

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## 3D REKONSTRUKCE OBLIČEJE

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

ZBYNĚK POULÍČEK

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ  
FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## 3D REKONSTRUKCE OBLIČEJE

3D FACE RECONSTRUCTION

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

ZBYNĚK POULÍČEK

VEDOUCÍ PRÁCE  
SUPERVISOR

ING. MICHAL ŠPANĚL

BRNO 2008

## Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2007/2008

### Zadání bakalářské práce

Řešitel: **Pouliček Zbyněk**

Obor: Informační technologie

Téma: **3D rekonstrukce obličeje**

Kategorie: Počítačová grafika

Pokyny:

1. Prostudujte základy zpracování obrazu. Zaměřte se zejména na problematiku 3D rekonstrukce objektů.
2. Analyzujte dostupné materiály na téma 3D rekonstrukce lidského obličeje z obrazových dat a zorientujte se v současných metodách.
3. Vyberte vhodnou metodu a vytvořte data pro její testování.
4. Experimentujte s vaší implementací a případně navrhněte vlastní modifikace metod.
5. Porovnejte dosažené výsledky a diskutujte možnosti budoucího vývoje.
6. Vytvořte stručný plakát prezentující vaši bakalářskou práci, její cíle a výsledky.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění prvních tří bodů zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Španěl Michal, Ing.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2007

Datum odevzdání: 14. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
Fakulta informačních technologií  
Ústav počítačové grafiky a multimédií  
602 00 Brno, Rožehářova 2



doc. Dr. Ing. Pavel Zemčík  
vedoucí ústavu

**LICENČNÍ SMLOUVA  
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO**

uzavřená mezi smluvními stranami

**1. Pan**

Jméno a příjmení: **Zbyněk Pouliček**  
Id studenta: 78786  
Bytem: U Splavu 2868/2, 690 02 Břeclav  
Narozen: 01. 03. 1986, Valtice  
(dále jen "autor")

a

**2. Vysoké učení technické v Brně**

Fakulta informačních technologií  
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305  
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....  
(dále jen "nabyvatel")

**Článek 1  
Specifikace školního díla**

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):  
bakalářská práce

Název VŠKP: 3D rekonstrukce obličeje  
Vedoucí/školitel VŠKP: Španěl Michal, Ing.  
Ústav: Ústav počítačové grafiky a multimédií  
Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:

tištěné formě            počet exemplářů: 1  
elektronické formě    počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

## Článek 2 Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
  - ihned po uzavření této smlouvy
  - 1 rok po uzavření této smlouvy
  - 3 roky po uzavření této smlouvy
  - 5 let po uzavření této smlouvy
  - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

## Článek 3 Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: .....

.....

Nabyvatel

  
.....

Autor

## **Abstrakt**

Tato práce se zabývá 3D rekonstrukcí obličeje s promítnutou pravidelnou mřížkou z 2D snímku. Zahrnuje také detekci mřížky v obraze a popis potřebných metod zpracování rastrového obrazu. Aplikace je napsána v jazyce C++ s využitím grafické knihovny OpenCV a je určena pro OS Windows XP a vyšší.

## **Klíčová slova**

3D rekonstrukce, mřížka, detekce, lokalizace, Houghova transformace, interpolace

## **Abstract**

The top of this thesis is 3D face reconstruction by the help the projected regular grid from the 2D image. It also includes the grid detection and the description of methods that are needed to raster image processing. The application was written in C++ programming language, it uses OpenCV graphic library, and also it is designed for OS Windows XP and higher.

## **Keywords**

3D reconstruction, grid, detection, localization, Hough transformation, interpolation

## **Citace**

Zbyněk Pouliček: 3D rekonstrukce obličeje, Brno, bakalářská práce, FIT VUT v Brně, 2008

# 3D rekonstrukce obličeje

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Michala Španěla.

Další informace mi poskytl Ing. Jaroslav Třešňák.

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

.....  
Zbyněk Poulíček  
12. května 2008

## Poděkování

Rád bych poděkoval vedoucímu bakalářské práce Ing. Michalu Španělovi za odborné vedení a vstřícný přístup, Ing. Jaroslavu Třešňákovi z firmy Oprox Brno a doc. Teodoru Balážovi z Univerzity obrany v Brně za cenné rady a pomoc při získávání podkladů.

© Zbyněk Poulíček, 2008.

*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

Úvod .....	6
1 Techniky zpracování obrazu .....	7
1.1 Konvoluce .....	7
1.2 Hranové detektory .....	8
1.3 Houghova transformace .....	8
1.4 Interpolace bodů .....	10
2 Vymezení základních pojmů .....	11
3 Výroba testovacích dat .....	13
3.1 Způsoby promítání mřížky .....	13
3.2 Výroba projekční mřížky .....	14
4 Návrh algoritmu .....	16
4.1 Pozice rekonstruovaného objektu .....	16
4.2 Zohlednění obličeje jako předmětu rekonstrukce .....	16
4.3 Části algoritmu .....	16
4.4 Detekce promítnuté mřížky .....	18
4.5 Lokalizace uzlových bodů v referenční mřížce .....	19
4.6 Vytvoření sítě bodů .....	21
4.7 Lokalizace uzlových bodů v deformované mřížce .....	21
4.8 Zjištění výšky bodů .....	22
4.9 Interpolace chybějících bodů .....	23
4.10 Sestavení polygonálního modelu .....	24
5 Popis implementace .....	25
6 Výsledky .....	27
6.1 Přesnost algoritmu .....	27
6.2 Výsledky rekonstrukce .....	30
Závěr .....	31
Literatura .....	32
Seznam příloh .....	33
Příloha 1: Manuálová stránka programu .....	34
Příloha 2: Obsah CD .....	35



# Úvod

Tématem bakalářské práce je vytvoření algoritmu pro 3D rekonstrukci obličeje. Vyvinutá metoda by se dala použít například k uchovávání biometrických dat a potažmo i k určování identity osob. V dnešní době se jedná o velmi progresivní oblast s množstvím již existujících postupů a metod, ale jejich nasazení v praxi bývá často náročné a výsledky ne zcela uspokojivé.

Existuje celá řada možností, jak provádět 3D rekonstrukci. Tato práce se zabývá variantou rekonstrukce z jediného obrázku, na němž je předmět s promítnutou pravidelnou mřížkou. Naskýtá se zde velký potenciál pro komerční a veřejné použití. Onu mřížku je teoreticky možné promítat v neviditelné části světelného spektra, a tak může být snímán obličej člověka bez vědomí dotyčného. Metoda je dostatečně obecná, aby se dala použít i pro rekonstrukci jiných typů konvexních objektů. I aplikace této metody v praxi je poměrně nenáročná a neklade ani velké nároky na pozici rekonstruovaného předmětu. (Uvědomme si, že v případě stereometrie, musí být objekt na přesně daném místě.)

Téma práce bylo zadáno firmou Oprox a.s. Ta se zabývá širokým spektrem specializovaných zařízení a projevila zájem o výzkum v této oblasti.

Cílem práce bylo definování metody a vytvoření algoritmu pro 3D rekonstrukci obličeje, jeho atestace a praktické ověření jeho vlastností. První kapitola vymezuje základní pojmy, se kterými se pak dále pracuje, druhá popisuje dostupné techniky zpracování obrazu, třetí se zabývá problematikou tvorby testovacích obrazů. Ve čtvrté kapitole je naznačena funkčnost celého algoritmu a v páté jsou detailněji popsány jednotlivé části programu. V závěru je vše shrnuto, čtenář se seznámí s dosavadními výsledky a je naznačen budoucí vývoj.

# 1 Techniky zpracování obrazu

Se schopností počítačů pracovat s fotografiemi a obecně s dvojrozměrným obrazem vznikla i potřeba definovat v nich obsažené informace. Jedná se o detekci různých znaků a obrazců, trasování, rekonstrukci atd. Využití těchto technik je mimořádně široké a pestré a v dnešní době nabývá na důležitosti. Používají se například v robotice, automobilovém průmyslu, v lékařství pro 3D rekonstrukci lidských orgánů a tkání, dále pro kontrolu identity, rozpoznávání SPZ, uchovávání biometrických dat, detekci obličeje a jeho výrazu a ještě v mnoha dalších oblastech lidské činnosti.

Obecně nejvýznamnější částí 3D rekonstrukce z fotografií, je hledání v nich obsažených užitečných informací. V tomto případě je cílem rozpoznat promítnutou mřížku a nalézt v ní uzlové body. Aby bylo možné postup rekonstrukce správně pochopit, je nejdříve potřeba popsat některé techniky v počítačovém vidění.

## 1.1 Konvoluce

Konvoluce je matematický operátor zpracovávající dvě funkce. Může být spojitá a diskrétní. V počítačové grafice se používá druhá z nich.

$$f(x, y) * h(x, y) = \sum_{i=-k}^k \sum_{j=-k}^k f(x-i, y-j) \cdot h(i, j) \quad (1)$$

V případě diskrétní konvoluce lze jádro chápat jako tabulku (konvoluční maska), kterou položíme na příslušné místo obrazu. Každý pixel překrytý tabulkou vynásobíme koeficientem v příslušné buňce a provedeme součet všech těchto hodnot. Tím dostaneme jeden nový pixel.



**Obrázek 1:** Konvoluce v obrázku [3]

Koeficienty uvnitř konvoluční masky udávají vliv hodnoty pixelu pod nimi. Lze tak nadefinovat velké množství operací např. derivace obrazu (u diskrétního obrazu mluvíme o tzv. odhadu derivace), tedy zvýraznění hran. [3]

Často bývá místo pojmu konvoluční maska používán název konvoluční jádro.

