

Štatistická analýza tvaru a obrazu

Interdisciplinárny prístup
postavený na matematických a štatistických základoch

Stanislav Katina

¹Ústav matematiky a štatistiky
Přírodovědecká fakulta
Masarykova Univerzita v Brně

Prednášky z analýzy tvaru [blok 1 a 2], ZS 2011



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

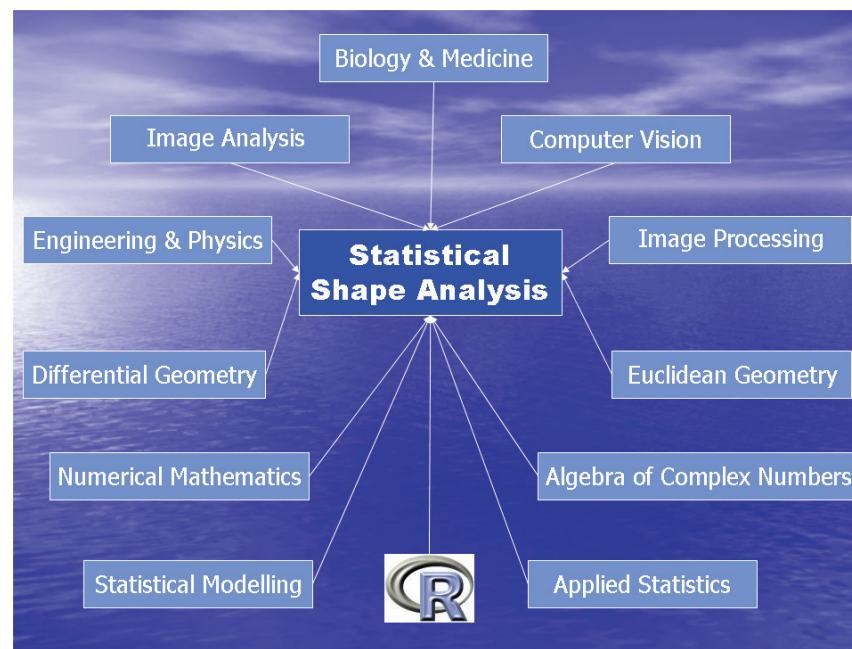


INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Stanislav Katina

Štatistická analýza tvaru a obrazu

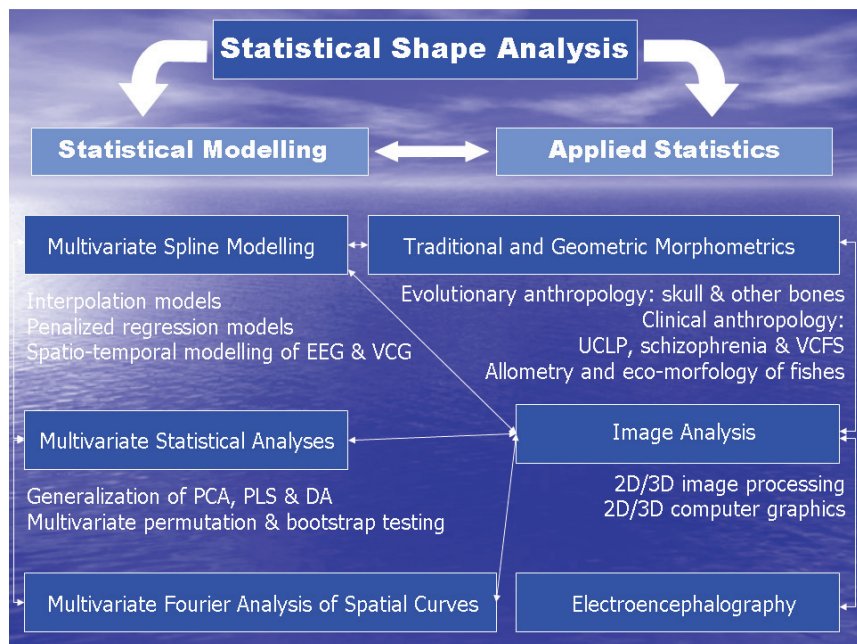
Shape Analysis—Interdisciplinary View



Stanislav Katina

Štatistická analýza tvaru a obrazu

Shape Analysis Vision \approx My Partial Research Tree



Stanislav Katina

Štatistická analýza tvaru a obrazu

Analýza obrazu

Digitálny obraz

Definition (Digitálny obraz – definícia)

Digitálny obraz I je dvoj-dimenzionálna funkcia prirodzených čísel vrátane nuly $\mathbb{N}_0 \times \mathbb{N}_0$ do možných hodnôt intenzity z množiny \mathbb{P} , teda

$$I : \mathbb{N}_0^2 \rightarrow \mathbb{P}; \text{ kde } u, v \in \mathbb{N}_0 \text{ a } I(u, v) \in \mathbb{P}.$$

Veľkosť obrazu je charakterizovaná jeho **šírkou** M (počet stĺpcov) a **výškou** N (počet riadkov) **obrazovej matice** I , kde $u_{\max} = M - 1$, $v_{\max} = N - 1$ a rozmery obrazu sú $M \times N$ **pixelov** (obrazových elementov). Čísla M, N sú zvyčajne rovné 2^k , kde k sa nazýva **bitová hĺbka**. Obrazový súradnicový systém sa riadi nasledovnými zásadami

- 1 y-ové súradnice idú zhora dole,
- 2 stred súradnicovej sústavy je bod $(u = 0, v = 0)$ ležiaci v ľavom hornom rohu.

Potom transformácia z obrazovej do karteziánskej súradnicovej sústavy bude nasledovná $I(u, v) \rightarrow I(N - 1 - v, u)$.

Stanislav Katina

Štatistická analýza tvaru a obrazu

Definition (Typy digitálneho obrazu)

- 1 Obraz v odtienoch sivej** – obraz pozostávajúci z jedného komponentu reprezentovaného intenzitou (nazývanou aj jas alebo hustota), ktorej hodnoty patria množine $\{0, 1, \dots, 2^k - 1\}$, zvyčajne $2^k - 1 = 255$; ($k = 8$) bitov (1 byte); 0 reprezentuje minimálny jas (čierna farba) a 255 the maximálny jas (biela farba).
- 2 Binárny obraz** – obraz, ktorý je špeciálnym typom obrazu v odtieni sivej, kde intenzita môže nadobúdať dve hodnoty nula a jedna (na pixel) a kóduje **farby čierna a biela**.
- 3 RGB farebný obraz** – obraz zložený z troch komponentov nazývaných aj primárne farby—**červená, zelená a modrá (RGB)**; typicky zaberajúce $k = 8$ bitov pre každý farebný komponent.

Definition (RGB farebný obraz)

RGB je **aditívny systém farieb**, čo znamená, že všetky farby sú vytvárané pridávaním primárnych farieb k základnej čiernej farbe. RGB môžeme vizualizovať ako troj-dimenzionálnu jednotkovú kocku (**RGB kocka**), kde osi tohoto systému nazývame osi primárnych farieb. Rozsah RGB hodnôt je $[0, C_{\max}]$. Každá možná farba korešponduje bodu C_i v RGB kocke

$$C_i = (R_i, G_i, B_i), \text{ kde } 0 \leq R_i, G_i, B_i \leq C_{\max}.$$

Rovinné usporiadanie farieb v skutočnom farebnom RGB obraze – jednotlivé farebné komponenty ležia v separátnych maticiach rovnakých rozmerov a funkcia intenzity má tvar $I = (I_R, I_G, I_B)$. Potom RGB farebný

obraz I je pole $M \times N \times 3$ typu $I = (I_R : I_G : I_B)$, kde I_R, I_G , a I_B sú $M \times N$ matice. Element (u, v, c) pola je definovaný ako $I_c(u, v)$, kde $c = R, G$ a B komponent. RGB metrika (vzdialenosť) nezodpovedá našmu zrakovému vnímaniu, t.j. RGB metrika a zrakové vnímanie sú neproporcionálne.

Definition (Transformácia RGB farieb do sivej škály)

Výsledkom transformácie RGB farieb do sivej škály je **iluminácia (jas) v sivej škále** definovaná ako

$$Y_g = \text{Avg}(R, G, B) = \frac{R + G + B}{3}.$$

Keďže červená a zelená sú vnímané ako oveľa jasnejšie ako modrá, výsledný obraz sa nám bude zdať tmavý v červených a zelených oblastiach a príliš svetlý v modrých. Preto je potrebné zaviesť **váženú ilumináciu (jas) v sivej škále**

$$Y_g = \text{Lum}(R, G, B) = w_R R + w_G G + w_B B,$$

kde $w_R = 0.2125$, $w_G = 0.7154$, and $w_B = 0.0720$ sú odporúčané váhy.

Bezfarebný (sivý) obraz definujeme ako obraz, kde každý RGB komponent má rovnakú hodnotu, t.j.

$$I_g(u, v) = \begin{pmatrix} R_{u,v}^{(g)} \\ G_{u,v}^{(g)} \\ B_{u,v}^{(g)} \end{pmatrix} \leftarrow \begin{pmatrix} Y_g \\ Y_g \\ Y_g \end{pmatrix}, \text{ kde } Y_g = \text{Lum}(R, G, B).$$

Definition (Obrazový histogram)

Obrazový histogram je histogram popisujúci početnosti hodnôt intenzity (jasu) obrazu. Histogram h obrazu v odtieňoch sivej I s hodnotami intenzity $I(u, v) \in [0, K - 1]$ obsahuje K hodnôt, kde pre typický 8 bitový obraz $K = 2^8 = 256$. Jednotlivé zložky histogramu sú definované ako

$$h(i) = \text{počet pixelov } v \text{ } I \text{ s intenzitou } i, \text{ pre všetky } i \in [0, K - 1],$$

$$h(i) = \text{card} \{ (u, v) | I(u, v) = i, i \in \mathbb{P} \}.$$

Interpretácia:

- 1 expozícia** – *pod- a preexponovaný obraz, dobre exponovaný obraz,*
- 2 kontrast** – rozsah hodnôt intenzity použitý v danom obraze; *plnokontrastový obraz*—efektívne používa celkový možný rozsah hodnôt intenzity $a \in [a_{\min}, a_{\max}]$ alebo $\{0, 1, \dots, K - 1\}$ (čierna–biela)
- 3 dynamický rozsah** – počet rozdielnych hodnôt intenzity v obraze (ideálne všetkých K hodnôt); pokiaľ máme $a \in [a_{\min} < a_{\text{low}}, a_{\text{high}} < a_{\max}]$ *maximálne možný dynamický rozsah* je možné dosiahnuť použitím všetkých možných hodnôt intenzity

Definition (Chyby alebo artefakty obrazu)

- 1 **chyby saturácie** – ideálne by mal byť rozsah kontrastu senzora väčší ako rozsah intenzity svetla snímanej scény, potom by bol histogram hladký na oboch koncoch; realita – často lesklé alebo tmave plochy; histogram je **saturovaný** na koncoch; signifikantné hroty na koncoch pri pod- a preexponovaných obrazoch
- 2 **chyby transformácie** – ideálne je rozdelenie intenzity hladké globálne ako aj lokálne; realita – zriedka v originálnom obraze, ale často v transformovanom obraze; **zvyšovanie kontrastu** vedie ku separácii hodnôt intenzity (diskretizácii; tvorbe dier); **znižovanie kontrastu** vedie ku zlučovaniu hodnôt intenzity, ktoré boli predtým rozdielne (diskretizácia; tvorba hrotov)
- 3 **chyby kompresie** – napr. počas kompresie do GIF je dynamický rozsah redukovaný na niekoľko hodnôt intenzity (**kvantovanie farieb**), tzv. **líniová štruktúra histogramu**
- 4 **chyby individuálnych komponent** – v ilumináčnom histograme (hist. intenzity sivej farby) neviditeľné chyby, ktoré sa objavujú v histogramoch jednotlivých komponent (**presaturovanie modrého komponentu**)

Definition (Bodové operácie)

Homogénna bodová operácia (globálna) – modifikácia intenzity bez zmeny veľkosti, geometrie alebo lokálnych štruktúr obrazu. Hodnoty intenzity a sú transformované na nové hodnoty a' použitím funkcie $f(a)$,

$$a' \leftarrow f(a) \text{ alebo } I'(u, v) \leftarrow f(I(u, v)), \text{ pre } \forall (u, v),$$

kde $f(\cdot)$ je **nezavislá na súradniciach** (u, v) , t.j. je všade rovnaká, napr.

