \in \mathcal{R}^{2m \times k_i}$  [8].

$$[S_i, K_i] = \text{AdaptivelySelectNeighbor}([F'_p, S_f]) \quad (3.11)$$

V dalším kroku je použit LLE algoritmus k výpočtu váhy  $W_{extract}$  (3.12) pro rekonstrukci  $F'_p$  pomocí  $S_i$  (3.13).

$$[W_{extract}, Y_{extract}] = \text{LLE}([F'_p, S_i], k_i, d), \quad (3.12)$$

$$F'_p = S_i \times W_{extract}, \quad (3.13)$$

kde  $k_i$  je velikost oblasti pro hledání souseda v prvním kroku a  $d$  je velikost souřadnic. Dále najdeme souřadnice  $Y'_{origin} \in \mathcal{R}^{d \times k_i}$  odpovídající souřadnicím  $Y_{origin}$ . Souřadnice objektu na vstupním snímku mohou být získány jako lineární kombinace  $Y'_{origin}$ :  $Y_{recon} = Y'_{origin} \times W_{extract}$  [8].

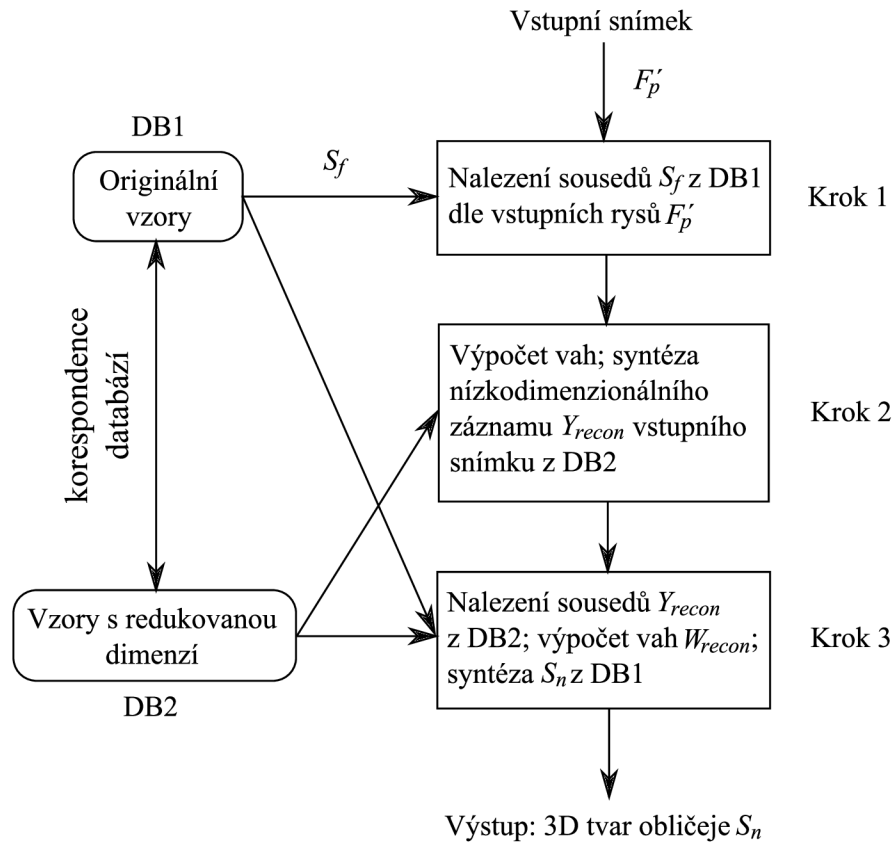
V posledním kroku je  $Y_{recon}$  použito k rekonstrukci tvaru obličeje.

$$[Y_r, K_r] = \text{AdaptivelySelectNeighbor}([Y_{recon}, Y_{origin}]) \quad (3.14)$$

Tyto souřadnice jsou reprezentovány jako  $Y_r = \mathcal{R}^{x \times K_i}$ . Můžeme tedy určit původní vzory  $S_r$ . Nyní můžeme použít LLE algoritmus pro určení váhy  $W_{recon}$ :

$$[W_{recon}, Y'] = \text{LLE}([Y_{recon}, Y_{origin}], K_r, d) \quad (3.15)$$

$S_n = S_r \times W_{recon}$  je pak zrekonstruovaným tvarem vstupního obličeje [8].