## 1.2 Hranové detektory

Slouží k rozpoznávání hran v obraze. Pracují na principu detekce změny jasové funkce obrazu. Hranu tedy můžeme definovat jako velkou změnu této funkce. Ta bude mít tedy v místě hrany velkou hodnotu první derivace. Nalezení hran v obrázku tak spočívá v hledání těchto vysokých hodnot. Pro jednoduchost výpočtu jsou hrany detekovány ve dvou, resp. čtyřech směrech a výpočet první derivace je aproximován konvolucí. Vznikla tak celá řada ustálených konvolučních jader. Ta nejzákladnější jsou  $(-1, 1)$  a  $(-1, 1)^T$ , případně  $(-1, 0, 1)$  a  $(-1, 0, 1)^T$ . Dále jsou známy operátory Robertsův, Prewittové, Sobelův, Robinsonův, Kirschův...



Obrázek 2: Ukázka hrany

### 1.2.1 Cannyho hranový detektor

Jedním z nejpoužívanějších detektorů je Cannyho. Kombinuje v sobě několik technik za účelem dosažení maximální přesnosti a jednoznačnosti. Nejprve je za účelem eliminace šumu obraz filtrován Gaussovým filtrem, dále se spočítají gradienty pomocí Sobelova operátoru a z nich se určí lokální maxima pomocí metody non-maxima suppression. Na závěr se provede prahování pomocí dvou prahů, přičemž body s odezvou mezi nimi jsou uznány jen v případě, byl-li sousední bod již dříve označen jako hrana. Tato technika se nazývá hystereze.

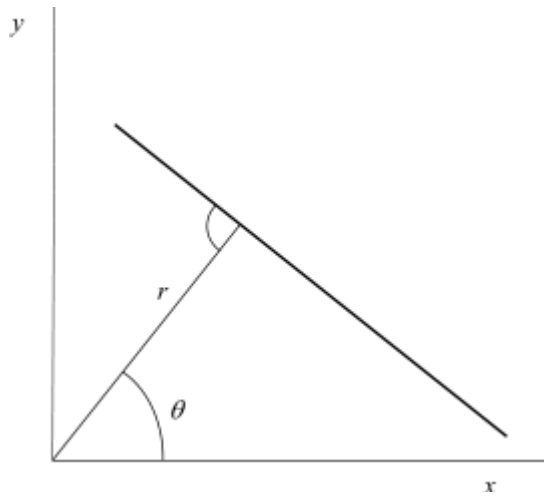
## 1.3 Houghova transformace

Často je potřeba v obraze najít konkrétní geometrické obrazce. K tomu slouží tato metoda. Vyhledává ve fotografii primitivní geometrické objekty, u nichž je znám jejich analytický popis. Její podstatou je transformace z Kartézského souřadnicového systému do polárního a výhodou velká robustnost. Je tedy odolná vůči poruchám obrazu a šumu. Její předností je, že známe analytické rovnice nalezených objektů, čehož se s výhodou využívá v této práci pro nalezení přímk v obraze a jejich průsečíků.

Parametrické vyjádření přímky je:

$$r = x \cdot \cos\theta + y \cdot \sin\theta \quad (2)$$

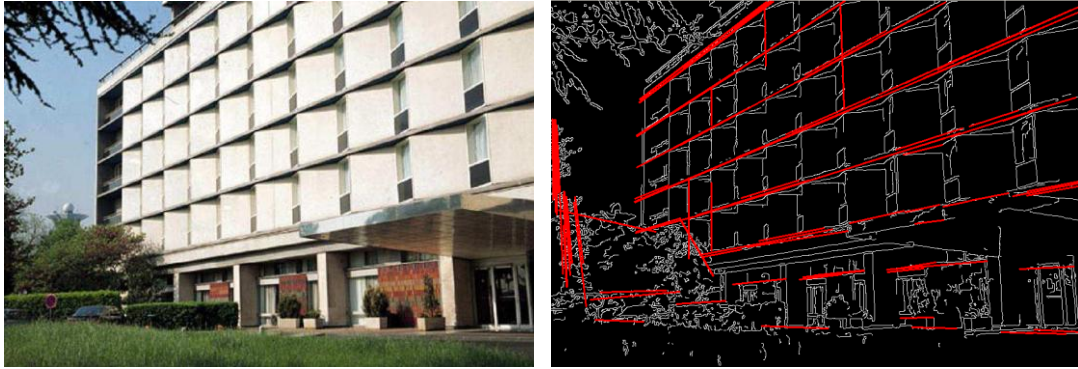
kde  $r$  je délka normály od přímky k počátku souřadnic a  $\theta$  je úhel mezi normálou a osou  $x$  (viz. obr. 6).



**Obrázek 3:** Parametrický popis přímky [4]

Jednotlivé body vstupního obrazu mají parametry  $x$  a  $y$  odpovídající jejich poloze v obraze. Tyto body se dají prezentovat křivkami v parametrickém prostoru. Parametrický prostor je obecně  $n$  rozměrný prostor, kde  $n$  odpovídá počtu hledaných parametrů. V našem případě se jedná o dvourozměrný prostor o parametrech  $r$  a  $\theta$ .

Samotná transformace pak probíhá tak, že parametry  $x$  a  $y$  každého vstupního bodu obrazu, který představuje bod úsečky nebo přímky, jsou dosazeny do rovnice [2]. Za hodnotu  $\theta$  je postupně dosazována každá možná hodnota – interval  $0^\circ, 360^\circ$  (matematicky by se jednalo o nekonečnou množinu hodnot reálných čísel představujících úhel  $\theta$ , ale v případě implementace jsou dosazovány hodnoty podle nastavení citlivosti Houghovy transformace – například po jednom stupni) a hodnota  $r$  se dopočítává. Tak v parametrickém prostoru vznikají křivky, pro každý vstupní bod jedna (v případě úseček se jedná o sinusoidy). Pokud jsou vstupní body kolineární (body ležící na jedné přímce), křivky parametrického prostoru se protínají nejčastěji v bodě představujícím parametry hledané přímky. [4]



Obrázek 4: Houghova transformace pro přímku [4]

## 1.4 Interpolace bodů

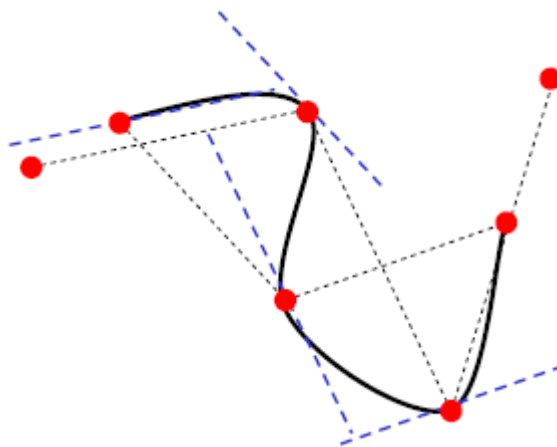
V numerické matematice se jedná o metodu vytváření nových bodů mezi již známými. Ve své podstatě jde o hledání funkce procházející všemi řídicími body. Rozlišujeme množství druhů interpolace: lineární, polynomická, pomocí splajnů, trigonometrická, diskretní Fourierova transformace atd.

Pro tuto práci se ukázala jako nejvhodnější interpolace pomocí Catnull-Rom splajnu.

### 1.4.1 Catmull-Rom splajn

Je určen pro množinu  $N$  bodů  $P_i$  pro  $i=0$  až  $N-1$ . Křivka vychází z bodu  $P_1$  a končí v bodě  $P_{N-2}$ . Minimální počet řídicích bodů je tedy 4. [6]

$$Q(t) = \frac{1}{2} \cdot \mathbf{T} \cdot \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} P_{i-3} \\ P_{i-2} \\ P_{i-1} \\ P_i \end{bmatrix} \quad (3)$$

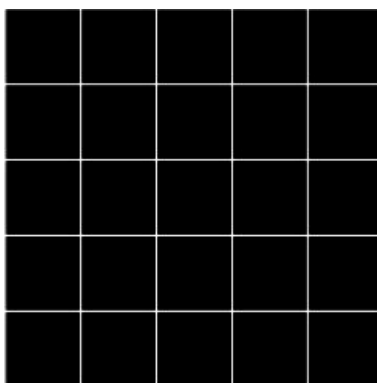


Obrázek 5: Catmull-Rom splajn [6]