- 1 globálna transformácia intenzity (jasu), kontrastu alebo farby
- 2 globálne kvantovanie obrazu a thresholding

Avšak funkcia $g(\cdot)$ ako **nehomogénna bodová operácia (lokálna)** berie do úvahy aj súradnice (u, v) , ale netransformuje ich na iné; t.j.

$$a' \leftarrow g(a, u, v) \text{ alebo } I'(u, v) \leftarrow g(I(u, v), u, v).$$

Napr.

- 1 lokálna transformácia intenzity (jasu), kontrastu alebo farby
- 2 lokálne kvantovanie obrazu a thresholding

Definition (Lineárne bodové operácie)

Lineárnu bodovú operáciu definujeme ako

$$f_{lio}(a) = k \cdot a + l,$$

kde k je nejaká **škálovacia konštanta intenzity** and l je **aditívna vyrovnávací konštanta intenzity** obrazu.

Saturačné (urezavacie, winsorizačné) podmienky definujeme nasledovne

- 1 ak $f_{lio}(a) < 0$, potom $f_{lio}(a) = 0$ (if $(a < 0)$ a $\leftarrow 0$)
- 2 ak $f_{lio}(a) > K - 1$ potom $f_{lio}(a) = K - 1$ (if $(a > K-1)$ a $\leftarrow K-1$)

Definition (Aditívne vyrovnávanie obrazu)

Aditívne vyrovnávanie obrazu: nech $k = 1, l \in \mathbb{Z}, |l| \leq K - 1$

$$f_{lio}(a) = a + l,$$

kde

- 1 $l \in \mathbb{Z}$, pretože chceme, aby intenzita bola kvantovaná z $\{0, 1, \dots, K - 1\}$
- 2 $|l| \leq K - 1$, pretože inak by bola intenzita mimo povoleného rozsahu
- 3 ak $l > 0$, potom transformovaný obraz bude **svetlejší** ako pôvodný
- 4 ak $l < 0$, potom transformovaný obraz bude **tmavší** ako pôvodný
- 5 l predstavuje **posun histogramu doľava alebo doprava**

Definition (Škálovanie obrazu)

Škálovanie obrazu: nech $l = 0$ a $k > 0$, potom

$$f_{lio}(a) = k \cdot a,$$

kde

- 1 $k > 0$, pretože $f_{lio}(a)$ musí byť kladné
- 2 nie je nutné, aby $k \in \mathbb{Z}$, pretože by sme mali len veľmi málo použiteľných možností
- 3 praktické zaokrúhľovanie (v prípade potreby) $f_{lio}(a) = \lfloor k \cdot a + 0.5 \rfloor$
- 4 ak $k > 1$, potom intenzita $f_{lio}(a)$ pokryje **širší interval hodnôt** ako a
- 5 ak $k < 1$, potom intenzita $f_{lio}(a)$ pokryje **užší interval hodnôt** ako a
- 6 **škálovanie naťahuje alebo stláča histogram v smere osi x**

Definition (Negatív obrazu)

Negatív obrazu: nech $k = -1$ a $l = K - 1$, potom

$$f_{lio}(a) = -a + (K - 1),$$

kde

- 1 škálovanie použitím $k = -1$ spôsobí **reverziu (flip) histogramu v smere osi x**
- 2 aditívna konštanta $l = K - 1$ spôsobuje, že **všetky transformované hodnoty sú kladné a patria do povoleného rozsahu**

Definition (Autokontrast)

Autokontrast:

$$f_{lio}(a) = k \cdot (a - c) + l, \text{ kde } l = a_{\min}, c = a_{\text{low}}, k = \frac{a_{\max} - a_{\min}}{a_{\text{high}} - a_{\text{low}}}, a_{\text{low}} \neq a_{\text{high}},$$

a intenzita je modifikovaná tak, aby jej hodnoty pokryli celý možný rozsah povolených hodnôt.

Definition (Thresholding)


Thresholding je špeciálnym typom kvantovania obrazu, ktoré separuje intenzitu do dvoch tried v závislosti na **prahovej konštante** a_{th} . **Prahová funkcia** $f_{\text{threshold}}(a)$ kategorizuje pixely do dvoch skupín, ktorým zodpovedajú hodnoty intenzity a_0 a a_1 , nasledovne

$$f_{\text{threshold}}(a) = \begin{cases} a_0 & \text{pre } a < a_{\text{th}} \\ a_1 & \text{pre } a \geq a_{\text{th}} \end{cases}, \text{ kde } 0 < a_{\text{th}} \leq a_{\max}$$

Typickými aplikáciami sú

- 1 **binarizácia intenzity obrazu** s hodnotami $a_0 = 0$ a $a_1 = 1$, ktorá v \mathbb{R} bude vyzeráť nasledovne
`a[which(a < a.th)] <- 0; a[which(a >= a.th)] <- 1.`
- 2 thresholding je najefektívnejší pri **bimodálnom histograme** – charakterizuje objekt a pozadie majúce rôznu priemernú intenzitu – tmavý objekt a svetlé pozadie alebo svetlý objekt a tmavé pozadie
- 3 cieľ – **separovať objekt od pozadia** alebo nájsť **obrysy objektu**

Example (R logo)

Majme \mathbb{R} logo  (rozmery 77×101 pixelov)—uložené ako **PPM (Portable Pixel Map, portable pixmaps)**. Aj napriek tomu, že ide o neefektívny formát rastrovaneho obrazu, je veľmi jednoduchý z hľadiska spracovania obrazu, a preto sa často používa.

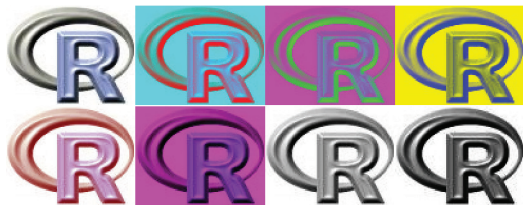
- 1 Načítajte a zobrazte toto logo M v \mathbb{R} .
- 2 Inveretujte \mathbb{R} (červený) komponent obrazu M .
- 3 Inveretujte \mathbb{G} (zelený) komponent obrazu M .
- 4 Inveretujte \mathbb{B} (modrý) komponent obrazu M .
- 5 Zvýraznite \mathbb{R} komponent obrazu M .
- 6 Odstráňte zelený komponent obrazu M .
- 7 Transformujte M do sivej škály.
- 8 Zvýraznite kontrast M v sivej škále pomocou funkcie $f(a) = a^k, k = 2$.

Example 10

Digitálny obraz

Riešenia:

- 1 knižnica `library(pixmap)`, príkaz
`M <- read.pnm(system.file("pictures/logo.ppm",
package="pixmap")) [1]`
- 2 `M1 <- M; M1@red <- 1-M@red; plot(M1)`
- 3 `M1 <- M; M1@green <- 1-M@green; plot(M1)`
- 4 `M1 <- M; M1@blue <- 1-M@blue; plot(M1)`
- 5 `M1 <- M; M1@red <- 0.5 + M@red/2; plot(M1)`
- 6 `M1 <- M; M1@green <- matrix(0,77,101); plot(M1)`
- 7 `M1 <- as(M,"pixmapGrey"); M2 <- M1; plot(M2)`
- 8 `M2 <- M1; M2@grey <- (M2@grey)^2; plot(M2)`



Obrázok: Bodové operácie s obrazom R logo

Stanislav Katina

Štatistická analýza tvaru a obrazu

Example 11

Digitálny obraz

Example (Binarizácia lastúry)

Majme lastúru (*Mytilus* sp.) uloženú ako PPM.

- 1 Načítajte obraz lastúry `M` v R.
- 2 Transformujte `M` do sivej škály a zobrazte použitím funkcie `plot()`.
- 3 Binarizujte obraz `M` pri thresholde 0.1 a vypočítajte počet pixelov lastúry.
- 4 Binarizujte obraz `M` pri thresholde 0.3 a vypočítajte počet pixelov lastúry.
- 5 Binarizujte obraz `M` pri thresholde 0.9 a vypočítajte počet pixelov lastúry.

Riešenia:

- 1 `library(pixmap); M <- read.pnm("mytilus.ppm")`
- 2 `M <- as(M,"pixmapGrey"); M1 <- M@grey plot(M,main="Grey
scale image")`
- 3 `M1 <- M@grey; M@grey[which(M1 >= 0.1)] <- 1; a.th <- .1
M@grey[which(M1 < a.th)] <- 0
plot(M,main="Binary image, threshold = 0.1")
length(M@grey[which(M1 < a.th)])`

Stanislav Katina

Štatistická analýza tvaru a obrazu

Example 11

Digitálny obraz

Grey scale image



Binary image, threshold = 0.3



Binary image, threshold = 0.1



Binary image, threshold = 0.9



Obrázok: Lastúra *Mytilus* sp. v sivej škále (vľavo hore) a binarizovaná pri rôznych hodnotách konštanty $a_{th} = 0.1, 0.3, 0.9$

Stanislav Katina

Štatistická analýza tvaru a obrazu

Example 12

Digitálny obraz

Example (Binarizácia lastúry, pokrač.)

Majme lastúru (*Mytilus* sp.) uloženú ako JPEG.

- 1 Transformujte obraz z formátu JPEG do formátu PPM.
- 2 Načítajte obraz lastúry `M` v R.
- 3 Transformujte `M` do sivej škály a zobrazte použitím funkcie `image()`.
Pozn.: Treba si uvedomiť, že **Pixel Aspect Ratio (PAR)**, kde PAR je rovné pomeru šírky a výšky pixela, nemusí byť rovné jednej, ale napr. 2/3, 3/4 alebo 9/16, čo sa dá ľahko ošetriť pomocou argumentu `asp=PAR` vo funkcii `image()`. Funkcia `plot()` priamo načítava PAR zo súboru PPM, a preto táto korekcia nebola potrebná. PAR je pre lastúru rovný 9/16. Na zobrazenie `M` použite všetky možné odtiene sivej (8-bitová škála sivej).
- 4 Na zobrazenie `M` použite len tri odtiene sivej (2-bitová škála sivej).
- 5 Na zobrazenie `M` použite len dva odtiene sivej (1-bitová škála sivej, monochromatický obraz).
- 6 Zobrazte `M` použitím funkcie `contour()` bez korekcie PAR.