**Obrázek 3.6:** Diagram postupu při rekonstrukci [8].

# Kapitola 4

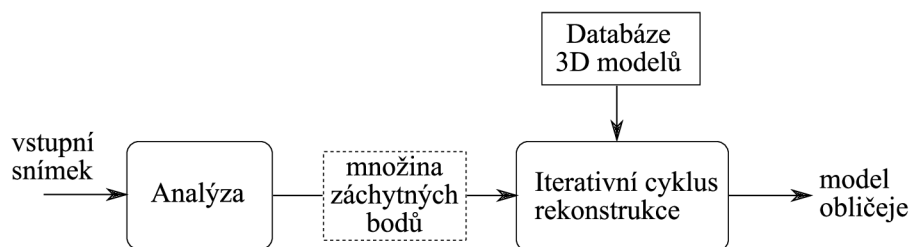
## Implementace

Tato kapitola popisuje návrh aplikace pro automatizovanou rekonstrukci. Přibližuje celkový návrh, způsob, jakým je obličej analyzován, a implementaci vlastního algoritmu pro rekonstrukci. Je zde zmíněno, které technologie byly při implementaci použity.

### 4.1 Návrh řešení

Na základě nastudovaných informací jsem se rozhodl implementovat metodu založenou na rozpoznání obličejových rysů a statisticky získaném 3D modelu. Kvůli složitosti rekonstrukce obličeje jsem se rozhodl problém zjednodušit. Aplikace bude předpokládat snímky obličejů pouze z předního pohledu. Obličeje nebudou ani částečně zakryté a nebudou obsahovat žádné módní doplňky či přehnané líčení. Dále bude tato metoda rekonstruovat pouze tvar obličeje, na rozdíl od některých metod, uvedených v předchozí kapitole. Ty se zabývaly i rekonstrukcí textury obličeje. Hloubka obličeje vzhledem k jeho dalším rozměrům není příliš velká, tudíž je při analýze a rekonstrukci zanedbána perspektiva.

Celý proces rekonstrukce by se měl skládat ze dvou samostatných částí (viz. obrázek 4.1). První je analýza obličeje, při níž je ve vstupním snímku rozpoznán obličej a je detekována množina záchytných bodů. Těmito body jsou například koutky očí a úst, hranice obličeje v úrovni očí, nosu, apod. Takto získané body jsou vstupem druhé části, ve které už probíhá samotná rekonstrukce.



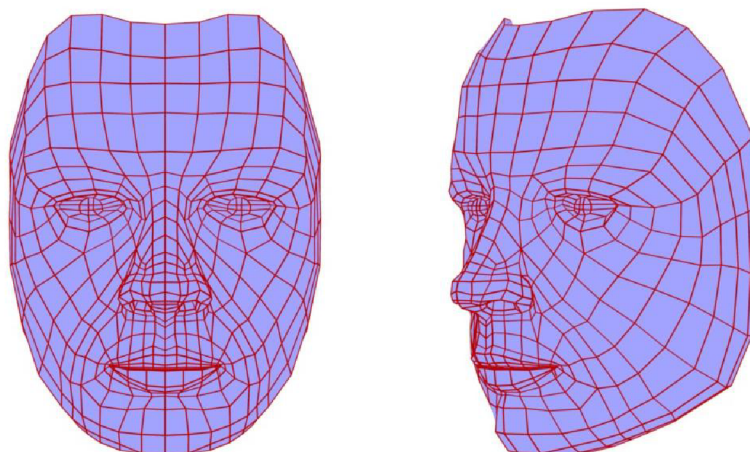
Obrázek 4.1: Schéma celého procesu rekonstrukce

## 4.2 Vytvoření modelu a databáze

Získání databáze 3D obličejů nascanováním hlav různých osob je náročné jak technicky (získat přístup k danému zařízení a dostatečný počet osob), tak časově, a převedení takto získaných scanů na 3D modely také není jednoduché. Volně dostupnou takovou databázi se mi také nepodařilo získat, přistoupil jsem tedy k jejímu ručnímu vytvoření.

### 4.2.1 Model obličeje

Prvním krokem bylo vytvořit 3D model obličeje, ze kterého budou vycházet všechny modely v databázi. Tím bude zajištěno, že všechny modely spolu budou korespondovat. K vytvoření modelu jsem použil program Blender, který je dostupný pod licencí GPL. Při modelování lidského obličeje, resp. hlavy, je vhodné dodržovat určité zásady (např. by síť měla být tvořena pouze čtyřúhelníky) a správnou topologií. Postupoval jsem tedy podle návodu pro vytvoření realistického portrétu [10], přičemž jsem vymodeloval pouze zjednodušenou oblast obličeje s několika úpravami (např. v oblasti očí) tak, aby byl obličej tvořen jedním celistvým objektem.