## 2 Vymezení základních pojmů

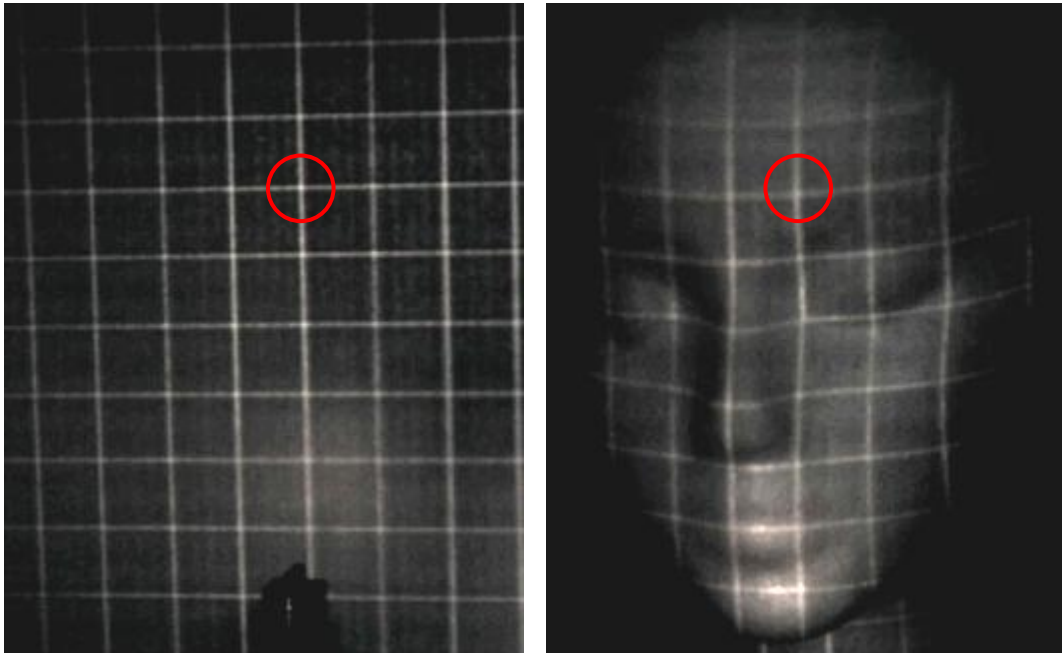
Aby bylo možno práci správně porozumět, je nejdříve třeba vymežit několik pojmů.

- Pravidelná mřížka – čtvercová či obdélníková síť, jež má světlé linky a tmavá okna (obecně tomu tak být nemusí).



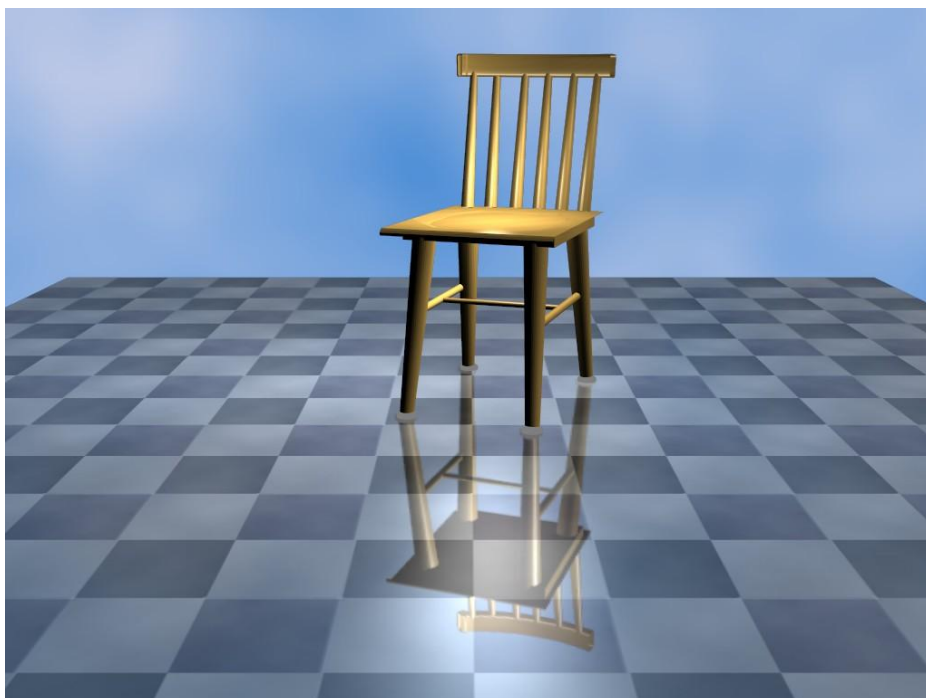
**Obrázek 6:** Pravidelná mřížka

- Uzlové body – průsečíky jednotlivých linií v mřížce.
- Referenční mřížka – pravidelná mřížka promítnutá na ploché stínítko v dané vzdálenosti. Není nijak deformovaná a je využita jako reference při specifikaci tvaru rekonstruovaného objektu. Krok mřížky je pak rozteč mezi jednotlivými liniemi.
- Deformovaná mřížka – pravidelná mřížka promítnutá na objekt, který je předmětem rekonstrukce. Zadní část objektu by se přitom měla nacházet ve vzdálenosti referenční mřížky.
- Korespondující uzlové body – dvojice bodů, kdy jeden je z deformované mřížky a druhý z referenční, přičemž mají v prosvěcované mřížce stejnou pozici. Někdy může být problém určit, které dva body to jsou (obzvláště, když má objekt rekonstrukce hodně zvrásněný tvar nebo je na snímku vidět jen část deformované mřížky).



**Obrázek 7:** Korespondence uzlových bodů

- Polygonální model – 3D model složený z polygonů, tedy 2D mnohoúhelníků (nejčastěji se používají trojúhelníky, jelikož jsou grafickou kartou nejrychleji zpracovatelné).



**Obrázek 8:** Ukázka polygonálního modelu

## 3 Výroba testovacích dat

Naprogramovaný algoritmus je třeba otestovat a k tomu jsou potřeba vstupní data. S jejich získáním autorovi pomáhal Ing. Jaroslav Třešňák z firmy Oprox Brno a pplk. doc. Ing. Teodor Baláž, CSc. z Univerzity obrany v Brně, která poskytla laboratoř a potřebné vybavení.

### 3.1 Způsoby promítání mřížky

Prvním problémem bylo správně stanovit způsob promítání mřížky na objekt. Promítnutá mřížka totiž musí být co nejostřejší a zároveň co nejhušší. Musí být také dostatečně kontrastní. Původně bylo zamýšleno promítat mřížku v infračervené části spektra světla, ale nebyla k dispozici vhodná kamera, která by byla více citlivá na tyto vlnové délky (výsledek byl podobný jako při promítání běžným světlem, jen mřížka byla méně kontrastní).

#### 3.1.1 Digitální projektor

Byl to první zkoušený způsob projekce, bohužel se neosvědčil. Digitální projektory dnes jen zřídka poskytují dostatečné rozlišení pro to, aby byla promítnutá mřížka uspokojivě ostrá a jemná. Dalším problémem pak bylo, že poskytují nestabilní obraz, který při snímání kamerou bliká.

#### 3.1.2 Optická kolejnice

Druhý způsob nanášení mřížky byl pomocí optické kolejnice. Na ni se poskládala příslušná optická soustava. Výhodou tohoto řešení je snadná regulace zobrazení, myslí se tím tedy velikost promítané mřížky, ostrost, apod. Problémem je naopak nalezení ideální sestavy poskytující optimální projekci a především velká propustnost světla kolem mřížky, což zvyšovalo celkovou světelnost scény a snižovalo tak kontrast promítnuté mřížky. Samotné promítané pravidelné mřížky byly vytisknuty na průhlednou folii a také vyřezané do plechového plátku.

Je třeba ještě poznamenat, že vzhledem ke světelným podmínkám byla ve fotografiích vysoká úroveň šumu.





**Obrázek 9:** Snímání pomocí optické kolejnice

### 3.1.3 Diaprojektor

Diaprojektor se ukázal jako nejvhodnější projekční zařízení. Velkou výhodou je ostrá projekce a snadná instalace. Navíc sám o sobě nelimituje jemnost promítané mřížky, ta je dána pouze kvalitou prosvěcovaného snímku.

## 3.2 Výroba projekční mřížky

Po nalezení vyhovujícího promítacího zařízení se ukázalo jako největší problém vytvoření samotné mřížky. K projekci je potřeba negativní typ mřížky, abychom dosáhli požadovaného výsledku. To je mřížka, která propouští světlo jen svými linkami.

### 3.2.1 Možnosti zhotovení

První měření byla prováděna s již vyrobenými mřížkami z plechových plátků, ty byly ale ve výsledku příliš hrubé a neposkytovaly tak dostatečné množství informací o tvaru předmětu.

Dalším řešením bylo použít mřížky vytisknuté na průhlednou folii za pomoci laserové tiskárny. Zde byl již výsledek o poznání lepší, i když poněkud méně kontrastní. Hlavním problémem byla samotná technika tisku. Na promítnuté mřížce bylo vidět zrno toneru a její linie byly rozštěpené.

Dospělo se k variantě nechat vyrobit mřížku profesionály. Dosáhlo se tak nejlepších výsledků, bohužel stále ne ideálních. Výtisky byly částečně zkresleny vlivem antialiasingu a jemné mřížky už příliš splývaly.

Jako ideální řešení se jevilo vyrobit projekční mřížku na diapositiv, to je ale bohužel v dnešní době již téměř nemožné. Nepodařilo se v Brně a okolí nalézt firmu, která by ještě diapositivity vyráběla.

### 3.2.2 Velikost mřížky

Při promítání mřížky je třeba počítat s tím, že s promítací vzdáleností roste i velikost zobrazovaného obrazce. Je tedy třeba zvolit takovou pozici rekonstruovaného předmětu, kde bude rozdíl ve velikostech mřížky na jeho přední a zadní části zanedbatelný. Tuto vzdálenost lze odvodit z parametrů projekčního zařízení podle následujícího vztahu.

$$Z = \frac{-a - f}{f} \quad (4)$$

Kde  $f$  je ohnisková vzdálenost,  $a$  vzdálenost předmětu od čočky a  $Z$  výsledné zvětšení [1]

V případě diaprojektoru se jednalo o zařízení osazené čočkou s ohniskovou vzdáleností  $f = 80\text{mm}$ . Výpočtem bylo zjištěno, že minimální přijatelná vzdálenost předmětu rekonstrukce je  $292\text{cm}$ . V tomto místě je mřížka zvětšena 3,65 krát.