Stanislav Katina

Štatistická analýza tvaru a obrazu

Example 12

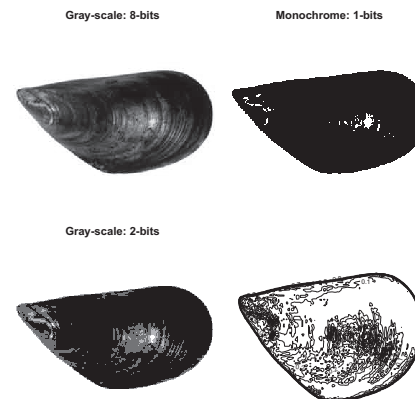
Digitálny obraz

Riešenia:

```
1 library(pixmap); library(rimage)
  shell("convert mytilus.jpg mytilus.ppm")
2 M <- read.pnm("mytilus.ppm")
3 M <- as(M, "pixmapGrey")
  image(t(M@grey[dim(M@grey)[1]:1,]), col=grey(0:255/255),
        asp=9/16, axes=FALSE, main="Gray-scale: 8-bits")
4 image(t(M@grey[dim(M@grey)[1]:1,]), col=grey(0:3/3),
        asp=9/16, axes=FALSE, main="Gray-scale: 2-bits")
5 image(t(M@grey[dim(M@grey)[1]:1,]), col=grey(0:1/1),
        asp=9/16, axes=FALSE, main="Monochrome: 1-bits")
6 contour(t(M@grey[dim(M@grey)[1]:1,]), axes=FALSE)
  title(main="Contour plot")
```

Example 12

Digitálny obraz



Obrázok: Lastúra *Mytilus* sp. v 8-bitovej sivej škále (vľavo hore), v 2-bitovej sivej škále (vľavo dole), v 1-bitovej sivej škále (vpravo hore) a kontúrový obraz použitím nesprávneho PAR

Example 13

Digitálny obraz

Example (Hmyzie krídlo, histogram intenzity a extrakcia súradníc landmarkov)

Majme hmyzie krídlo uložené ako **JPEG**.

- 1 Načítajte obraz krídla M v R.
- 2 Transformujte obraz z formátu JPEG do formátu PPM.
- 3 Transformujte M do sivej škály, zobrazte histogram intenzity M a zobrazte M použitím funkcie `plot()`.
- 4 Zvýraznite kontrast M v sivej škále pomocou funkcie $f(a) = a^k$, $k = 3$, zobrazte histogram intenzity M a zobrazte použitím funkcie `plot()`.
- 5 Použitím funkcie `locator()` lokalizujte 5 landmarkov (viď. obrázok), označte ich ako $+$ a extrahujte ich súradnice. Do obrázku dopíšete čísla landmarkov 1 – 5.
- 6 Lokalizujte hranice časti krídla (viď. obrázok) ako landmarky 6 – 10 použitím funkcie `locator()` a vykreslite polygón vnútri konvexného obalu týchto landmarkov pomocou funkcie `polygon()`.
- 7 Čo sme pri extrakcii landmarkov zanedbali?

Example 13

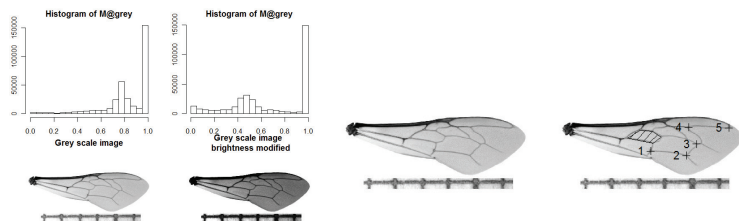
Digitálny obraz

Riešenia:

```
1 library(pixmap); library(rimage)
  M <- read.jpeg("wing.jpg")
2 shell("convert wing.jpg wing.ppm")
  M <- read.pnm("wing.ppm")
3 M1 <- M; M1 <- as(M1, "pixmapGrey")
  hist(M1@grey)
  plot(M1, main="Grey scale image")
4 M1@grey <- (M1@grey)^3
  hist(M@grey)
  plot(M, main="Grey scale image brightness modified")
5 plot(M)
  lok <- locator(5, type="p", pch=3)
  text(lok, pos=2, labels=1:5)
6 lok <- locator(5, type="l")
  polygon(lok, density=12)
```

Example 13

Digitálny obraz



Obrázok: Histogramy (prvý riadok vľavo) a obrázky v sivej škále, modifikovanej sivej škále (druhý riadok vľavo) obrázu hmyzie krídlo a obraz hmyzie krídlo v sivej škále s lokalizovanými landmarkami (vpravo)

Stanislav Katina

Štatistická analýza tvaru a obrázu

Example 14

Digitálny obraz

Example (Sánka, negatív obrazu a extrakcia súradníc landmarkov)

Majme sánku uloženú ako **JPEG**.

- 1 Načítajte obraz sánky M v R .
- 2 Transformujte M do sivej škály a zobrazte M použitím funkcie `plot()`.
- 3 Invertujte obraz M v sivej škále (vytvorte negatív obrazu) a zobrazte použitím funkcie `plot()`.
- 4 Použitím funkcie `locator()` odmerajte vzdialenosť 1cm.
- 5 Použitím funkcie `locator()` lokalizujte 10 landmarkov (viď. obrázok), označte ich ako o . Preškálujte extrahované súradnice na správnu mierku.

Riešenia: `library(rimage)`

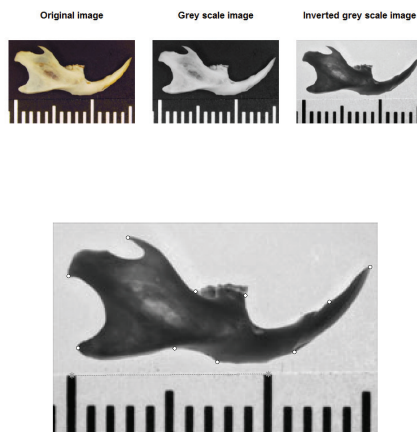
- 1 `M <- read.jpeg("jawd.jpg")`
- 2 `M <- rgb2grey(M); plot(M)`
- 3 `M <- 1 - M; plot(M)`
- 4 `lok <- locator(2, type="o", pch=8, lwd=2, col="grey60", lty="11")`
`scale.one <- sqrt(sum(diff(lok$x)^2 + diff(lok$y)^2))`
- 5 `b <- locator(10, type="p")`
`conf.mat <- rbind(bx, by) / scale.one`

Stanislav Katina

Štatistická analýza tvaru a obrázu

Example 14

Digitálny obraz



Obrázok: Obrázok sánky v RGB škále (prvý riadok vľavo), v sivej škále (prvý riadok v strede), negatív obrazu v sivej škále (prvý riadok vpravo) a negatív s lokalizovanými landmarkami (druhý riadok)

Stanislav Katina

Štatistická analýza tvaru a obrázu

Analýza obrázu

Uzavreté obrysy

Definition (Automatická extrakcia uzavretých obrysov a reťazové kódovanie)

Objekt v rovine je reprezentovaný jeho **vnútrom** a **obrysom**. Obrys môžeme automaticky extrahovať pomocou **reťazového (sekvenčného, Freemanovho) kódovania** definovaného pomocou sekvencie smerových zmien na diskretnom rastrovnom obraze.

- 1 definujeme **štartovací bod** \mathbf{x}_S vnútri objektu \mathcal{R} v rovine
- 2 uzavretý obrys je definovaný pomocou sekvencie bodov $\mathbf{c}_{\mathcal{R}} = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{C-1}]$, kde $\mathbf{x}_i = (u_i, v_i)$
- 3 jednotlivé elementy sekvencie $\mathbf{c}'_{\mathcal{R}} = [c'_0, c'_1, \dots, c'_{C-1}]$ definujeme ako $c'_i = \text{Code}(\Delta u_i, \Delta v_i)$, kde

$$(\Delta u_i, \Delta v_i) = \begin{cases} (u_{i+1} - u_i, v_{i+1} - v_i) & \text{pre } 0 \leq i < C - 1 \\ (u_0 - u_i, v_0 - v_i) & \text{pre } i = C - 1 \end{cases}$$

a $\text{Code}(\Delta u_i, \Delta v_i)$ definujeme pomocou **8-pixelového susedstva**

Δu	1	1	0	-1	-1	-1	0	0		3	2	1
Δv	0	1	1	1	0	-1	-1	-1		4	pixel	0
$\text{Code}(\Delta u_i, \Delta v_i)$	0	1	2	3	4	5	6	7		5	6	7

Stanislav Katina

Štatistická analýza tvaru a obrázu

Algoritmus:

1. I : binarizovaný obraz; $I(u, v) = 0$ (objekt), $I(u, v) = 1$ (pozadie)
2. lokalizujú súradnice štartovacieho pixela, $\mathbf{x}_S = (x_S, y_S)$, vnútri objektu tak, aby bolo jeho 4-pixelové susedstvo v objekte; potom transformujú obrazové súradnice od karteziánskych, kde $\mathbf{x}_S = (\dim(I)_{(1)} - y_S, x_S)$ (pozri slajd "Digitálny obraz – definícia")
3. fixujú $a = 1$ (štartovací bod pre pohyb z jedného pixela do druhého, kde index pixela korešponduje a , ktoré sa zvýši o jednotku, keď je nájdený ďalší pixel proti smeru hodinových ručičiek), $S = 6$ (štartovací bod; pixel č.6) a $SS = NA$ (reťazec); $\mathbf{x} = 0$; $\mathbf{y} = 0$ (štartovacie body; x - a y -ové súradnice); nech matica Δ má riadky Δu a Δv , potom $\mathbf{D} = (\Delta_{(\cdot,8)}; \Delta_{(\cdot,7)}; \Delta_{(\cdot,6)}; \Delta_{(\cdot,5)})$, kde $\Delta_{(\cdot,i)}$ je i -ty stĺpec matice Δ
4. while $(\mathbf{x}_{(a)}, \mathbf{y}_{(a)}) \neq \mathbf{x}_S$ (pokiaľ sa dosiahne opať štartovací bod) or dĺžka vektora \mathbf{x} je menšia ako 3 (vyhneme sa nekonečnej slučke) choď na (5) – (6)
5. if $|I(x_S + \mathbf{D}_{(1,S+1)}, y_S + \mathbf{D}_{(2,S+1)}) - I(x_S, y_S)| < \text{threshold}$, potom $a = a + 1$
 $\mathbf{x}_{(a)} = x_S$; $\mathbf{y}_{(a)} = y_S$
 $\mathbf{x}_S = \mathbf{x}_S + \mathbf{D}_{(\cdot,S+1)}$
 $SS_{(a)} = S + 1$; $S = (S + 7) \bmod 8$ (skontroluj pixel 5 a choď na pixel 5)

Algoritmus (pokrač.):