Obrázek 4.2: Polygonální síť výchozího modelu

Program Blender ukládá informace o scénách v binárním formátu, bylo tedy nutné najít textový formát, do kterého by bylo možné objekty exportovat. Jako vhodný se ukázal formát Wavefront OBJ. Výchozí model obličeje byl vyexportován s údaji o vrcholech, čtyřúhelnících tvořících plochy, ale i normálovými vektory, které jsou potřeba při výpočtu stínování. Tento soubor tedy tvoří šablonu pro všechny záznamy v databázi. U modelů tvořících databázi pak stačí uchovat informaci o pozici vrcholů. Příklad několika řádků ve formátu OBJ ze souboru `normface.obj`:

```
# Blender3D v248 OBJ File: faces.blend
# www.blender3d.org
v 0.000000 -0.249714 0.228850
v -0.022967 -0.235567 0.228660
v -0.027683 -0.264726 0.233041
:
```

```

vn 0.000000 0.023469 0.999695
vn -0.260506 0.040315 0.964599
vn -0.200476 0.241707 0.949400
:
f 1//1 2//2 3//3 4//4
f 4//4 3//3 5//5 6//6
f 3//3 2//2 7//7 8//8
:

```

Znaky od začátku po první bílý znak určují, jaká informace je na řádku uložena. Řádky začínající znakem `v` nesou informace o vrcholech modelu, řádky začínající `vn` obsahují informaci o normálových vektorech a `f` popisují shlukování vrcholů do trojúhelníků, resp. v tomto případě čtyřúhelníků. O dalších možnostech tohoto formátu je možné se dočíst ve specifikaci [3].

Další informací, kterou musí výchozí model obličeje obsahovat, je umístění obličejových rysů. Blender umožňuje shlukovat vrcholy do pojmenovaných skupin (tzv. *vertex groups*). Pozice každého rysu je tedy uložena jako seznam vrcholů, které se na daném místě v modelu nachází. Ukládání informací o těchto skupinách ovšem neumožňuje formát OBJ. Pro Blender je ale možné vytvářet skripty v jazyce Python. Tento problém jsem tedy vyřešil krátkým skriptem, který vytvoří textový soubor, v němž je na každém řádku název skupiny a vrcholy, které ji tvoří.

## 4.2.2 Databáze

Z výchozí polygonální sítě jsem jednoduchými úpravami vytvořil 20 různých obličejů, tvořících databázi pro rekonstrukci. Záznamů v databázi tedy není mnoho, ale k prezentaci metody se ukázal i takový počet dostačující. Z obličejů obsažených v databázi lze vytvářet nové jejich lineárními kombinacemi.

## 4.3 Detekce

Prvním krokem celého procesu rekonstrukce je analýza vstupního snímku. Jak je vidět v kapitole 2, existuje k tomuto problému mnoho přístupů. Mnou implementovaný postup je možné shrnout do tří kroků:

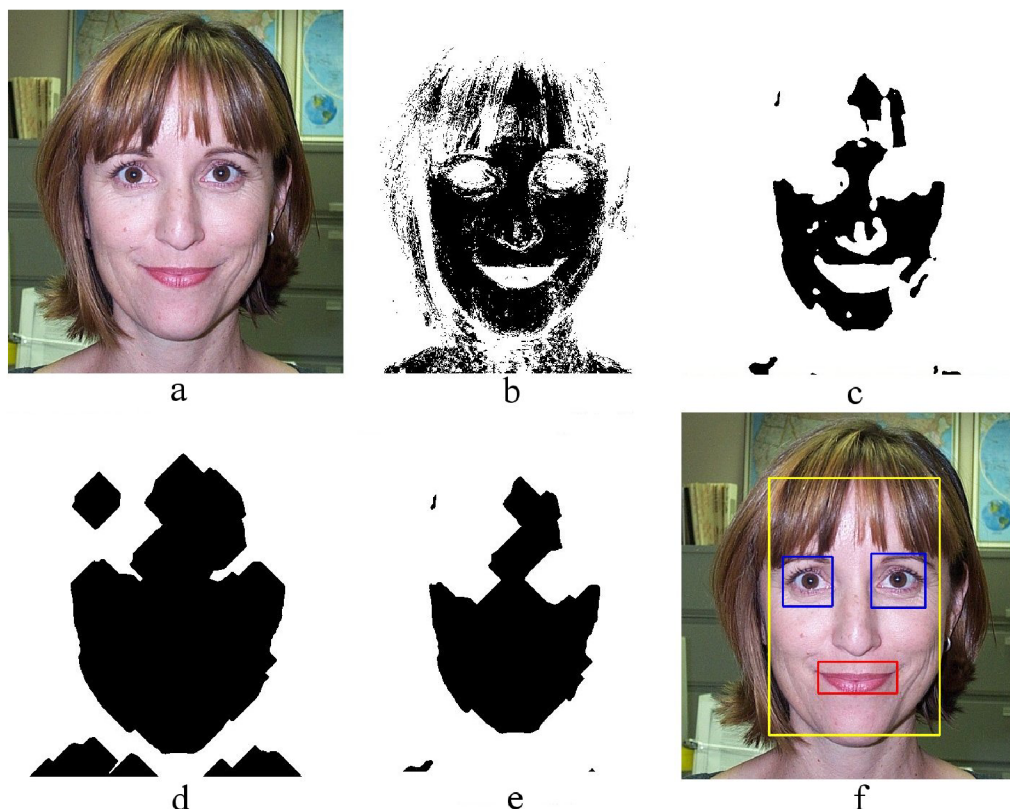
1. Nalezení kandidátských oblastí obličeje, očí a úst
2. Výběr nejvhodnějších kandidátů a validace obličeje
3. Nalezení záchytných bodů pro následující rekonstrukci