## 4 Návrh algoritmu

Metoda vychází ze skutečnosti, že tvar objektu se promítne do podoby na něj promítnuté mřížky. Je tedy potřeba popsat její deformace. Ty budeme brát jako výchylky od referenční mřížky. Budeme-li uvažovat pravidelnou mřížku, máme k dispozici informace vždy buď o horizontální, nebo vertikální výchylce a v případě uzlových bodů o obojích. Jsou tedy nejpřesnějšími ukazateli tvaru rekonstruovaného objektu a jejich lokalizace je tak základem algoritmu.

Jak už bylo naznačeno, pro rekonstrukci jsou tedy potřeba dvě mřížky – referenční a deformovaná. Parametry první z nich by bylo možné za určitých okolností analyticky spočítat, není to ale příliš univerzální řešení. Mnohem výhodnější je referenční mřížku získat projekcí na ploché stínítko. Aplikace je tedy navržena tak, že nejprve se rozpozná ona referenční mřížka a poté se začne rekonstruovat samotný předmět. Celý algoritmus je koncipován tak, aby se nemusela referenční mřížka detekovat při každé rekonstrukci, ale pouze jednou (pokud bude ovšem scéna neměnná).

### 4.1 Pozice rekonstruovaného objektu

Předmět rekonstrukce by měl být ve scéně umístěn tak, aby od snímacího zařízení jeho nejbližší část byla ve vzdálenosti promítnuté referenční mřížky. To je ideální stav, ale jeho dodržení není podmínkou. Tvar objektu se totiž správně zrekonstruuje, jen budou výšky všech bodů posunuty o jeho vzdálenost vůči referenční mřížce. Stejně je to i s jeho horizontálním a vertikálním posunem.

### 4.2 Zohlednění obličeje jako předmětu rekonstrukce

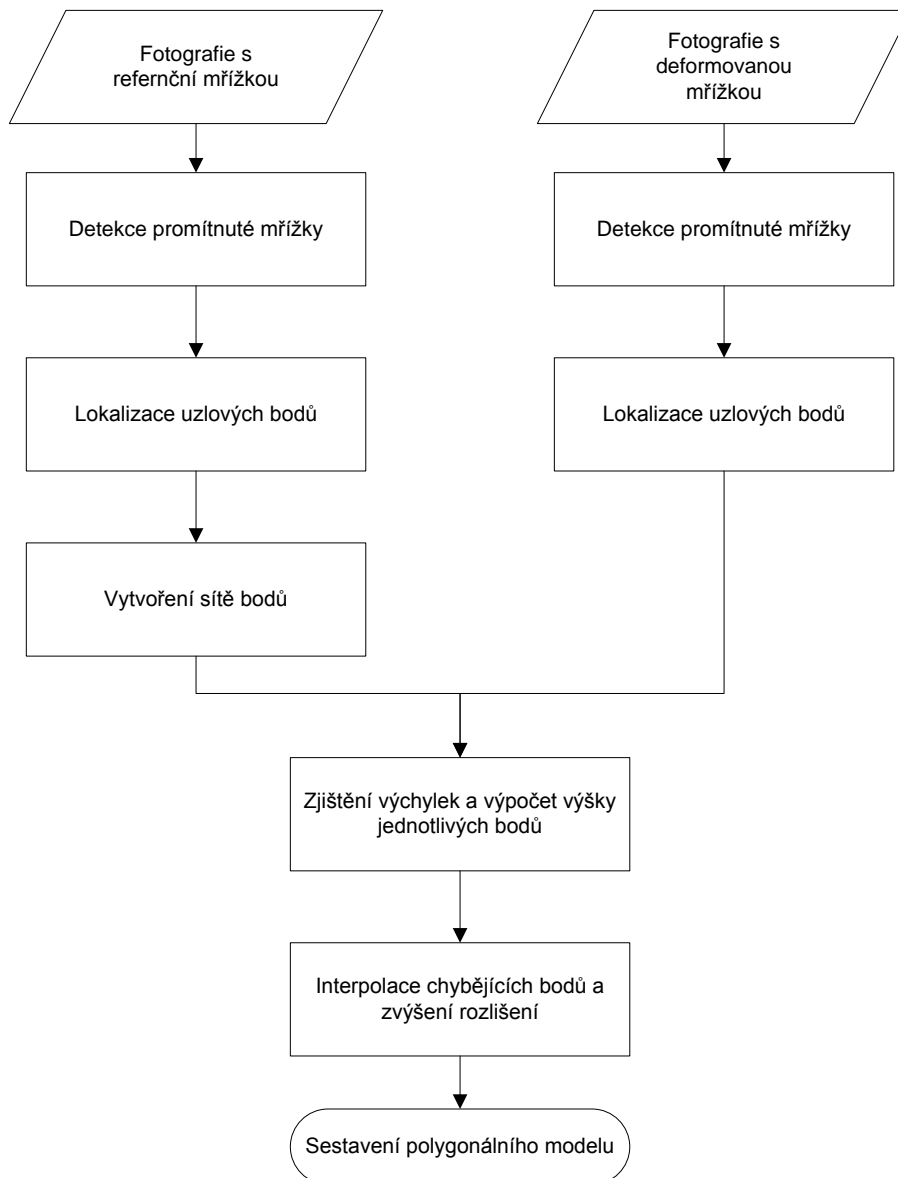
Lidský obličej má tvar a vlastnosti, díky kterým je pro jeho rekonstrukci zvolená metoda mimořádně vhodná. Je konvexní, jeho povrch nemá ostré hrany a mění se jen pozvolna, díky čemuž je promítnutá mřížka dobře identifikovatelná a deformovaná jen lehce. Na druhou stranu je třeba mřížku a její body lokalizovat velmi přesně, neboť i drobná odchylka od skutečnosti může způsobit znehodnocení celého modelu.

Tvaru lidského obličeje by se dal teoreticky přizpůsobit i tvar promítané mřížky, tím se ale tato práce nezabývá, bude to předmětem až dalšího vývoje.

### 4.3 Části algoritmu

Celý program se skládá z těchto částí:

- Detekce promítnuté mřížky
- Lokalizace uzlových bodů referenční mřížky
- Vytvoření sítě bodů
- Lokalizace uzlových bodů deformované mřížky
- Zjištění výchylek a výpočet výšky jednotlivých bodů
- Interpolace chybějících bodů a zvýšení rozlišení
- Sestavení polygonálního modelu a vizualizace



**Obrázek 10:** Vývojový diagram

Jednotlivé části algoritmu budou blíže popsány v následující kapitole „Popis implementace“.

## 4.4 Detekce promítnuté mřížky

U zvolené metody rekonstrukce je nezbytná co nejpřesnější lokalizace mřížky v obraze. Pro člověka je to naprosto triviální záležitost, ne však pro počítač. Nejprve je třeba specifikovat podobu vstupních dat. Ta byla stanovena ve spolupráci s firmou Oprox tak, aby vyhovovala jejich požadavkům (viz. kapitola 4). Bude se tedy pracovat s fotografiemi ve stupních šedi, na kterých je pozitivní mřížka.

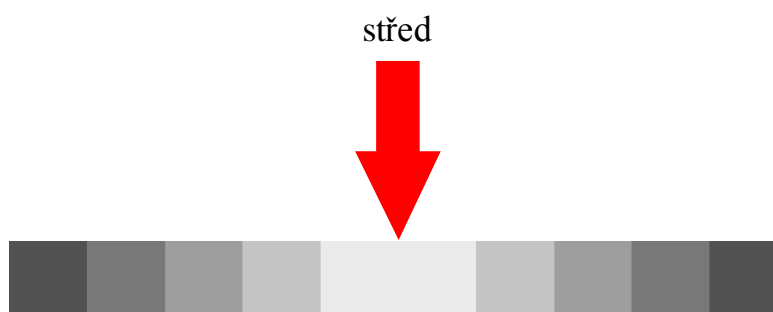
K nalezení mřížky se zde nabízí použití hranových detektorů, ty ale nejsou vhodné. Pro každou linku totiž generují dvě hrany po jejích stranách. Pro rekonstrukci je ale potřeba detekovat pouze jednotlivé linie, potažmo jejich průsečíky. K tomuto účelu byla vyvinuta následující metoda.

Vychází se ze skutečnosti, že jas v jednom řádku resp. sloupci v obraze, de facto představuje dvourozměrný signál. Vrcholy v tomto signálu pak značí linie mřížky. Je třeba ale odlišit užitečnou informaci od šumu. K tomu je použito prahování s prahem rovnajícím se průměrnému jas v obrázku.

### 4.4.1 Určení středu linií

Účelem je získat co nepřesnější a zároveň co nejtenčí reprezentaci mřížky. Vychází se ze skutečnosti, že ona promítnutá mřížka má v obraze největší jas. Hledáme tedy nejjasnější body. Aby byla lokalizace linií co nejpřesnější, jsou jednotlivé linky detekovány po jedné v každém směru. Celkově je tedy třeba projít fotografii hned dvakrát, jednou v horizontálním a podruhé vertikálním směru. Velikost zkoumané oblasti je nastavitelný parametr, zvaný citlivost (angl. sensitivity), dle kterého se řídí i jiné části celého algoritmu. Oblast by měla být rovna minimálně šířce linií mřížky v obrázku. Pokud je citlivost příliš nízká, nepodaří se mřížku správně rozpoznat. S vyšší citlivostí se pak eliminuje šum, ale také roste riziko vzniku děr v detekci. Je tedy potřeba volit kompromis a tento parametr pro každou scénu pečlivě nastavit.