6. if else $|I(x_S + \mathbf{D}_{(1,S+2)}, y_S + \mathbf{D}_{(2,S+2)}) - I(x_S, y_S)| < \text{threshold}$, potom $a = a + 1$
 $\mathbf{x}_{(a)} = x_S$; $\mathbf{y}_{(a)} = y_S$
 $\mathbf{x}_S = \mathbf{x}_S + \mathbf{D}_{(\cdot,S+2)}$
 $SS_{(a)} = S + 2$; $S = (S + 7) \bmod 8$ (skontroluj pixel 6 a choď na pixel 6)
7. if else $|I(x_S + \mathbf{D}_{(1,S+3)}, y_S + \mathbf{D}_{(2,S+3)}) - I(x_S, y_S)| < \text{threshold}$, potom $a = a + 1$
 $\mathbf{x}_{(a)} = x_S$; $\mathbf{y}_{(a)} = y_S$
 $\mathbf{x}_S = \mathbf{x}_S + \mathbf{D}_{(\cdot,S+3)}$
 $SS_{(a)} = S + 3$; $S = (S + 7) \bmod 8$ (skontroluj pixel 7 a choď na pixel 7)
8. else choď na obrisy obrazu (smer – napr. diagonálne dole vpravo, $S = (S + 1) \bmod 8$; pokiaľ nenájdeš pixel s intenzitou menšou ako threshold ; t.j. prvý pixel pozadia)
9. return $\mathbf{x} = \mathbf{y}_{(-1)}$ a $\mathbf{a} = (\dim(I)_{(1)} - \mathbf{x})_{(-1)}$ (vymaž prvý element, ktorý je rovný poslednému – štartovací bod; pozri slajd "Digitálny obraz – definícia")

Example 15

Uzavreté obrisy

Example (pokrač. príkladu 11 a 12)

Majme lastúru (*Mitilus* sp.) uloženú ako PPM. Použite binarizovaný obraz M pri thresholde 0.9 z príkladu 11. Extrahujte súradnice obrisy lastúry.

1. Zobrazte binarizovaný obraz M (pri thresholde 0.9).
2. Pomocou funkcie `locator()` označte štartovací bod \mathbf{x}_S vnútri lastúry.
3. Na identifikáciu obrisy použite algoritmus `kontura`.
4. Resamplujte súradnice bodov obrisy na $k = 32$, kde body budú **ekvidišťantne vzdialené – s rovnakou uhlovou vzdialenosťou medzi nimi**, kde je potrebné vybrať $k = 32$ ekvidišťantných bodov z 629 identifikovaných bodov s použitím funkcie `seq(1, 629, length=32)`.
5. Resamplujte súradnice bodov obrisy na $k = 32$, kde body budú **ekvidišťantne vzdialené – s rovnakou radiálnou vzdialenosťou medzi nimi**, kde najprv vypočítate centroid lastúry (aritmetický priemer súradníc obrisy získaných v bode (4)), potom pomocou znalostí z analýzy komplexných čísel vyberiete tie z 629 súradníc identifikovaných algoritmom `kontura`, ktoré patria prieniku kontúry a ramien uhlov $\frac{2 \times i \times \pi}{k}$, $i = 0, 1 \dots k$, s vrcholom v bode (0, 0).

Example 15

Uzavreté obrisy

Riešenia:

```

1 plot(M, main="Binary image, threshold = 0.9")
2 x.start <- locator(1)
  x.start <- round(c(x.start$x, x.start$y))
3 myt.contour <- kontura(x.start, M@grey, start.threshold =
  0.1, threshold = 0.1) lines(myt.contour$x, myt.contour$y, lwd=3)
4 k.r=32; k <- length(myt.contour$x)
  myt.contour.x <- (myt.contour$x[seq(1, k, length=k.r)])
  myt.contour.y <- (myt.contour$y[seq(1, k, length=k.r)])
  plot(myt.contour.x, myt.contour.y, type="l", lwd=1.1,
  asp=1, axes=FALSE, main="Equidistantly spaced
  coordinates")
  points(myt.contour.x, myt.contour.y, pch=16, cex=0.7)
5 plot(myt.contour.x, myt.contour.y, type="l", lwd=1.1, asp=1, axes=FALSE,
  main="Radially spaced coordinates")
  mean.x <- mean(myt.contour$x)
  mean.y <- mean(myt.contour$y)
  points(mean.x, mean.y, pch=16)
  r.coords <- radial.coords(myt.contour$x, myt.contour$y, 32)
  points(mean.x + r.coords$coords[,1], mean.y +
  r.coords$coords[,2], pch=16, cex=0.7)
  for (i in 1:k.r) {
    segments(mean.x, mean.y,
             mean.x + r.coords$coords[,1],
             mean.y + r.coords$coords[,2])
  }

```


Example 15

Uzavreté obrysy

Riešenia (pokrač.):

```
"radial.coords" <- function(Rx, Ry, k1) {  
  k <- length(Rx)  
  M <- matrix(c(Rx, Ry), k, 2)  
  M1 <- matrix(c(Rx-mean(Rx), Ry-mean(Ry)), k, 2)  
  V1 <- complex(real = M1[,1], imaginary = M1[,2])  
  M2 <- matrix(c(Arg(V1), Mod(V1)), k, 2)  
  V2 <- NA  
  for (i in 0:(k1-1)) V2[i+1] <- which.max((cos(M2[,1] - 2*i*pi/k1)))  
  V2 <- sort(V2)  
  RES <- list("IDs" = V2, "radii" = M2[V2,2], "coords" = M1[V2,])  
  return(RES)  
}
```

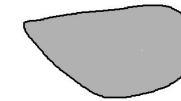
Example 15

Uzavreté obrysy

Grey scale image



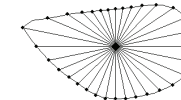
Binary image, threshold = 0.9



Equidistantly spaced coordinates



Radially spaced coordinates



Obrázok: Obrázok lastúry *Mytilus sp.* v sivej škále (prvý riadok vľavo), binarizovaný s extrahovaným obrysom (prvý riadok vpravo), ekvidištantné body obrysu s identickou uhlovou vzdialenosťou (druhý riadok vľavo) a s identickou radiálnou vzdialenosťou (druhý riadok vpravo)

Example 16

Uzavreté obrysy

Example (pokrač. príkladu 13)

Majme hmyzie krídlo uložené ako **JPEG**. Extrahujte súradnice obrysu krídla.

- 1 Binarizujte obraz M (pri thresholde 0.95).
- 2 Pomocou funkcie `locator()` označte štartovací bod x_S vnútri krídla.
- 3 Na identifikáciu obrysu použite algoritmus `kontura`.
- 4 Resamplujte súradnice bodov obrysu na $k = 64$, kde body budú **ekvidištantne vzdialené – s rovnakou uhlovou vzdialenosťou medzi nimi**.
- 5 Resamplujte súradnice bodov obrysu na $k = 64$, kde body budú **ekvidištantne vzdialené – s rovnakou radiálnou vzdialenosťou medzi nimi**.

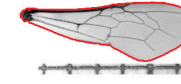
Example 16

Uzavreté obrysy

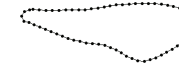
Grey scale image, threshold = 0.95



Original image



Equidistantly spaced coordinates



Radially spaced coordinates



Obrázok: Obrázok hmyzieho krídla binarizovaný s extrahovaným obrysom (prvý riadok vľavo), v sivej škále s extrahovaným obrysom (prvý riadok vpravo), ekvidištantné body obrysu s identickou uhlovou vzdialenosťou (druhý riadok vľavo) a s identickou radiálnou vzdialenosťou (druhý riadok vpravo)

Example 17

Uzavreté obrisy

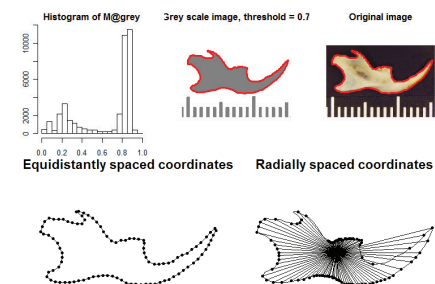
Example (pokrač. príkladu 14)

Majme sánku uloženú ako **JPEG**. Extrahujte súradnice obrisy sánky.

- 1 Binarizujte obraz M (pri thresholde 0.7).
- 2 Pomocou funkcie `locator()` označte štartovací bod x_s vnútri sánky.
- 3 Na identifikáciu obrisy použite algoritmus kontúra.
- 4 Resamplujte súradnice bodov obrisy na $k = 100$, kde body budú **ekvidištantne vzdialené – s rovnakou uhlovou vzdialenosťou medzi nimi**.
- 5 Resamplujte súradnice bodov obrisy na $k = 100$, kde body budú **ekvidištantne vzdialené – s rovnakou radiálnou vzdialenosťou medzi nimi**.
- 6 Je možné použiť súradnice bodov ekvidištantne vzdialených (s identickou radiálnou vzdialenosťou)? Ak nie prečo?

Example 17

Uzavreté obrisy



Obrázok: Histogram intenzity sivéj (prvý riadok vľavo), obrázok sánky binarizovaný s extrahovaným obrysom (prvý riadok v strede), v RGB škále s extrahovaným obrysom (prvý riadok vpravo), ekvidištantné body obrisy s identickou uhlovou vzdialenosťou (druhý riadok vľavo) a s identickou radiálnou vzdialenosťou (druhý riadok vpravo)

Example 18

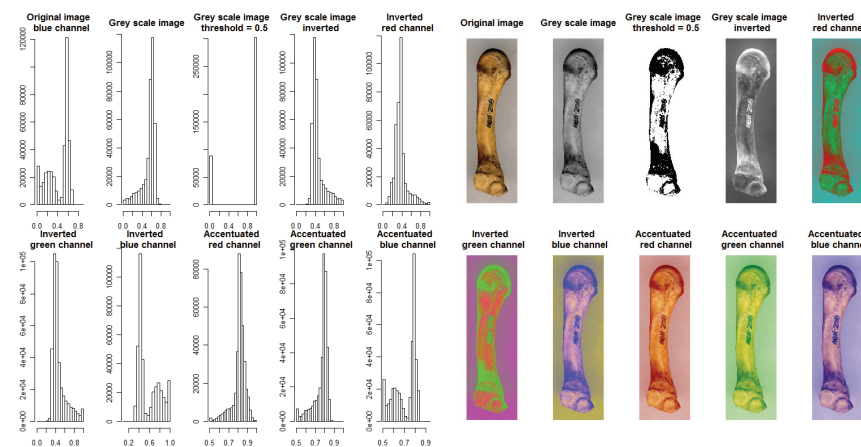
Uzavreté obrisy – zložitejšia situácia na metakarpe ľudskej ruky 1



Obrázok: Originálny obraz metakarpu ľudskej ruky

Example 18

Uzavreté obrisy – zložitejšia situácia na metakarpe ľudskej ruky 2



Obrázok: Histogramy rôznych transformácií farebných komponentov obrazu a nim zodpovedajúce obrazy metakarpu ľudskej ruky