### 4.3.1 Hledání kandidátských oblastí

Pro nalezení oblasti obličeje jsem použil metodu uvedenou v části 2.1. Ručním výběrem částí obličeje z několika fotografií a jejich spojením byl vytvořen jediný bitmapový obrázek se vzorky barev kůže. V aplikaci je pak podprostor reprezentován jako třída, ke které se přistupuje jako ke trojrozměrnému poli. Každý rozměr představuje jeden z kanálů RGB a jednotlivé položky pole nesou binární informaci, která udává, zda barva patří do podprostoru barev kůže.

Prvním krokem při hledání oblasti s obličejem je vytvoření binární masky. Každý pixel snímku je testován, zda patří do podprostoru barev kůže. Takto vzniklá maska není úplně přesná – kladně mohou být testovány i pixely mimo obličej, zvláště při velmi komplexním pozadí. Je tedy použito několika filtrů a morfologických operací. Nejprve je maska rozmazána a následně je použito prahování s vhodně zvoleným prahem. Dále jsou použity operace dilatace a eroze. Parametry všech těchto operací byly zvoleny experimentálně na základě testování. Výsledná maska obsahuje shluky, které jsou označeny jako kandidátské oblasti.

K detekci oblastí očí a úst jsem se rozhodl použít knihovnu *OpenCV*, která nabízí detekci oblastí pomocí Haarových filtrů. To umožnilo jednodušší implementaci této části aplikace.



**Obrázek 4.3:** Hledání kandidátských oblastí: Vstupní snímek (a), binární maska příslušnosti do podprostoru barev (b), po aplikaci rozostření a prahování (c), dilatace (d), eroze (e), validní obličej (e).

#### 4.3.2 Validace obličeje

Výsledkem předchozí části jsou seznamy kandidátských oblastí obličejů, očí a úst. Úkolem této části je z nich vybrat ty nejvhodnější (pokud existují).

V jednoduchém cyklu jsou vybírání kandidáti oblastí obličeje a jsou testovány, zda pro ně lze vybrat vhodné oblasti očí a úst. Oblast oka je schválena, pokud se celá nachází v horní části obličeje a v pravé, resp. levé polovině. Podobně pro ústa platí, že musí být v dolní části obličeje a v horizontálním směru přibližně uprostřed. Pokud existuje více kandidátů, kteří těmto podmínkám odpovídají, je vybrán ten, který se blíží nejvíce ideálnímu umístění.

Jsou-li nalezeny oblasti obličeje, pravého a levého oka a úst odpovídající validnímu obličejí, je možné přistoupit k detekci konkrétních bodů. V opačném případě je celý proces ukončen.

### 4.3.3 Hledání záchytných bodů

V aplikaci je implementována detekce následujících bodů:

- Koutky očí
- Krajní body rtů
- Spodní okraj brady
- Hranice obličeje v úrovni úst

Postup při detekci koutků očí a rtů je podobný. Na detekované oblasti je použit Sobelův hranový detektor, prahování a operace dilatace a eroze s experimentálně zvolenými parametry. Tím je získána binární maska jejíž nejkrajnější body v horizontálním směru jsou označeny jako koutky očí, resp. úst.

Detekce hranic obličeje a spodní části brady vychází z detekovaných koutků úst. V případě hranic obličeje je v oblasti mezi okrajovými částmi oblasti s ústy a hranicemi oblasti obličeje v horizontálním směru hledán nejostřejší přechod. Při detekci spodního okraje brady je tento přechod hledán v části obličeje mezi spodními hranicemi oblasti rtů a obličeje.

## 4.4 Rekonstrukce

Po získání množiny záchytných bodů je možné přistoupit k rekonstrukci. Nejdříve je přibližně určena poloha modelu a následně je model v cyklu transformován tak, aby ve výsledku souřadnice záchytných bodů na něm odpovídaly detekovaným bodům.

### 4.4.1 Afinní transformace

Prvním krokem rekonstrukce jsou afinní transformace výchozího modelu obličeje, jejichž cílem je upravit jeho pozici a velikost tak, aby co nejvíce odpovídala obličejí na snímku. Ze souřadnic detekovaných bodů je vypočítán jejich geometrický střed, tím je získán bod, do kterého je třeba model posunout. Velikost modelu je upravena iteračním cyklem, ve kterém je model zmenšován/zvětšován do doby, než je odchylka (výpočet odchylky viz. dále) od předlohy minimální.