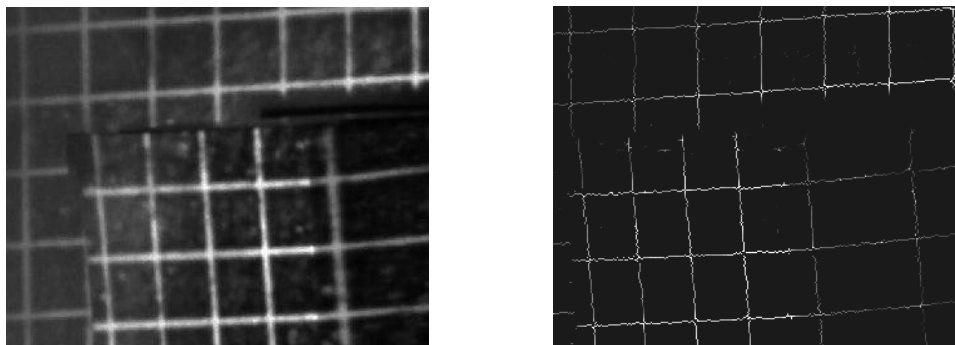
Pro přesné zjištění středu nejjasnějších oblastí je vyžita metoda *non-maxima suppression*. Pokud bude více bodů s nejvyšší hodnotou jasu vedle sebe, vybere se prostřední z nich (v případě sudého počtu se použije zaokrouhlení). Chyba tedy může být teoreticky max. +/- 0,5 pixelu. Z toho je zřejmé, že rozlišení vstupních fotografií zásadně ovlivňuje věrnost celé rekonstrukce.



Obrázek 11: Střed linie

Problém zde činí samotné linie a uzlové body. V těchto místech je totiž koncentrace jasných bodů velmi vysoká a není tak možné určit, zda se jedná o horizontální či vertikální linii, případně, kde má střed. Tento stav je ošetřen pomocí *hystereze*. Zda je dotyčný bod středem některé z linií mřížky či nikoliv, je určeno na základě hodnoty předchozího bodu.

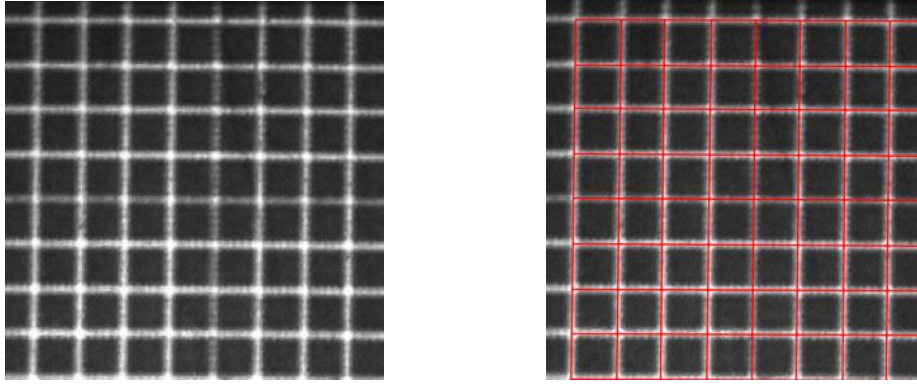
Výsledkem celé detekce je tedy monochromatický obrázek o rozměrech původní fotografie, ve kterém je zobrazena pouze požadovaná mřížka a to o tloušťce jeden pixel.



**Obrázek 12:** Výsledek detektoru

## 4.5 Lokalizace uzlových bodů v referenční mřížce

Pro kvalitní rekonstrukci je naprosto nezbytné správně identifikovat uzlové body v referenční mřížce. Využívá se zde obrázku s rozpoznanou mřížkou a Houghovy transformace pro přímky. Ta umožňuje přesně lokalizovat linie mřížky a je velmi odolná vůči šumu a vadám obrazu. Bohužel nebylo možné využít její implementaci v knihovně OpenCV, poněvadž je zde výsledná množina přímek vytvářena na základě prahování a váhy jednotlivých přímek nejsou známy. Výsledkem tedy může být pro každou linii v mřížce několik potencionálních přímek nebo naopak žádná. Pro lokalizaci uzlových bodů je ale potřeba získat pro každou linku vždy právě jednu přímku, což bude opět vyžadovat dva průchody obrázkem. Za tímto účelem byl tedy sestaven algoritmus, který opět na základě prahování vybírá pro každou linii v mřížce množinu potencionálních přímek, u nichž jsou však známy jejich váhy a mohou být tak vybrány jen ty nejvíce odpovídající. Navíc také zjistí počet linek v horizontálním a vertikálním směru.



**Obrázek 13:** Houghova transformace pro přímky

### 4.5.1 Získávání vah potencionálních linií

Základem této metody je algoritmus pro rasterizaci úseček. Ty jsou vytvářeny na základě znalosti polohy počátečního a koncového bodu. Nejsou-li tyto řídicí body v horizontální či svislé rovině, je výsledná úsečka rozdělena do rovnoměrných bloků tak, aby oba body spojila, jak je vidět na obrázku. Výsledná váha je pak součtem hodnot pixelů v obrázku pod teoreticky vykreslenou úsečkou.



**Obrázek 14:** Rasterizovaná úsečka

### 4.5.2 Stanovení prahu

Aby se nejpravděpodobnější úsečka nevybírala ze zbytečně velkého množství kandidátů, je nutno použít prahování. Samotný práh zde lze poměrně dobře stanovit. Vzhledem k tomu, že je mřížka rozpoznána s přesností na pixely a chyba polohy detekovaných bodů max. +/- 0,5 pixelu, můžeme předpokládat, že u různě naklonených mřížek budou nejpravděpodobnější úsečky protínat alespoň třetinu rozpoznávaných bodů dané linky. Prahová hodnota je tedy 1/3 šířky resp. výšky obrázku s rozpoznanou mřížkou.

### 4.5.3 Zmenšování zkoumané oblasti

Jelikož je předpokládána pravidelná mřížka, nabízí se zde dobrá možnost optimalizovat výpočet. Po detekci první přímky už budeme totiž znát směrnice všech zbývajících. Vyhledávání dalších úseček se bude tedy odvíjet od jejich předpokládaného směrového vektoru a nejpravděpodobnější úsečka se tak bude moci hledat v mnohem menší výseči.

#### 4.5.4 Výpočet polohy uzlových bodů

Popisovaná metoda má obrovskou výhodu v tom, že jsou zjištěné úsečky matematicky definované (v tomto případě obecnou rovnicí přímky). Pro získání uzlových bodů referenční mřížky je tedy potřeba nalézt průsečíky všech úseček. K tomu využijeme rovnice z analytické geometrie.

### 4.6 Vytvoření sítě bodů

Vytvořit síť bodů je potřeba k pozdějšímu sestavení polygonálního modelu. Síť je pravoúhlá a uspořádaná. K její výrobě jsou v první řadě potřeba informace o rozměrech. Ty nám poskytne algoritmus pro lokalizaci bodů v referenční mřížce. Zjištěné rozměry však ještě závisí na rozlišení, což je další uživatelsky nastavitelný parametr. Přepočítávají se podle níže uvedených vzorců a do výsledné sítě se doplní nalezené uzlové body.

$$\begin{aligned}h &= rh_0 - r + 1 \\w &= rw_0 - r + 1\end{aligned}\tag{5}$$

Kde  $h$  je výška sítě,  $h_0$  počet horizontálních linií,  $w$  šířka sítě,  $w_0$  počet vertikálních linií a  $r$  je rozlišení

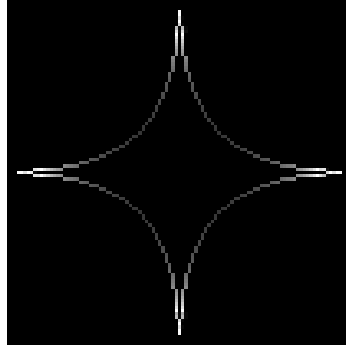
### 4.7 Lokalizace uzlových bodů v deformované mřížce

Deformovaná mřížka vyžaduje pro hledání uzlových bodů zcela jiný přístup. Zde již totiž není mřížka tvořena úsečkami nýbrž obecnými křivkami, jejichž analytický popis pochopitelně není znám. Navíc se jednotlivé linie mohou sbíhat v uzlových bodech z různých stran. Řešením je použití konvoluce.

#### 4.7.1 Konvoluční jádro

Aby byla metoda účinná, je potřeba mimo jiné použít správné konvoluční jádro. To bylo zvoleno ve tvaru duté čtyřcípé hvězdy a jeho vhodnost byla experimentálně ověřena. Hvězda je dutá z toho důvodu, aby se eliminovalo označování šumu jako validní uzlový bod.