Example 18

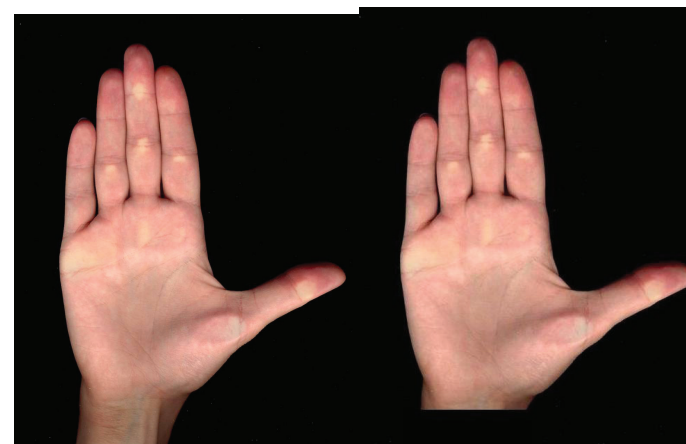
Uzavreté obrysy – zložitejšia situácia na metakarpe ľudskej ruky 3



Obrázok: Extrahovaný obrys metakarpu ľudskej ruky

Example 19

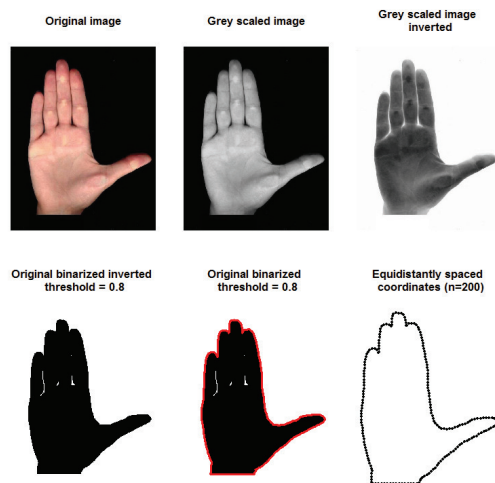
Uzavreté obrysy – zložitejšia situácia ľudskej ruky 1



Obrázok: Obraz ľudskej ruky

Example 19

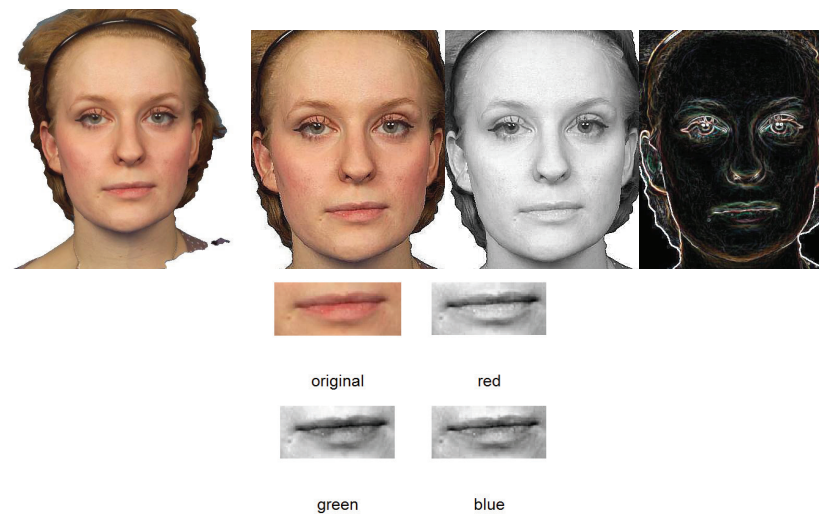
Uzavreté obrysy – zložitejšia situácia ľudskej ruky 2



Obrázok: Extrakcia obrysu ľudskej ruky

Example 20 A – otvorený problém

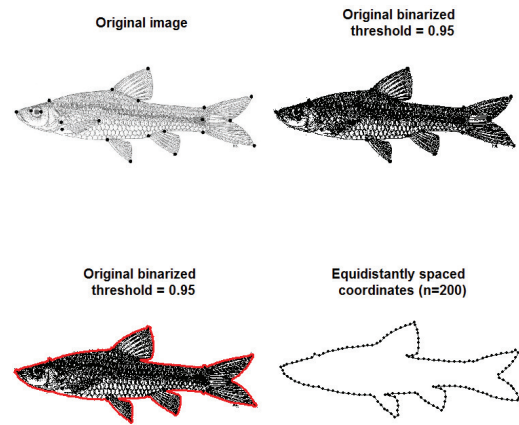
Uzavreté obrysy – (pravdepodobne) neriešiteľná situácia ľudskej tváre a pier



Obrázok: Obrazy rôznych transformácií komponentov obrazu ľudskej tváre a pier, extrahované hrany (vpravo hore)

Example 20 B – otvorený problém

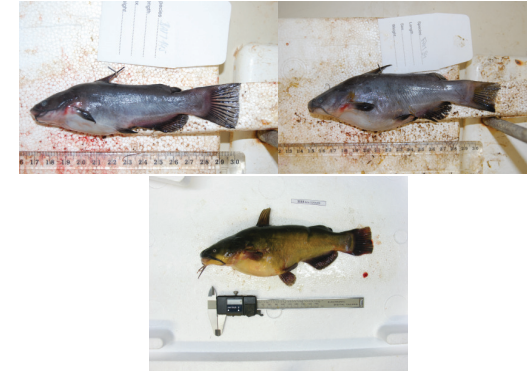
Uzavreté obrisy – stačí len extrakcie obrysu?



Obrázok: Extrakcia obrisy hrúzovca sieťovaného (*Pseudorasbora parva*)

Example 20 B – otvorený problém

Uzavreté obrisy a ich vnútro – potrebná extrakcie objektu ako celku



Obrázok: Extrakcia sumčeka čierneho (*Ameiurus melas*) z pozadia

Analýza obrazu

Geometrické operácie

Definition (Geometrické operácie)

Geometrické operácie transformujú obraz I do nového obrazu I' transformáciou súradníc jednotlivých pixelov,

$$I(u, v) \rightarrow I'(u', v'),$$

kde hodnoty intenzity obrazu I pôvodne v bode (u, v) sú transformované do bodu (u', v') v novom obraze I' . Transformačná funkcia má potom tvar

$$T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$$

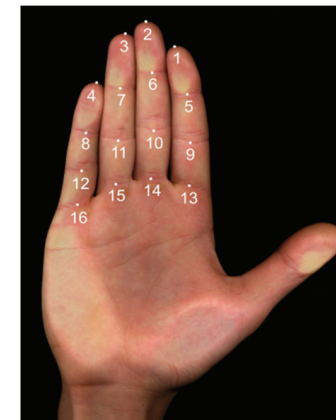
a je definovaná pre každý bod *vzorového obrazu* $\mathbf{x} = (u, v)$ a korešpondujúci bod *transformovaného obrazu* $\mathbf{x}' = (u', v')$, kde $\mathbf{x}' = T(\mathbf{x})$.

Príklady geometrických operácií:

- 1 **afinne transformácie** – **otočenie**, **posunutie**, **škálovanie**, **skosenie** a **zrkadlenie**
- 2 **TPS modely** – **interpoláčny thin-plate splajn (TPS) model** [IM1, IM3] a **penalizovaný TPS regresný model** [PRM1, PRM3]

Example 21

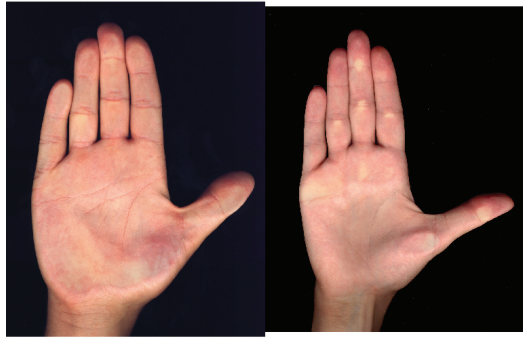
Geometrické operácie – **warping** ľudskej ruky 1 (pozor nie **morfing**)



Obrázok: Ľudská ruka a 16 landmarkov

Example 21

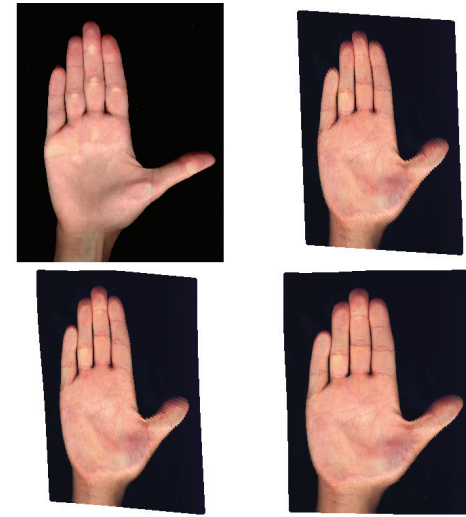
Geometrické operácie – **warping** ľudskej ruky 2 (pozor nie **morfing**)



Obrázok: Dve ľudské ruky – chceme transformovať ľavú na pravú

Example 21

Geometrické operácie – **warping** ľudskej ruky 3 (pozor nie **morfing**)



Obrázok: Ľudské ruky – vzorová ruka, odhadnutá ruka, affinná a neaffinná komponenta transformácie (po stĺpcoch) [approx. 1.6mil premenných]

Example 22

Geometrické operácie – **warping** Fredovej tváre (pozor, nie **morfing**)



Obrázok: Prof. Fred Bookstein – originálna (vľavo) a transformovaná fotografia (vpravo) [s láskavým dovolením zakladateľa odboru Analýza tvaru]

Analýza tvaru

Analýza obrysov

Definition (Obrys)

Obrys je **uzavretá krivka** definovaná súradnicami k bodov (semilandmarkov) patriacich tomuto obrysu, kde body sú

- 1 ekvidistantne vzdialené s rovnakou **radiálnou vzdialenosťou** medzi nimi.
- 2 ekvidistantne vzdialené s rovnakou **uhlovou vzdialenosťou** medzi nimi.

Definition (Analýza obrysov)

Štatistická analýza obrysu závisí od toho, o aký typ obrysu ide.

- 1 Ak ide o obrys typu (1), používa sa **radiálna Fourierova analýza**.
- 2 Ak ide o obrys typu (2), používa sa **tangenciálna Fourierova analýza** alebo **eliptická Fourierova analýza**.

2D/3D Fourierova analýza je zovšeobecnením klasickej Fourierovej analýzy používanej v časových radoch na analýzu periodického signálu v dátach, kde sa aplikuje rozklad Fourierovho radu použitím diskretných Fourierových transformácií. Služi ja na **výraznú redukciu dimenzii**.

Definition (Klasická Fourierova analýza)

Fourierov rozklad periodickej funkcie $f(t)$, kde $t \in \mathbb{R}^+$ s periódou $T_\lambda = \frac{\pi}{\lambda}$ v nejakých časových jednotkách, bude mať tvar

$$f(t) = \frac{a_0}{2} + \sum_{i=1}^p (a_i \cos(\lambda_i t) + b_i \sin(\lambda_i t)), \text{ kde } \lambda_i = \frac{i}{T} 2\pi$$

je i -ta frekvencia funkcie $f(t)$ v radiánoch $\lambda \in (0, 2\pi)$,

$$a_i = \frac{2}{T} \int_{t_1}^{t_2} f(t) \cos(\lambda_i t) dt; b_i = \frac{2}{T} \int_{t_1}^{t_2} f(t) \sin(\lambda_i t) dt$$

sú **párne a nepárne Fourierove koeficienty** a T je potrebné zvoliť. Aplikácia $f(t)$ predstavuje prepis do podoby **nelineárneho regresného modelu** tvaru

$$f(t) = a_{01} + a_{02}t + \sum_{i=1}^p (a_i \cos(\lambda_i t) + b_i \sin(\lambda_i t)) + \epsilon_t, \text{ kde } \lambda_i = i \frac{\pi}{T},$$

kde je potrebné odhadnúť $3p + 3$ parametrov a model linearizovať.