### 4.4.2 Výpočet odchylky modelu od předlohy

Odchylka od předlohy je vypočítána následovně. Pro všechny body, které byly ve vstupním snímku detekovány, je zjištěna jejich odchylka od bodů na modelu. Z těchto je vypočten aritmetický průměr a celková odchylka je dána aritmetickým průměrem absolutních hodnot odchylek všech bodů od tohoto průměru.

$$\text{odchylka} = \frac{1}{N} \sum_{i=1}^N |d - d_i| \quad (4.1)$$

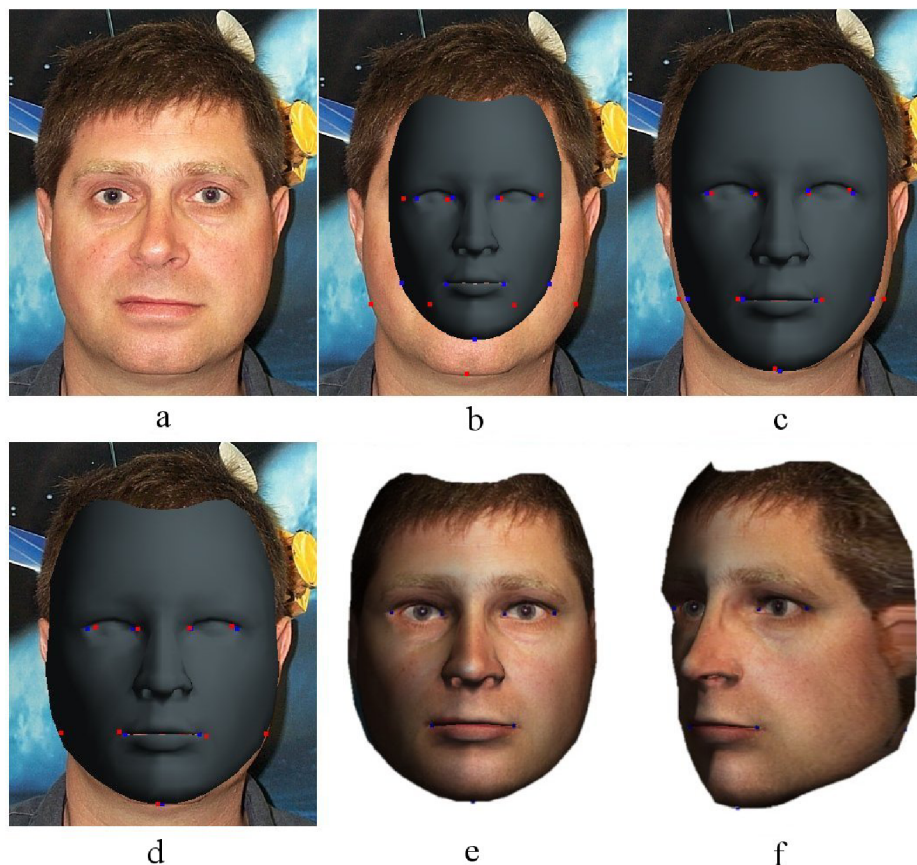


Původně jsem volil pouze výpočet aritmetického průměru. Tento postup je ovšem vhodnější, jelikož je bráno v úvahu, že model se může svým tvarem blížit obličejí na snímku, pouze může být posunutý. A toto posunutí je možné vyřešit aplikací afinní transformace.

#### 4.4.3 Iterační cyklus rekonstrukce

Pokud je model přibližně zarovnán s obličejem na obrázku, je možné přistoupit k samotné rekonstrukci. Aktuální tvar obličeje je vždy dán lineární kombinací obličejů z databáze. Pro každý záznam z databáze je uložen koeficient udávající váhu, jakou má daný obličej při aktuálním tvaru. Součet těchto koeficientů je vždy 1.0. V každém průběhu iteračního cyklu jsou postupně vybírány obličeje z databáze a je měněna jejich váha. Po každé změně je vypočtena odchylka výše uvedeným postupem a pokud je menší než před aplikací změny, je změna modelu a odchylky uložena. V opačném případě je tvar modelu vrácen do původní podoby.

Experimentálně je nastavena výchozí velikost kroku na hodnotu 1.0, tzn. že v prvním cyklu se na výsledném modelu vždy podílí jen jeden obličej a ostatní jsou zanedbány. V dalších krocích už jsou obličeje navzájem kombinovány.



**Obrázek 4.4:** Postup rekonstrukce: Vstupní snímek (a), výchozí poloha modelu (b), model po aplikaci afinních transformací (c), tvar modelu po skončení rekonstrukce (d), výsledný model s namapovanou texturou (e, f).