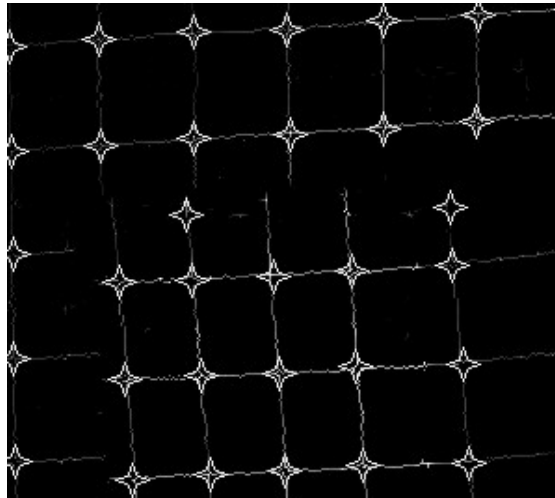




**Obrázek 15:** Konvoluční jádro

### 4.7.2 Detekce uzlového bodu

O tom, zda je bod hledaným uzlem či nikoliv, zde nerozhoduje výsledná hodnota po provedení konvoluce, ale odezva na jednotlivých cípech hvězdy. Aby se jednalo o potenciální uzlový bod, musí být nenulová reakce alespoň na dvou sousedních cípech. (Pozn. v ideálním případě by měla být odezva na všech čtyřech.) V případě, že je této podmínce vyhověno, je výsledkem algoritmu celková odezva konvolučního jádra. Prahováním této hodnoty se pak odfiltrují body, jež mají malou pravděpodobnost, že jsou hledaným uzlem.

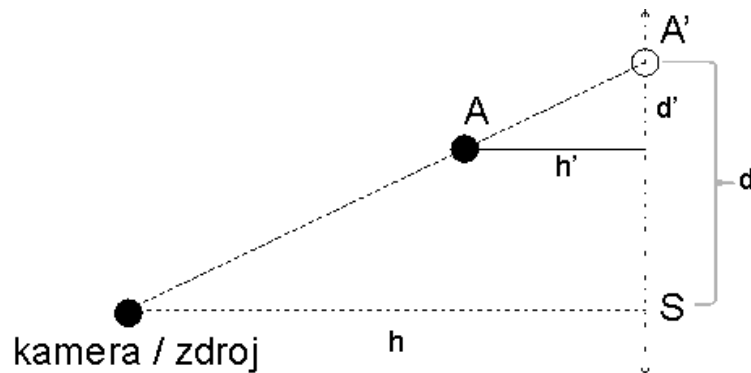


**Obrázek 16:** Detekované uzlové body

## 4.8 Zjištění výšky bodů

Tato část rekonstrukce je stěžejní. Vychází se zde z předpokladu, že tvar rekonstruovaného objektu lze odvodit z míry deformací na něj promítnuté mřížky a ty zjistíme porovnáním deformované mřížky s referenční. Ze znalostí výchylek bodů v deformované mřížce a parametrů scény je pak možné spočítat jejich reálnou výšku. (Výškou je zde míněna vzdálenost od stínítka s referenční mřížkou.)

Výchytky jednotlivých bodů se určí hledáním korespondujících bodů. Ty se zjistí tak, že ke každému uzlu deformované mřížky je postupně přiřazen nejbližší referenční uzlový bod. Odečtením jejich pozice se pak získá výchytky v horizontálním a vertikálním směru. Výpočet výšky se provede na základě trojúhelníkové podobnosti, přičemž kamera a zdroj se zde aproximuje jako jeden bod (což je možné, pokud jsou oba dostatečně blízko sebe a rekonstruovaný objekt dostatečně daleko).



Obrázek 17: Výpočet výšky bodu

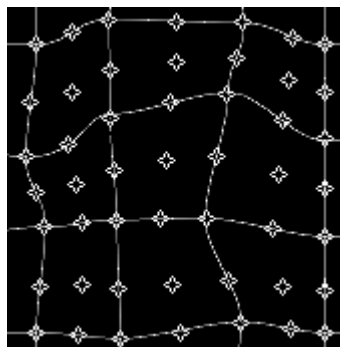
$$h' = \frac{d'}{d} h \quad (6)$$

Tento způsob hledání korespondujících bodů je účinný v případě, kdy použitá mřížka není příliš jemná a rekonstruovaný předmět nemá ostré změny tvaru (což platí pro lidský obličej). Ošetření metody i pro ostatní případy bude předmětem dalšího zkoumání a vývoje.

## 4.9 Interpolace chybějících bodů

Interpolace je potřeba ze dvou důvodů. Za prvé se může stát, že nebude rozpoznán některý z uzlových bodů v deformovaném obrázku, a tak je potřeba dopočítat výchytky resp. výšku v daném uzlu, a druhým důvodem je potřeba doplnit chybějící referenční body, pokud je zvýšeno rozlišení výsledného polygonového modelu. Aby vypočtené body co nejvíce odpovídaly skutečnosti, je jejich pozice vypočtena pomocí křivek.

K výpočtu nových referenčních bodů a výchytek je použit v zásadě stejný postup. Pro každý řádek a sloupec v síti bodů se naleznou řídicí body, mezi nimiž jsou dopočítávány požadované informace.



Obrázek 18: Interpolace bodů

### 4.9.1 Korekce interpolovaných výchylek

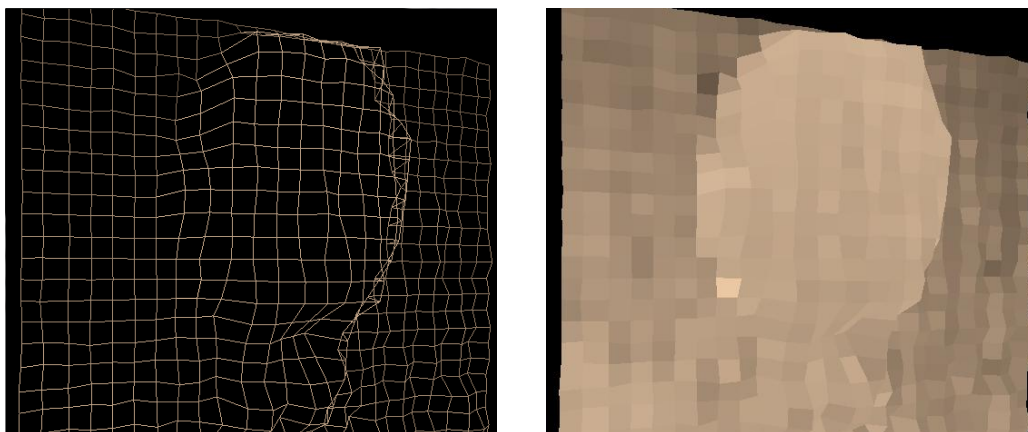
Dopočítané výchylky se ještě zpřesňují pomyslným přichycením daných bodů k liniím deformované mřížky. Tím se využije maximum informací z fotografie. Princip přichycování spočívá v hledání rozpoznané mřížky v okolí bodu. Velikost okolí je přitom dána polovinou průměrné vzdálenosti bodů v síti v pixelech.



Obrázek 19: Body přichycené k mřížce

## 4.10 Sestavení polygonálního modelu

Po aplikování předchozích algoritmů jsou již k dispozici veškerá potřebná data pro vizualizaci polygonálního modelu.



Obrázek 20: Polygonální model hlavy bez textury

## 5 Popis implementace

V této kapitole je podrobněji rozepsána funkčnost celé aplikace. Je velmi pravděpodobné, že v případě nasazení této rekonstrukční metody v praxi, bude zřejmě implementována pro vestavěný systém. Ty mají ovšem omezené výpočetní prostředky, kód algoritmu tedy musí být přenositelný a co neoptimálnější. Je naprogramován neobjektově v jazyce C++ s využitím knihovny *OpenCV*. Ta je využívána především pro nahrávání fotografií a zobrazování doprovodných oken. K samotné vizualizaci polygonálního modelu je použito *OpenGL* s nadstavbou *GLUT*.

Jelikož by měl být algoritmus použit v reálném prostředí na veřejných místech, musí být co nejvíce odolný vůči různým vlivům, jako je šum v získaných fotografiích, umístění objektu ve scéně apod. Je třeba si také uvědomit, že mřížka viditelná na rekonstruovaném objektu bude jen výřezem skutečně promítané mřížky.

### OpenCV

*OpenCV (Open Computer Vision library)* open-source grafická knihovna pro analýzu a zpracování obrazu.

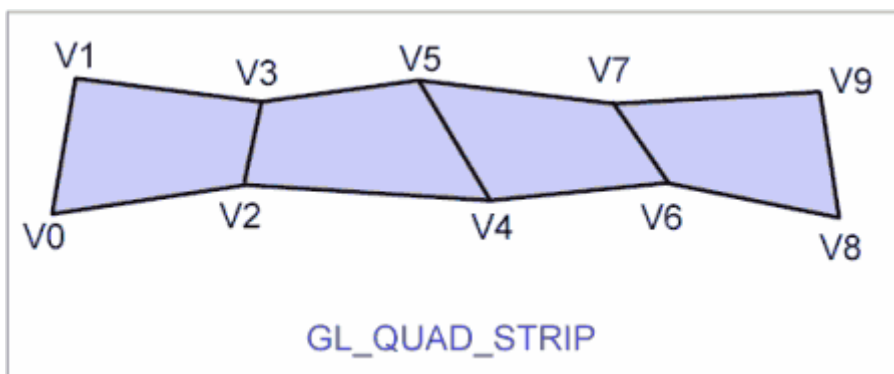
Cílem *OpenCV* je podpořit komerční využití počítačového vidění pro interakci člověk-stroj, robotiku, monitorovací a bezpečnostní systémy. *OpenCV* je zdarma a nabízí otevřenou infrastrukturu zpracování obrazu pro širokou veřejnost. Původně byla vytvořena firmou Intel s motivací využití vysokých výkonů nových procesorů. [5]

V této práci je využita pro nahrávání snímků, počítání jejich průměrného jasu a zobrazení průběhu algoritmu.