Definition (Radiálna Fourierova analýza, RFA)

Majme obrys centrováný do bodu $(0, 0)$. **Lúče (ramená)** $r_j, j = 1, 2, \dots, k$, **uhlov** θ_j s vrcholom v bode $(0, 0)$ je možné popísať **periodickou funkciou** $r(\theta)$ nejakého uhla θ nasledovne

$$r(\theta) = \frac{a_0}{2} + \sum_{i=1}^p (a_i \cos(\lambda_i \theta) + b_i \sin(\lambda_i \theta)), \text{ kde } \lambda_i = i.$$

Potom j -ta harmonická zložka je rovná $r(\theta_j)$, $a_0 = 2\sqrt{\sum_{j=1}^k r_j/k}$,

$$a_i = \frac{2}{k} \sum_{j=1}^k r_j \cos(i\theta_j); b_i = \frac{2}{k} \sum_{j=1}^k r_j \sin(i\theta_j)$$

sú **párne a nepárne Fourierove koeficienty**, k je **počet lúčov** a zároveň semilandmarkov obrysu a p je **počet frekvencií**. Musí byť splnená **podmienka** $p < \frac{k}{2}$, lebo máme dva parametre na jednu harmonickú zložku, ktorá je funkciou nejakého jedného uhla θ . **Označenia:** obrys \mathbf{M} a $\mathbf{M}_{RFA}^{(p)}$.

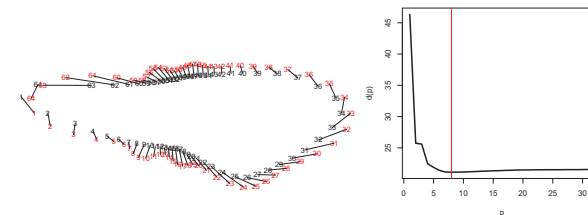
Metodologické poznámky:

- 1 všetky obrysy z náhodného výberu **musia byť automaticky extrahované pomocou reťazového kódu rovnakým smerom**
- 2 treba mať na zreteli, že **smer výpočtu lúčov v RFA je proti smeru hodinových ručičiek a odhadovanie je sekvenčné**
- 3 **štartovací bod a smer zoradenia bodov na obryse musia byť kompatibilné so štartovacím bodom a smerom odhadovania lúčov RFA**, teda proti smeru hodinových ručičiek; ak nie je, je potrebné zoradenie v smere hodinových ručičiek zmeniť na proti smeru hodinových ručičiek
- 4 **uhol a orientácia prvého ramena musí byť rovnaká** pri všetkých obrysoch z náhodného výberu – všetky obrysy je potrebné rotovať do tejto polohy, t.j. nulté rameno je na osi x -ovej, smeruje k jej kladnej polovici s vrcholom v bode $(0, 0)$, čo docielime centrovaním obrysu do jeho centroidu (aritmetický priemer súradníc bodov obrysu) a otočením nultého ramena do osi x
- 5 ak chceme porovnávať **obrysy párového typu** (ľavá a pravá strana), je potrebné extrahovať jednu stranu proti smeru hodinových ručičiek a druhú v smere hodinových ručičiek s **opačne orientovaným nultým uhlom** a potom napr. ľavú stranu transformovať pomocou osovej súmernosti (s osou v osi y)
- 6 **minimálny počet harmonických koeficientov** p sa odhadne ako $d_p = \arg \min_{\nu p} \|\mathbf{M} - \mathbf{M}_{RFA}^{(p)}\|^2$
- 7 RFA **nemôže byť použitá v prípadoch, keď aspoň jedno rameno (lúč) pretne obrys viac ako jedenkrát**

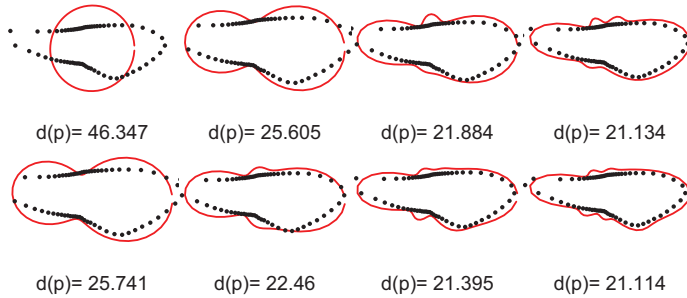
Example (pokrač. príkladu 16)

Majme hmyzie krídlo uložené ako **JPEG** a resamplované súradnice semilandmarkov obrysu pomocou radiálnych vzdialeností vypočítané v príklade 16.

- 1 Odhadnite obrys krídla pomocou RFA pri optimálnom p .



Obrázok: Obrys \mathbf{M} superponovaný s obrysom $\mathbf{M}_{RFA}^{(8)}$ so segmentami spájajúcimi korešpondujúce body, rozptylový graf počtu harmonických koeficientov voči d_p (suboptimálne $p = 8$)



Obrazok: Obrys M superponovaný s obrysami $M_{RFA}^{(\rho)}$, $\rho = 1, 2, \dots, 8$

Pozn.: Súradnice sú vypočítané v tvare **komplexných čísel**, kde modulus = $r(\theta_j)$, argument = θ_j .

Definition (Eliptická Fourierova analýza, EFA)

Majme obrys centrováný do bodu $(0, 0)$. Nech T je **obvod obrysu**, $\lambda = 2\pi/T$ je **frekvencia** a nech $t \in \langle 0, T \rangle$ je **chordálna (uhlová) vzdialenosť**. Potom je možné pomocou t vyjadriť súradnice semilandmarkov obrysu ako $x(t)$ a $y(t)$ nasledovne

$$x(t) = \frac{a_0}{2} + \sum_{i=1}^{\infty} (a_i \cos(i\lambda t) + b_i \sin(i\lambda t)), \text{ kde}$$

$$a_i = \frac{2}{T} \int_0^T x(t) \cos(i\lambda t) dt; b_i = \frac{2}{T} \int_0^T x(t) \sin(i\lambda t) dt,$$

$$y(t) = \frac{c_0}{2} + \sum_{i=1}^{\infty} (c_i \cos(i\lambda t) + d_i \sin(i\lambda t)), \text{ kde}$$

$$c_i = \frac{2}{T} \int_0^T y(t) \cos(i\lambda t) dt; d_i = \frac{2}{T} \int_0^T y(t) \sin(i\lambda t) dt.$$

Definition (EFA, pokrač.)

Ak je obrys definovaný pomocou k semilandmarkov, potom môžeme **Fourierove koeficienty** odhadnúť nasledovne

$$a_i = \frac{T}{2\pi^2 n^2} \sum_{j=1}^k \frac{\Delta x_j}{\Delta t_j} \left(\cos\left(i \frac{2\pi t_j}{T}\right) - \cos\left(i \frac{2\pi t_{j-1}}{T}\right) \right), \Delta x_1 = x_1 - x_k,$$

$$b_i = \frac{T}{2\pi^2 n^2} \sum_{j=1}^k \frac{\Delta x_j}{\Delta t_j} \left(\sin\left(i \frac{2\pi t_j}{T}\right) - \sin\left(i \frac{2\pi t_{j-1}}{T}\right) \right), a_0 = \frac{2}{T} \sum_{j=1}^p x_j,$$

$$c_i = \frac{T}{2\pi^2 n^2} \sum_{j=1}^k \frac{\Delta y_j}{\Delta t_j} \left(\cos\left(i \frac{2\pi t_j}{T}\right) - \cos\left(i \frac{2\pi t_{j-1}}{T}\right) \right), \Delta y_1 = y_1 - y_k,$$

$$d_i = \frac{T}{2\pi^2 n^2} \sum_{j=1}^k \frac{\Delta y_j}{\Delta t_j} \left(\sin\left(i \frac{2\pi t_j}{T}\right) - \sin\left(i \frac{2\pi t_{j-1}}{T}\right) \right), c_0 = \frac{2}{T} \sum_{j=1}^p y_j.$$

Vypočítané koeficienty a_0, a_i, b_i, c_0, c_i a d_i použijeme na odhad $x(t_j)$ a $y(t_j)$ dosadením do rovníc z predchádzajúceho slajdu, kde ∞ nahradíme p .

Metodologické poznámky:

- 1 všetky obrysy z náhodného výberu **musia byť automaticky extrahované pomocou reťazového kódu rovnakým smerom**
- 2 treba mať na zreteli, že **smer výpočtu EFA je identický so smerom zoradenia bodov obrysu, odhadovanie je sekvenčné a obe metódy majú rovnaký štartovací bod**
- 3 ak chceme porovnávať **obrysy párového typu** (ľavá a pravá strana), je potrebné extrahovať jednu stranu proti smeru hodinových ručičiek a druhú v smere hodinových ručičiek s opačne orientovanou hlavnou osou a potom napr. ľavú stranu transformovať pomocou osovej súmernosti (s osou v osi y)
- 4 **minimálny počet eliptických koeficientov p sa odhadne ako**

$$d_p = \arg \min_{\nu p} \left\| M - M_{RFA}^{(\rho)} \right\|^2$$
- 5 parametrická forma EFA umožňuje **jednoduché rozšírenie do 3D** pridaním $z(t)$
- 6 **väčšie eliptické koeficienty korešpondujú s väčšími elipsami, kde obrys vzniká kombináciou superponovaných elíps**
- 7 počet eliptických koeficientov môžeme odhadnúť pomocou **Furierovej sily** definovanej ako $Power_i = (a_i^2 + b_i^2 + c_i^2 + d_i^2)/2$, ktorá je proporcionálna amplitúde koeficientov a **kumulatívnej Furierovej sily** $\text{cumsum}(Power_i)$
- 8 najviac informácií o tvare obrysu je obsiahnutých v prvej elipse, pretože ide o najlepšiu aproximáciu obrysu

Definition (NEFA – normalizácia (štandardizácia) obrysu)

Normalizovaná eliptická Fourierova analýza (NEFA) je EFA invariantná na veľkosť a rotáciu **prvej elipsy** a štartovací bod, kde sú koeficienty a_i, b_i, c_i, d_i transformované na a'_i, b'_i, c'_i, d'_i použitím nasledovného algoritmu

$$\begin{pmatrix} a'_i & b'_i \\ c'_i & d'_i \end{pmatrix} = \frac{1}{s_e} \begin{pmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{pmatrix} \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix} \begin{pmatrix} \cos i\theta & -\sin i\theta \\ \sin i\theta & \cos i\theta \end{pmatrix},$$

kde s_e je **dĺžka hlavnej poloosi prvej elipsy**, ψ súvisí s **orientáciou elipsy** a θ s **rotáciou štartovacieho bodu** na koniec elipsy. Potom

$$\psi = 0.5 \arctan \frac{2(a_1 b_1 - c_1 d_1)}{a_1^2 + c_1^2 - b_1^2 - d_1^2}, s_e = \sqrt{a_*^2 + c_*^2}, \theta = \arctan \frac{c_*}{a_*},$$

kde $a_* = a_1 \cos \psi + b_1 \sin \psi$ a $c_* = c_1 \cos \psi + d_1 \sin \psi$. Koeficienty prvej eliptickej zložky $a'_1 = 1, b'_1 = c'_1 = 0$. Zostávajúci koeficient d'_1 súvisí s **eliptickou**

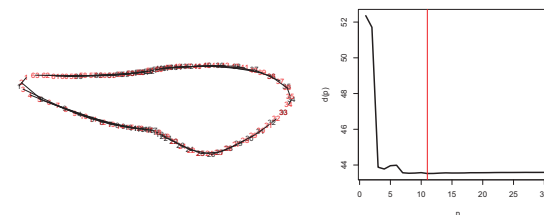
excentricitou, t.j. **šírko-dĺžkovým pomerom obrysu**.