## 4.5 Ovládání aplikace

Aplikace je spouštěna následovně:

```
facereconstruction [options] <image>
```

kde <image> je vstupní obrázek s obličejem a [options] jsou některé z parametrů:

-h	Vytisknutí nápovědy k programu.
-d	Pouze detekce. Výsledkem je obrázek s detekovanými rysy.
--only-detection	viz. -d
-r	Není prováděna detekce. Uživatel označí body manuálně.
--without-detection	viz. -r
-p	Umožní po skončení rekonstrukce prohlížení modelu
--preview	viz. -p
-o <filename>	Explicitní zadání výstupního souboru

Při použití parametru `-r` (resp. `--without-detection`) je místo automatické detekce použito jednoduché rozhraní, ve kterém uživatel zadá body manuálně. Zadání souřadnic probíhá kliknutím levým tlačítkem myši v okně se vstupním snímkem, pravým tlačítkem myši lze zadání aktuálního bodu přeskočit. Výzvy k zadání bodu jsou zobrazovány v konzolovém okně. Zadávání je ukončeno po označení všech bodů, případně po stisku klávesy **Enter**.

Není-li použit žádný z parametrů `-r` (`--without`), `-p` (`--preview`) chová se aplikace jako čistě konzolová (tohoto je využito u testovacích skriptů).

Aplikace je vytvořena v programovacím jazyce C++ za použití knihoven SDL a OpenCV, které je třeba mít pro správný chod řádně nainstalované.

# Kapitola 5

## Výsledky

V této kapitole je popsáno, jakým způsobem byla aplikace testována. Jsou zde uvedeny chyby, které se při použití zvolených algoritmů projevují, příčiny jejich vzniku a dopady, jaké mají na výsledný model.

### 5.1 Testování

K testování byla použita databáze fotografií obsahující 450 snímků přibližně 27 unikátních osob (databáze byla získána na <http://www.vision.caltech.edu/html-files/archive.html>). Z této databáze bylo vybráno několik desítek snímků odpovídajících požadavkům a omezením uvedeným v návrhu aplikace v kapitole (4). Rovněž byly vyřazeny snímky se zhoršenou kvalitou, jelikož testování mělo ukázat především výsledky implementace algoritmu pro rekonstrukci, nikoliv pro zpracování obrazu.

Na přiloženém médiu jsou k dispozici skripty, které byly použity pro testování. Skript `test_detection.sh` spouští aplikaci pouze pro detekci a výsledky ukládá do adresáře `test_detection`. Výsledkem jsou snímky s vyznačenými detekovanými oblastmi. Tyto informace sloužily ke zjištění, zda byla případná chyba způsobena detekcí a v jaké míře. Druhým skriptem je `test_reconstruction.sh`, jehož výsledkem jsou soubory `.obj` a `.mtl` obsahující informace o výsledných modelech a jejich materiálu.

#### 5.1.1 Chybná detekce

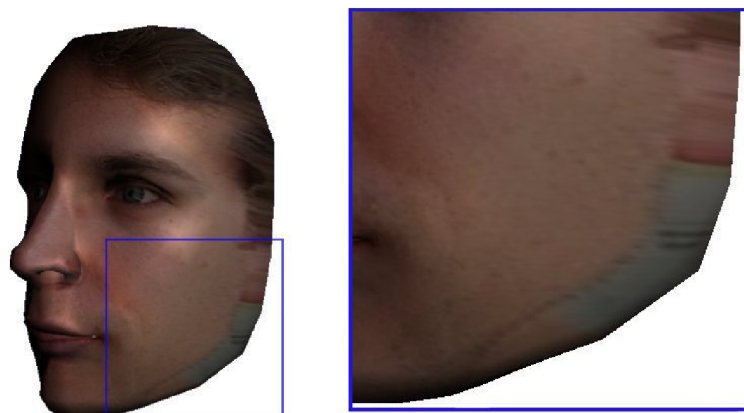
Nejmarkantnější chyby byly způsobeny nepřesnou detekcí. Ukázalo se, že kvalita výstupu je na detekci velmi závislá. Většina chyb vzniklých tímto problémem byla fatálních. Důsledkem nesprávně detekovaného bodu je nepřesná rekonstrukce oblasti v okolí tohoto bodu. Tyto odchylky jsou poté viditelné po namapování textury. Příklad chyby vzniklé nesprávnou detekcí koutku oka je na obr. (5.1).



**Obrázek 5.1:** Příklad chyby v oblasti oka vzniklé chybnou detekcí.

### 5.1.2 Nedostatečné množství detekovaných bodů

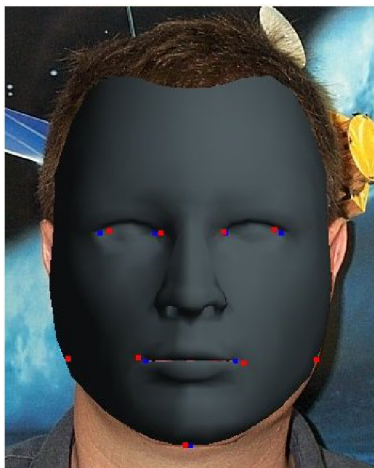
Mimo chyb způsobených detekcí, ovlivňovaly kvalitu výstupu chyby vlivem nedostatečného množství detekovaných bodů. Malý počet bodů, pro které jsou implementovány detekční funkce má za následek, že při rekonstrukci nejsou známy informace o některých rysech. Tyto chyby se projevily zejména v oblastech hranic obličeje (5.2). Ukazuje se, že kvůli omezení počtu bodů, pro které je implementována detekce, je obtížné rekonstruovat detaily obličeje.