### OpenGL

*OpenGL (Open Graphics Library)* je průmyslový standard specifikující multiplatformní rozhraní (API) pro tvorbu aplikací počítačové grafiky. Používá se při tvorbě počítačových her, CAD programů, aplikací virtuální reality či vědeckotechnické vizualizace apod. [7]

Pro sestavení a vizualizaci polygonálního modelu se využívá geometrických primitiv. Těch je v *OpenGL* několik typů, v této práci se však využívá vykreslování pomocí `GL_QUAD_STRIP`.



**Obrázek 21:** GL\_QUAD\_STRIP [6]

## GLUT

*GLUT (OpenGL Utility Toolkit)* tvoří doplněk ke grafické knihovně *OpenGL*. Základem této nadstavbové knihovny je podpora pro práci s okny (včetně zpracování událostí), vyskakovacími menu a písmem. Tyto činnosti totiž nejsou v knihovně *OpenGL* přímo podporovány - důvodem je snaha o co největší zachování platformové nezávislosti. Funkce pro práci s okny či menu, které jsou systémově závislé, se dříve, tj. v době, kdy knihovna *GLUT* neexistovala, musely naprogramovat pro každý operační systém (resp. jeho grafickou nadstavbu) zvlášť, což od vývojáře aplikace vyžadovalo podrobnou znalost funkcí daného operačního systému, grafické nadstavby a správce oken. [8]

# 6 Výsledky

## 6.1 Přesnost algoritmu

Přesnost rekonstrukce vždy závisí na kvalitě informací obsažených ve vstupních rastrových snímcích. Aby se tyto informace daly zpracovávat, je nejdříve třeba je definovat a zde může dojít k odchylkám a nepřesnostem. Jedná se tedy především o části algoritmu pracující přímo se vstupními fotografiemi, což je část detekce mřížky v obrázku a potažmo lokalizace referenční mřížky a uzlových bodů. Na těchto komponentách pak závisí přesnost celé rekonstrukce. V této kapitole bude uvedena přesnost algoritmu na příkladu jednoho ze snímků referenční mřížky pořízených v laboratoři.

### 6.1.1 Odchylka

Odchylkou se zde myslí vzdálenost zjištěného bodu od jeho skutečné polohy.

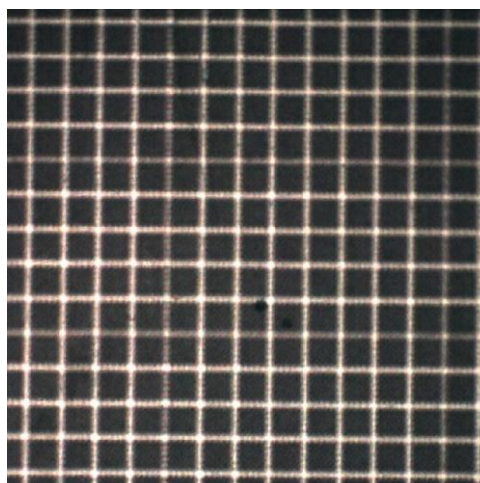
### 6.1.2 Relativní chyba

Relativní chyba je poměr výchylky a kroku mřížky vyjádřený v procentech.

$$e = 100 \cdot \frac{d}{k} \quad (7)$$

Kde  $e$  je výsledná chyba,  $d$  uvažovaná odchylka a  $k$  je krok mřížky.

### 6.1.3 Detekce mřížky



**Obrázek 22:** Referenční mřížka

Samotná mřížka je detekována s přesností na jeden pixel, což samo o sobě může znamenat jisté odchylky. Chyba je zde nepřímo úměrná rozlišení vstupních obrázků a jemnosti a „tenkosti“ mřížky. U uvažovaného snímku je tak maximální odchylka  $\pm 0,38\text{mm}$  a relativní chyba vzhledem k hustotě mřížky 1,92%.

### 6.1.4 Lokalizace uzlových bodů v referenční mřížce

Ty, jak už bylo zmíněno, jsou lokalizovány pomocí Houghovy transformace z obrázku, kde je pouze detekovaná mřížka a jejich přesná pozice je určena analyticky. Tato metoda je velice přesná a dokonce do jisté míry kompenzuje chybu detekce mřížky. Vytváří tedy ideální referenci a lze konstatovat, že v této fázi algoritmu nedochází ke zkreslení rekonstrukce.

### 6.1.5 Lokalizace uzlových bodů v deformované mřížce

Jedná se o část rekonstrukce, která se momentálně na zkreslení výsledku podílí největší měrou. Její přesnost závisí především na kvalitě detekce deformované mřížky. Lokalizace uzlových bodů je nejvíce zkreslena v místech, kde není detekovaná mřížka celistvá. Celková odchylka se zde sčítá s chybou detekce mřížky. Pro atestaci algoritmu je jako deformovaná mřížka použit obrázek č. 22 a porovnán s referenční mřížkou. Průměrná relativní chyba pak činí 2,34%.

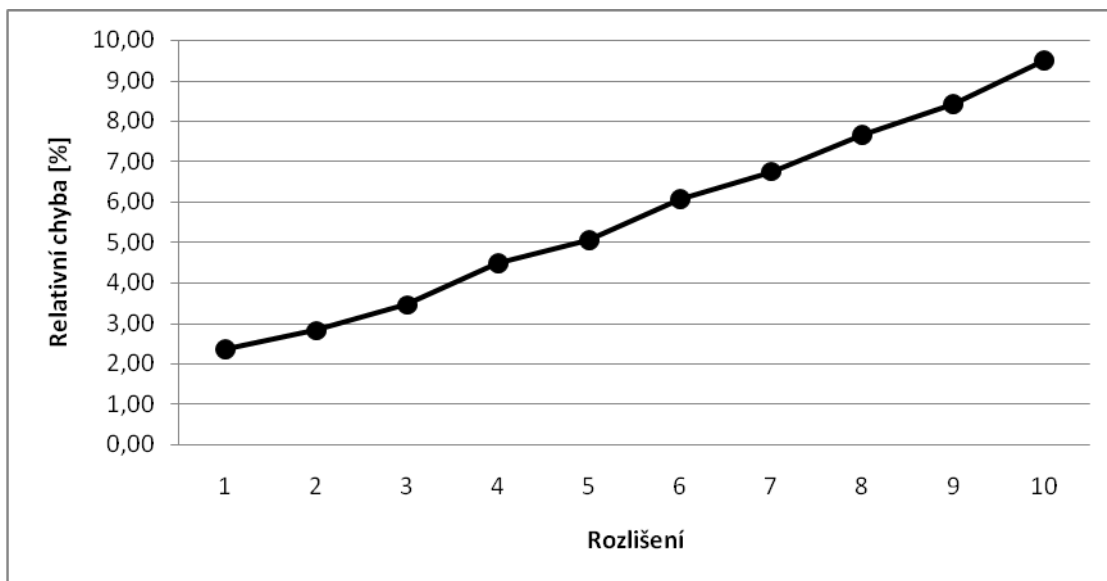
### 6.1.6 Vliv změny rozlišení na přesnost detekce bodů

Zvolené rozlišení je násobek počtu oken referenční mřížky, neboli určuje množství interpolovaných bodů. Vliv interpolací na přesnost rekonstrukce by mohla být v podstatě náhodná, nebýt korekcí vzhledem k detekované mřížce. I dopočítávané body tak nesou alespoň část informace o rekonstruovaném objektu. Jejich přínos je vidět na následujících grafech.

Rozlišení	Odchylka [px]	Odchylka po korekci [px]	Relativní chyba po korekci [%]
1	1,214	1,214	2,34
2	0,763	0,728	2,80
3	0,655	0,595	3,44
4	0,638	0,582	4,49
5	0,588	0,524	5,05
6	0,600	0,524	6,06
7	0,579	0,500	6,74
8	0,579	0,496	7,65
9	0,564	0,485	8,42
10	0,571	0,493	9,51

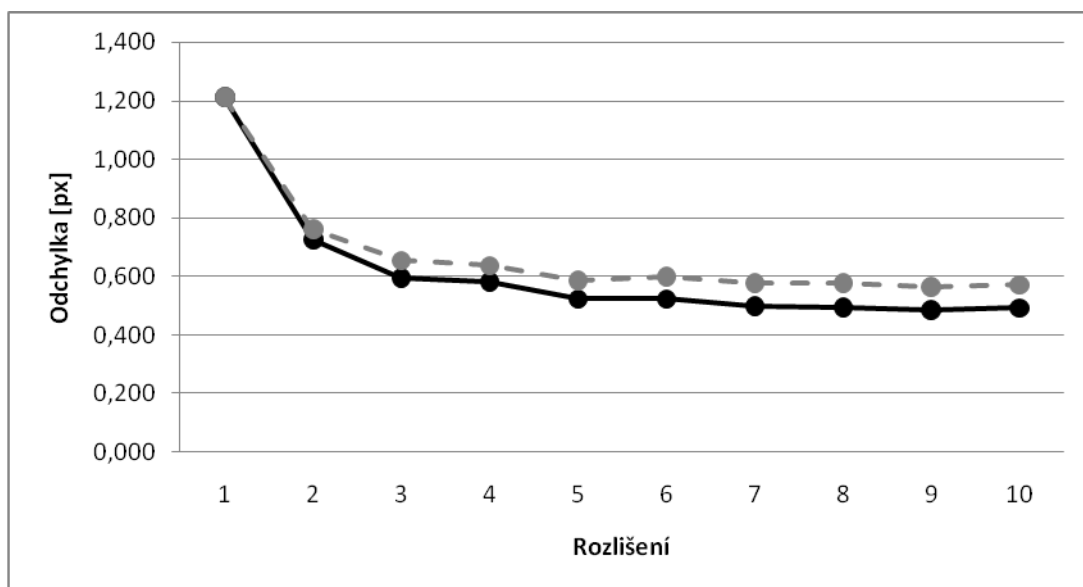
**Tabulka 1:** Přesnost rekonstrukce podle zvoleného rozlišení<sup>1</sup>

<sup>1</sup> Násobek počtu oken referenční mřížky



**Obrázek 23:** Relativní chyba

Obrázek 23 ukazuje poměrně rychlý nárůst relativní chyby se zvyšujícím se rozlišením rekonstrukce.