Ak máme k dispozícii aj **súradnice landmarkov** na obryse, môžeme použiť na normalizáciu **GPA**.

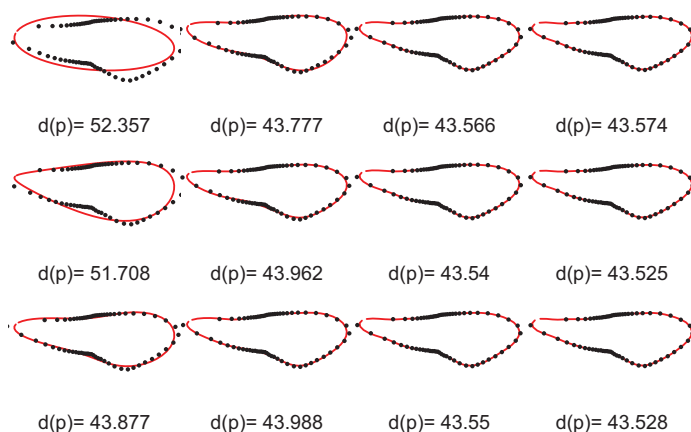
Example (pokrač. príkladu 16 a 23)

Majme hmyzie krídlo uložené ako **JPEG** a resamplované súradnice semilandmarkov obrysu pomocou uhlových vzdialeností vypočítané v príklade 16.

- 1 Odhadnite obrys krídla pomocou EFA pri optimálnom p .



Obrázok: Obrys M superponovaný s obrysom $M_{EFA}^{(11)}$ so segmentami spájajúcimi korešpondujúce body, rozptylový graf počtu harmonických koeficientov voči d_p (suboptimálne $p = 11$)



Obrázok: Obrys M superponovaný s obrysami $M_{EFA}^{(p)}, p = 1, 2, \dots, 12$

Definition (Tangenciálna Fourierova analýza, TFA)

Majme obrys centrováný do bodu $(0, 0)$. Semilandmarky obrysu sú ekvidistantne vzdialené s rovnakou uhlovou vzdialenosťou medzi nimi. Nech T je **obvod obrysu**, ktorý je pre jednoduchosť potrebné škálovať na 2π . Potom je možné popísať **kumulatívnu zmenu uhla dotykového vektora** $\phi(t)$ v jednotlivých bodoch obrysu ako funkciu (kumulatívnej) chordálnej (uhlovej) vzdialenosti t ako $\phi(t) = \theta(t) - \theta(0) - t$, kde $\theta(t)$ je **uhol dotykového vektora** vo vzdialenosti t , $\theta(0)$ je uhol dotykového vektora štartovacieho bodu a jeho odpočítanie slúži na štandardizáciu. Potom platí

$$\phi(\theta) = \frac{a_0}{2} + \sum_{i=1}^p (a_i \cos(i\theta) + b_i \sin(i\theta)), \text{ kde}$$

$$a_i = \frac{2}{k} \sum_{j=1}^k \phi(t) \cos(i\theta_j); b_i = \frac{2}{k} \sum_{j=1}^k \phi(t) \sin(i\theta_j); a_0 = 2 \sum_{j=1}^k \phi(t) / k.$$

Pozn.: Tak ako aj pri RFA – používame na výpočet **komplexné číslo**, v tomto prípade dostaneme Z , kde **modulus** = $2\pi/k$, **argument** = $\theta(t) = \phi(t) + \theta(0) + t$, výsledné súradnice obrysu sú **cumsum(Z)**, kt. musíme centrovať do $(0, 0)$.

Example 25

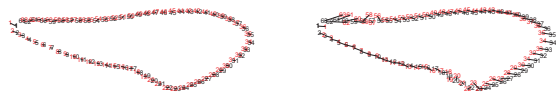
TFA 2

Example (pokrač. príkladu 16, 23 a 24)

Majme hmyzie krídlo uložené ako **JPEG** a resamplované súradnice semilandmarkov obrysu pomocou uhlových vzdialeností vypočítané v príklade 16.

1 Odhadnite obrys krídla pomocou TFA pri optimálnom p .

Pozn.: Najjednoduchší spôsob aproximácie uhla dotykového vektora je pomocou rozdielu dvoch susediacich bodov obrysu. Treba si uvedomiť, že vplyv uhla $\theta(0)$ môže mať výrazný efekt na odhad obrysu.



Obrázok: Obrys M superponovaný s obrysom $M_{TFA}^{(20)}$ so segmentami spájajúcimi korešpondujúce body (bez a s odpočítaním $\theta(0)$)

Stanislav Katina

Statistická analýza tvaru a obrazu

Example 25

TFA 3



$d(p) = 25.269$

$d(p) = 3.702$

$d(p) = 2.638$

$d(p) = 2.352$



$d(p) = 4.928$

$d(p) = 3.877$

$d(p) = 2.368$

$d(p) = 2.406$



$d(p) = 5.414$

$d(p) = 2.397$

$d(p) = 2.502$

$d(p) = 2.318$

Obrázok: Obrys M superponovaný s obrysami $M_{TFA}^{(p)}$, $p = 1, 2, \dots, 12$

Stanislav Katina

Statistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia

Definition (Matica ohybovej energie a ohybová energia)

Majme **TPS** model [IM3] definovaný ako

$$\begin{pmatrix} \mathbf{Y} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{S} & \mathbf{1}_k & \mathbf{X} \\ \mathbf{1}_k^T & \mathbf{0} & \mathbf{0} \\ \mathbf{X}^T & \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{W} \\ \mathbf{c}^T \\ \mathbf{A} \end{pmatrix}, \mathbf{L} = \begin{pmatrix} \mathbf{S} & \mathbf{1}_k & \mathbf{X} \\ \mathbf{1}_k^T & \mathbf{0} & \mathbf{0} \\ \mathbf{X}^T & \mathbf{0} & \mathbf{0} \end{pmatrix},$$

Nech je inverzia matice \mathbf{L} rovná

$$\mathbf{L}^{-1} = \begin{pmatrix} \mathbf{L}_{k \times k}^{11} & \mathbf{L}_{k \times 3}^{12} \\ \mathbf{L}_{3 \times k}^{21} & \mathbf{L}_{3 \times 3}^{22} \end{pmatrix},$$

potom

1 matica ohybovej energie: $\mathbf{B}_e = \mathbf{L}_{k \times k}^{11}$

2 ohybová energia alebo penalta:

$$J(\mathbf{f}) = \sum_{m=1}^2 \int \int_{\mathbb{R}^2} \left[\sum_{i,j} \left(\frac{\partial^2 f_m}{\partial x^{(i)} \partial x^{(j)}} \right)^2 \right] dx^{(1)} dx^{(2)}, \text{ s riešením modelu IM3}$$

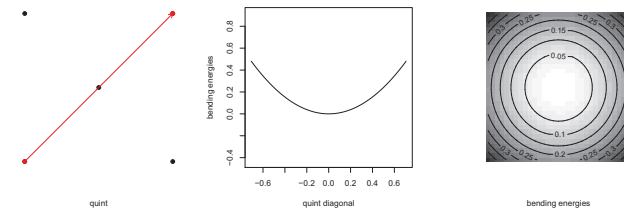
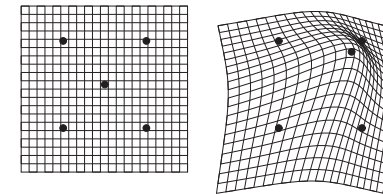
$$J(\mathbf{f}) = \text{tr}(\mathbf{W}^T \mathbf{S} \mathbf{W}) = \text{tr}(\mathbf{Y}^T \mathbf{B}_e \mathbf{Y})$$

Stanislav Katina

Statistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia



Obrázok: **Cirkularita ohybovej energie** vo vzťahu ku polohe piateho landmarku v prostriedku štvorca

Stanislav Katina

Statistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia

Definition (Analýza kriviek – posúvanie semilandmarkov po krivke)

Nech $\mathbf{X}_{k \times 2} = (\mathbf{x}_1, \dots, \mathbf{x}_k)^T = (\mathbf{x}^{(1)}, \mathbf{x}^{(2)})$ a $\mathbf{Y}_{k \times 2} = (\mathbf{y}_1, \dots, \mathbf{y}_k)^T = (\mathbf{y}^{(1)}, \mathbf{y}^{(2)})$ sú konfiguračné matice s riadkami $\mathbf{x}_j = (x_j^{(1)}, x_j^{(2)})^T$ a $\mathbf{y}_j = (y_j^{(1)}, y_j^{(2)})^T$, kde $\mathbf{x}^{(m)} = (x_1^{(m)}, x_2^{(m)}, \dots, x_k^{(m)})^T$, $\mathbf{y}^{(m)} = (y_1^{(m)}, y_2^{(m)}, \dots, y_k^{(m)})^T$, $j = 1, 2, \dots, k$ a $m = 1, 2$. Nech body sublistu \mathbf{y}_{j_i} **posúvame mimo ich pôvodnej polohy** \mathbf{x}_{j_i} , $i = 1, 2, \dots, q \leq k$, **pozdĺž tangenciálneho smeru** $\mathbf{u}_i = (u_i^{(1)}, u_i^{(2)})^T$, kde $\|\mathbf{u}\|_2 = 1$. Potom nová poloha \mathbf{x}_{j_i} je definovaná ako

$$\mathbf{y}_{j_i} = \mathbf{x}_{j_i} + t_i \mathbf{u}_i, i = 1, 2, \dots, q \leq k, \text{ kde } \mathbf{u}_i = \frac{\mathbf{x}_{i+1} - \mathbf{x}_{i-1}}{\|\mathbf{x}_{i+1} - \mathbf{x}_{i-1}\|_2} \text{ a } \mathbf{t} = (t_1, \dots, t_q)^T.$$

Je potrebné **minimalizovať kvadratickú formu**

$$\mathbf{y}^T \mathbf{B} \mathbf{y} = (\mathbf{x} + \mathbf{U} \mathbf{t})^T \mathbf{B} (\mathbf{x} + \mathbf{U} \mathbf{t}),$$

kde $\mathbf{x} = \text{Vec}(\mathbf{X})$ a $\mathbf{y} = \text{Vec}(\mathbf{Y})$, $\mathbf{B} = \text{diag}(\mathbf{B}_e, \mathbf{B}_e)$, \mathbf{B}_e je závislá iba na nejakej (referenčnej) konfiguračnej matici \mathbf{X}^* .