**Obrázek 5.2:** Chyba vzniklá nedostatečným množstvím detekovaných bodů v oblasti hranic obličeje.

### 5.1.3 Ovlivnění výsledků počtem záznamů v DB

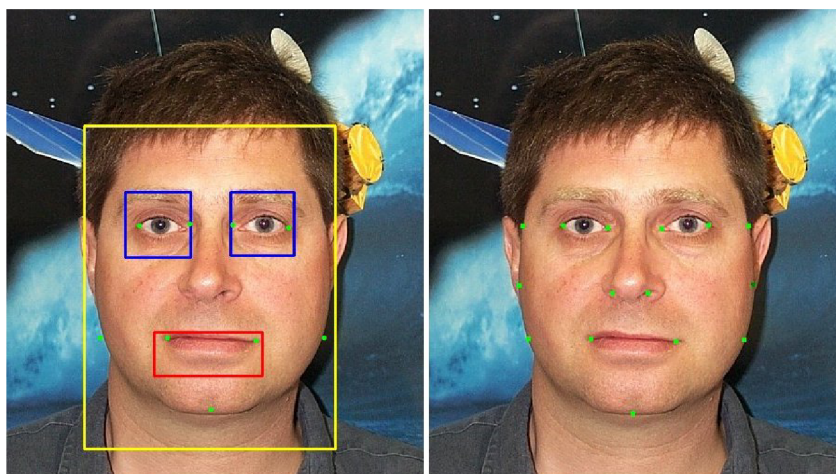
Posledním faktorem ovlivňujícím výsledek bylo malé množství záznamů v databázi obličejů. To se projevuje neúplným zarovnáním bodů na modelu s detekovanými body. Model se tak blíží tvaru obličeje na snímku, ale nemusí mu zcela odpovídat. Tyto chyby ovšem ovlivňovaly výsledek ze všech nejméně. Ukázka ukončené rekonstrukce s nekorrespondujícími body je na obr. (5.3).



**Obrázek 5.3:** Ukázka ukončené rekonstrukce, při které body na modelu (modře) nekorespondují s detekovanými body (červeně) – neotexturovaný model.

## 5.2 Manuální označení záchytných bodů

Aplikaci je možné spouštět v režimu, kdy uživatel zadává body v oblasti obličeje manuálně. Lze takto zadat daleko více bodů, než je aplikace schopna automaticky rozlišit a rekonstrukce je tím přesnější. Rozdíly v chybách mezi takto provedenou rekonstrukcí a rekonstrukcí závislou na správné detekci je možné vidět na obr. (5.5).



**Obrázek 5.4:** Příklad záchytných bodů detekovaných automaticky (ukázka včetně vyznačených oblastí obličeje, očí a úst) a body zadané manuálně (vpravo).



**Obrázek 5.5:** Model obličeje po rekonstrukci s automatickou detekcí (vlevo) a s manuálně zadanými záchytnými body.

U rekonstrukce s manuálně zadanými body je patrná lepší přesnost v umístění očí, která je při automatizované rekonstrukci omezena nepřesnou detekcí koutků očí, i v lepším dodržení okrajů obličeje, jež jsou omezeny počtem bodů, pro které je detekce implementována.

## Kapitola 6

# Závěr

Cílem rekonstrukce je vytvoření objektu, který co nejpřesněji odpovídá reálné předloze obličejů na 2D snímku. Informace, kterou nese dvojrozměrná projekce určitého 3D objektu, není sama o sobě dostatečná k jeho rekonstrukci, je proto nutné získat nějakým způsobem další údaje. Naprostá většina metod využívá databázi statisticky získaných dat s informacemi o tvaru a barvě několika desítek až stovek obličejů. Rozdíl mezi jednotlivými přístupy je pak v analýze vstupního snímku a v implementaci iteračního cyklu samotné rekonstrukce.

Z tohoto jsem vycházel při tvorbě zjednodušené aplikace pro rekonstrukci tvaru obličejů. Analýza spočívá ve vyhledání množiny bodů, které odpovídají obličejovým rysům. Pozice těchto rysů je známá i na statisticky získaném modelu. Proces rekonstrukce tedy spočívá v nalezení takového obličejů, kde budou rysy odpovídat rysům na snímku.