**Obrázek 24:** Odchylka v pixelech

Na obrázku 24 je naznačena odchylka v pixelech čárkovanou linkou bez použití korekce a plnou čarou s korekcí. Ukazuje se, že přínos korigování umístění interpolovaných bodů je 12,66%. Dále je z grafu patrné, že u vyšších rozlišení se odchylka zmenšuje jen pozvolna, popřípadě dokonce narůstá.

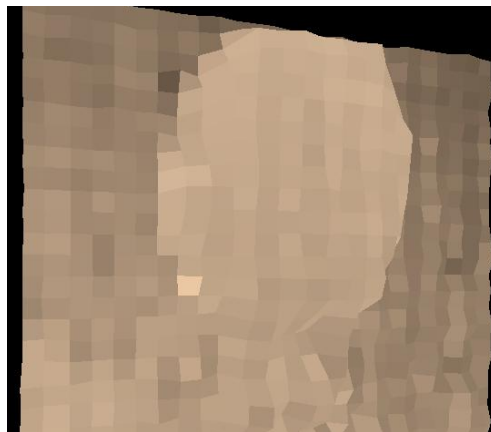
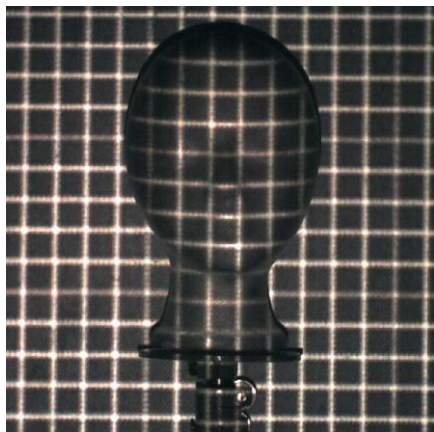
Jako ideální se tedy jeví zvolit rozlišení 2, kdy dochází ke značnému zmenšení průměrné odchylky a zároveň je relativní chyba ještě na únosné úrovni.



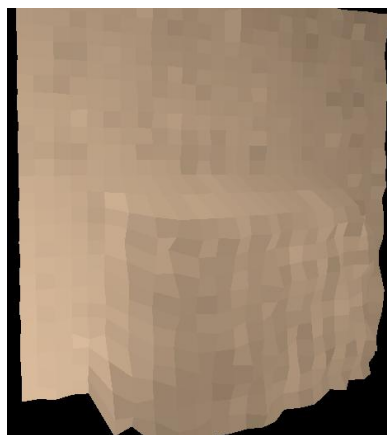
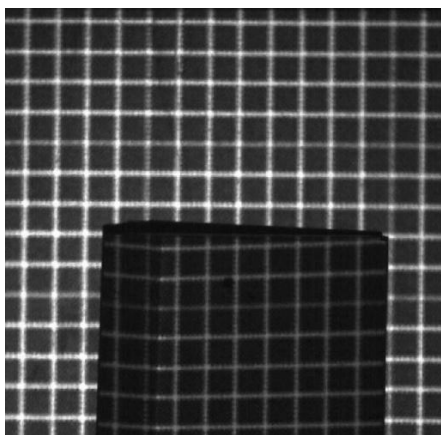
## 6.2 Výsledky rekonstrukce

Přesnost celé rekonstrukce by měla odpovídat relativní chybě detekce uzlových bodů. Tuto skutečnost je však ještě třeba experimentálně ověřit. Obecně by se ale chyba rekonstrukce při použití ideálních (synteticky vyrobených) snímků měla teoreticky blížit nule.

Pro demonstraci funkčnosti aplikace jsou zde uvedeny některé příklady rekonstrukce:



**Obrázek 25:** Ukázka 3D rekonstrukce



**Obrázek 26:** Ukázka 3D rekonstrukce

# Závěr

Dokument popsal metodu 3D rekonstrukce obličeje. Výsledný algoritmus je vhodný pro rekonstrukci obličeje a je plně funkční, čímž bylo splněno zadání práce. Je ale také vhodný pro rekonstrukci libovolných konvexních předmětů, které nemají příliš ostré rysy. Zároveň je i dostatečně robustní pro fungování v reálném prostředí. Kvalita snímků je ovšem přímo úměrná výsledku rekonstrukce. Pro získání přesných informací, je tedy především potřeba používat kvalitní snímací techniku a promítat dostatečně ostrou a zároveň jemnou mřížku.

Další vývoj povede ke zkoumání možností promítání mřížky a její snímání. Cílem tedy bude celou rekonstrukci ještě více zpřesnit. Toho může být teoreticky dosaženo i použitím různých projekčních mřížek. Ty nemusí být jen čtvercové (resp. obdélníkové), mohou mít v sobě například i diagonály. V úvahu přicházejí také mřížky nepravidelné, které by měly svůj tvar optimalizovaný pro zdůraznění tvaru obličeje. Podaří-li se metodu dostatečně zpřesnit, bude pravděpodobně využívána pro ověřování totožnosti osob, je jí ale možné využít k získání a uchování tvaru i jiných konvexních předmětů.

# Literatura

- [1] Wikipedia: The Free Encyclopedia: Čočka (optika) [online]. Aktualizováno 2008-03-31. Dostupný na URL: [http://cs.wikipedia.org/wiki/Čočka\\_\(optika\)](http://cs.wikipedia.org/wiki/Čočka_(optika)) (duben 2008)
- [2] Wikipedia: The Free Encyclopedia: Detekce hran [online]. Aktualizováno 2008-04-25. Dostupný na URL: [http://cs.wikipedia.org/wiki/Detekce\\_hran](http://cs.wikipedia.org/wiki/Detekce_hran) (duben 2008)
- [3] Wikipedia: The Free Encyclopedia: Konvoluce [online]. Aktualizováno 2008-02-15. Dostupný na URL: <http://cs.wikipedia.org/wiki/Konvoluce> (duben 2008)
- [4] Beránek, J.: Metody detekce a reprezentace hran v obraze [bakalářská práce], Brno, FIT VUT v Brne, 2007, s. 9-10. Dokument dostupný na URL: <http://www.fit.vutbr.cz/study/DP/rpfile.php?id=3045> (duben 2008).
- [5] Robotika.cz: OpenCV [online]. Aktualizováno 2003-09-22. Dostupný na URL: <http://robotika.cz/software/opencv/cs> (duben 2008)
- [6] Kršek, P., Španěl, M.: Prezentace přednášek předmětu Základy počítačové grafiky, FIT VUT Brno
- [7] Wikipedia: The Free Encyclopedia: OpenGL [online]. Aktualizováno 2008-02-28. Dostupný na URL: <http://cs.wikipedia.org/wiki/OpenGL> (duben 2008)
- [8] Wikipedia: The Free Encyclopedia: GLUT [online]. Aktualizováno 2008-01-11. Dostupný na URL: <http://cs.wikipedia.org/wiki/GLUT> (duben 2008)

# Seznam příloh

Příloha 1. Manuálová stránka programu

Příloha 2. Obsah CD

# Příloha 1: Manuálová stránka programu

```
*****  
*                                     *  
* Program Facer                       *  
* Autor: Zbynek Poulicek             *  
* xpouli01@stud.fit.vutbr.cz        *  
*                                     *  
*****
```

Program provadi 3D rekonstrukci ze snimku s promitnutou pravidelnou mřízkou

```
////////////////////////////////////
```

Pouziti: Facer -h  
Facer refImage srcImage sensitivity  
Facer refImage srcImage sensitivity resolution  
Facer refImage -e sensitivity  
Facer refImage -e sensitivity resolution

```
////////////////////////////////////
```

Popis parametru:  
-h Vypise tuto obrazovku s napovedou.  
refImage Obrazek s referencni mřízkou.  
srcImage Obrazek s rekonstruovanim predmetem.  
sensitivity Citlivost detektoru mřízky.  
resolution Rozliseni rekonstrukce.  
-e Za rekonstruovany obrazek se veme referencni a vypocete se presnost.

```
////////////////////////////////////
```

Sensitivity by melo priblizne odpovidat tloustce linky mřízky v pixelech

## **Příloha 2: Obsah CD**

- Elektronická verze této technické zprávy
- Programová dokumentace
- Spustitelný program
- Zdrojové soubory