Jako zásadní se ukázala být samotná detekce. Nejmarkantnější chyby způsobovala totiž právě chybná detekce bodů. Další chyby způsobovalo relativně malé množství detekovatelných bodů, čímž vznikaly chyby např. v okrajových částech obličejů. Kvůli omezenému množství záznamů v databázi se často stávalo, že při rekonstrukci nebylo dosaženo úplné korespondence detekovaných bodů s body na modelu. Většinou se ovšem umístění bodu blížilo požadovanému a tyto chyby nebyly velké.

K odstranění posledního zmíněného problému by stačovalo jednoduše přidat další obličejů do databáze. Tím se zvýší pravděpodobnost výskytu obličejů v podprostoru tvořeném lineárními kombinacemi obličejů v databázi. Co se týká chyb způsobených detekcí, zvýšení počtu detekovatelných bodů by teoreticky zvýšilo přesnost, ovšem i složitost implementace (zvláštní funkce pro každý bod). Tento přístup k analýze snímku se tedy ukázal být nepříliš vhodným.

Zlepšení výsledků rekonstrukce by bylo možné využitím některého jiného přístupu k analýze z uvedených v teoretické části práce. Přitom současně implementovaná detekce by sloužila k přibližnému umístění modelu a k porovnávání modelu se vstupem při iteračním cyklu by posloužil vhodnější algoritmus. Samotná rekonstrukce využitím databáze statistických dat se ovšem ukázala být velmi efektivní.

Mimo zlepšení algoritmů pro analýzu by se tedy případné pokračování v této práci mohlo zabývat implementací algoritmu pro *renderování* scény s obličejem, což by lépe posloužilo k porovnávání modelu se vstupním snímek. Výpočet by měl brát v úvahu i osvětlení pod různými úhly a různé ohniskové vzdálenosti kamery. Dále by bylo vhodné rozšířit afinní transformace o rotaci ve všech osách a tím umožnit rozšíření množiny přijímaných snímků o ty, na kterých nejsou obličejů z čelního pohledu. S tím by souviselo i upravení algoritmu pro odhadnutí pozice obličejů.

# Literatura

- [1] Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1999, ISBN 0201485605, s. 187–194.
- [2] Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ročník 25, č. 9, 2003.
- [3] Bourke, P.: *Data Formats*. [online]. [cit. 2009-01-05].  
Dostupné na WWW: <http://local.wasp.uwa.edu.au/~pbourke/dataformats/obj/>
- [4] Cootes, T. F., Edwards, G. J., Taylor, C.: Active Appearance Models. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Springer, 1998, s. 484–498.
- [5] Cootes, T. F., Edwards, G. J., Taylor, C.: Comparing active shape models with active appearance models. In *in Proc. British Machine Vision Conf*, BMVA Press, 1999, s. 173–182.
- [6] Cootes, T. F., Taylor, C. J.: *Imaging Science and Biomedical Engineering*. 2004.
- [7] Hassner, T., Basri, R.: Example Based 3D Reconstruction from Single 2D Images. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, Washington, DC, USA: IEEE Computer Society, 2006, ISBN 0-7695-2646-2, str. 15.
- [8] Jian, Z., Yue-ting, Z.: Sample based 3D face reconstruction from a single frontal image by adaptive locally linear embedding. *Journal of Zhejiang University SCIENCE A*, 2006: s. 550–558, ISSN 1862-1775.
- [9] Částek, P.: *Identifikace Obličejů*. diplomová práce, FIT VUT v Brně, Brno, 2008.
- [10] Terrier-Infographie: *Organic modeling tutorial: Realistic portrait*. [online]. 2009, [cit. 2009-07-04].  
Dostupné na WWW: <http://terrier.infographie.free.fr/index.php?id=396>
- [11] Vlach, J., Přinosil, J.: Lokalizace obličejů v obraze s komplexním pozadím. *Elektrorevue* [online], 2007, ISSN 1213–1539, [cit. 2009-21-04].  
Dostupné na WWW: <http://www.elektrorevue.cz/cz/download/lokalizace-obliceje-v-obraze-s-komplexnim>



- [12] Wang, S. F., Lai, S. H.: Efficient 3D Face Reconstruction from a Single 2D Image by Combining Statistical and Geometrical Information. 2006, s. II:427–436.
- [13] Wikipedia, the free encyclopedia: *Support vector machine*. [online]. Rev. 6. April 2009. [cit. 2009-07-04].  
Dostupné na WWW: [http://en.wikipedia.org/wiki/Support\\_vector\\_machine](http://en.wikipedia.org/wiki/Support_vector_machine)

# Příloha A

## Obsah CD

- Text technické zprávy ve formátu PDF
- Zdrojové soubory technické zprávy
- Zdrojové kódy aplikace
- Soubor ve formátu programu Blender se vzorovými obličejí
- Vzorové obličejí ve formátu OBJ
- Skripty pro testování aplikace
- Databáze snímků obličejů
- Výsledky testovacích skriptů