Stanislav Katina

Štatistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia

Definition (Analýza kriviek – posúvanie semilandmarkov po krivke; pokrač.)

Kvadratickú formu $\mathbf{y}^T \mathbf{B} \mathbf{y}$ minimalizujeme cez **hyperrovinu**

$$\mathbf{y} = \mathbf{x} + \mathbf{U} \mathbf{t},$$

kde \mathbf{X} je matica landmarkov vzoru (pôvodná poloha) a \mathbf{Y} je matica landmarkov obrazu (nová poloha), \mathbf{U} je matica riadkov dĺžky $2k$ a stĺpcov dĺžky q , kde (j, i) -ty element označujeme $u_i^{(j)}$ a $(k + j, i)$ -ty element $u_i^{(j)}$, na iných miestach sú umiestnené nuly. Vektor \mathbf{t} je riešením nasledujúcej rovnosti

$$\begin{aligned} \frac{\partial}{\partial \mathbf{t}} (\mathbf{x} + \mathbf{U} \mathbf{t})^T \mathbf{B} (\mathbf{x} + \mathbf{U} \mathbf{t}) &= \frac{\partial}{\partial \mathbf{t}} (\mathbf{x}^T \mathbf{B} \mathbf{x} + \mathbf{x}^T \mathbf{B} (\mathbf{U} \mathbf{t}) + (\mathbf{U} \mathbf{t})^T \mathbf{B} \mathbf{x} + (\mathbf{U} \mathbf{t})^T \mathbf{B} (\mathbf{U} \mathbf{t})) \\ &= \mathbf{x}^T \mathbf{B} \mathbf{U} + \mathbf{U}^T \mathbf{B} \mathbf{x} + 2 \mathbf{U}^T \mathbf{B} (\mathbf{U} \mathbf{t}) \\ &= 2(\mathbf{x}^T \mathbf{B} \mathbf{U} + \mathbf{U}^T \mathbf{B} (\mathbf{U} \mathbf{t})) = 0 \end{aligned}$$

Riešenie (podobné zovšeobecnenej MNS) má tvar

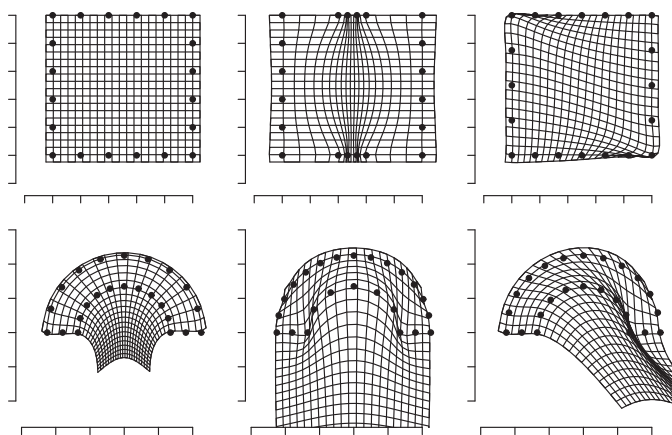
$$\mathbf{t} = -(\mathbf{U}^T \mathbf{B} \mathbf{U})^{-1} \mathbf{U}^T \mathbf{B} \mathbf{x}.$$

Stanislav Katina

Štatistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia



Obrázok: Posúvanie bodov na krivke pri troch situáciách a dvoch tvaroch

Stanislav Katina

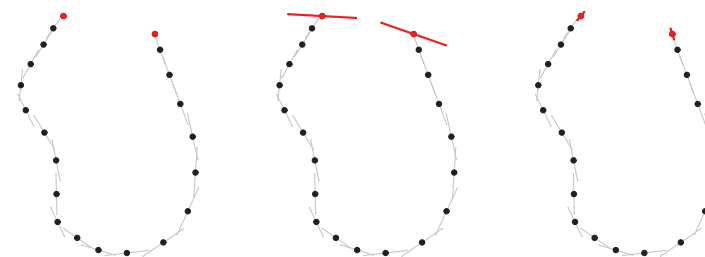
Štatistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia

Example (Symphyseálna krivka, pokrač.)

Majme symphyseálnu krivku z príkladu 3, kde $k = 21$, dáta (symphysis). Na **optimalizáciu polohy semilandmarkov na krivke v zmysle ohybovej energie (geometrická homológia)** použijeme konfiguračnú maticu \mathbf{X} a referenčnú (napr. priemernú) krivku \mathbf{X}_R (voči ktorej prebieha optimalizácia).



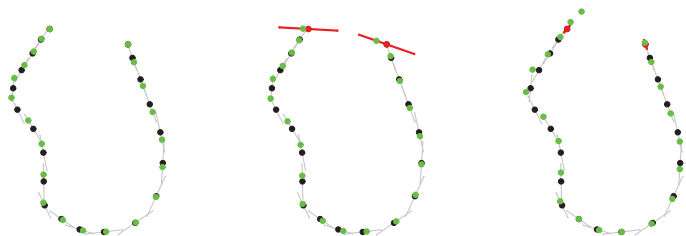
Obrázok: Posúvanie bodov na krivke pri troch situáciách – rôzne chápanie krivky (uzavretá vs otvorená), koncové body fixované vs voľné na posúvanie [dotyčnice v smere \mathbf{u}_i , $i = 1, 2, \dots, k$; ozn. • pôvodné pozície \mathbf{x}_i]

Stanislav Katina

Štatistická analýza tvaru a obrazu

Analýza tvaru

Analýza kriviek (sliding on curves) – geometrická homológia



Obrázok: Posúvanie bodov na krivke pri troch situáciách – rôzne chápanie krivky (uzavretá vs otvorená), koncové body fixované vs voľné na posúvanie [dotyčnice v smere \mathbf{u}_i s riešeniami $\mathbf{y}_i = \mathbf{x}_i + t_i \mathbf{u}_i, i = 1, 2, \dots, q$, ozn. \bullet pôvodné pozície \mathbf{x}_i , \bullet odhadnuté pozície \mathbf{y}_i]

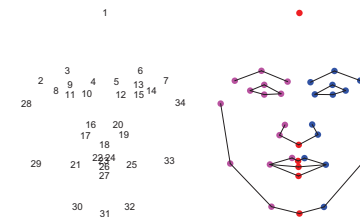
Stanislav Katina

Statistická analýza tvaru a obrazu

2D tvár – snímok z klasického fotoaparátu

Dpt. of Anthropology, University of Vienna, Vienna, Austria

20 dievčat, 19 – 31ročných, 46 + 26 (semi)landmarkov



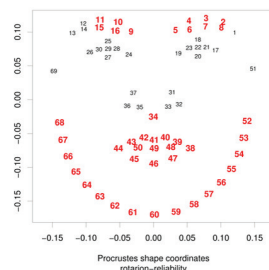
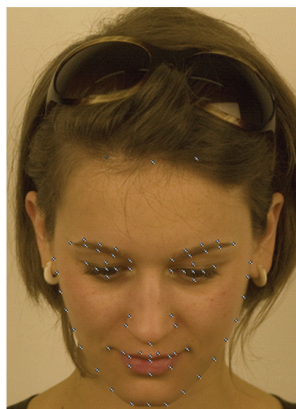
Obrázok: (Semi)landmarky na ľudskej tvári a **pravo-ľavá (ne)kompatibilita kódovania semilandmarkov na krivkách** [podmnožina (semi)landmarkov]

Stanislav Katina

Statistická analýza tvaru a obrazu

2D tvár – snímok z klasického fotoaparátu

Dpt. of Anthropology, University of Vienna, Vienna, Austria



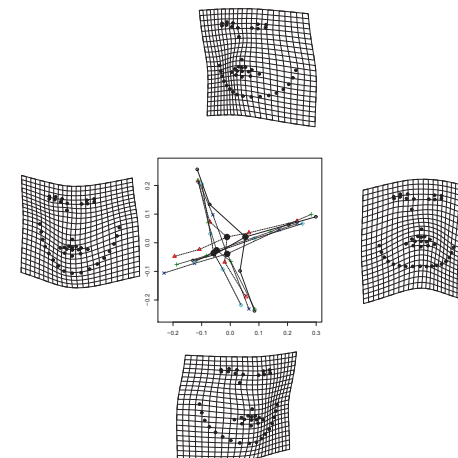
Obrázok: Reliabilita náklonu hlavy pri snímaní v rôznych uhloch [podmnožina (semi)landmarkov] a jej PCA

Stanislav Katina

Statistická analýza tvaru a obrazu

2D tvár – snímok z klasického fotoaparátu

Dpt. of Anthropology, University of Vienna, Vienna, Austria



Obrázok: PCA reliability náklonu hlavy pri snímaní v rôznych uhloch [podmnožina (semi)landmarkov]


Stanislav Katina

Statistická analýza tvaru a obrazu

2D tvár

Získavanie dát a ich úprava v PC

Získavanie dát a ich úprava v PC:

- 1 **nasnímanie dát klasickým fotoaparátom** – protokol snímania (dotazník, kalibrácia snímacieho systému, participanti a časový harmonogram)
- 2 **extrakcia 2D súradníc a RGB farieb** – z .jpeg a .tiff súborov a pod. do .dmp súborov čitateľných v 
- 3 **validačná štúdia (štúdia reliability)** – vhodná/optimálna orientácia tváre v anatomickom súradnicovom systéme, opakovateľnosť presnosti snímania/merania (Technical Measurement Error, TEM), lineárny regresný model so zmiešanými efektami
- 4 **(polo)automatické meranie/extrakcia súradníc (semi)landmarkov, kriviek** $[(46 + 26) \times 2$ (semi)landmarkov]
- 5 iteratívny výpočet **súradníc geometricky homologických semilandmarkov na krivkách** použitím **TPS warpingu** $[(46 + 26) \times 2$ (semi)landmarkov a viac ako 2mil pixelov]
- 6 **výpočet symetrizovaného priemerného tvaru (template)** (semi)landmarky na referenčnej tvári a preznačenej a zrkadlovo súmernej tvári musia byť spriemerované

Stanislav Katina

Statistická analýza tvaru a obrazu

3D tvár – snímok zo stereo-kamerového systému

Dental clinic, The University of Glasgow, UK; Face 3D data



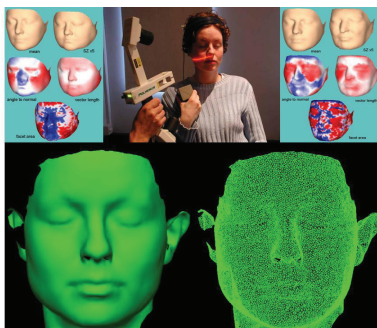
Stanislav Katina

Statistická analýza tvaru a obrazu

3D tvár – snímok z laserového skaneru

Royal College of Surgeons in Ireland, Dublin; Face 3D data

42 párov naskenovaných tvárí, 23 landmarkov, 1664 geometricky homologických semilandmarkov na krivkách a ploche, 59242 bodov plochy trinagulovaných použitím 117386 trojuholníkov



Obrázok: VCFS tvár, laserový skaner a (ne)triangulované semilandmarky na ploche


Stanislav Katina

Statistická analýza tvaru a obrazu

3D tvár

Získavanie dát a ich úprava v PC

Získavanie dát a ich úprava v PC:

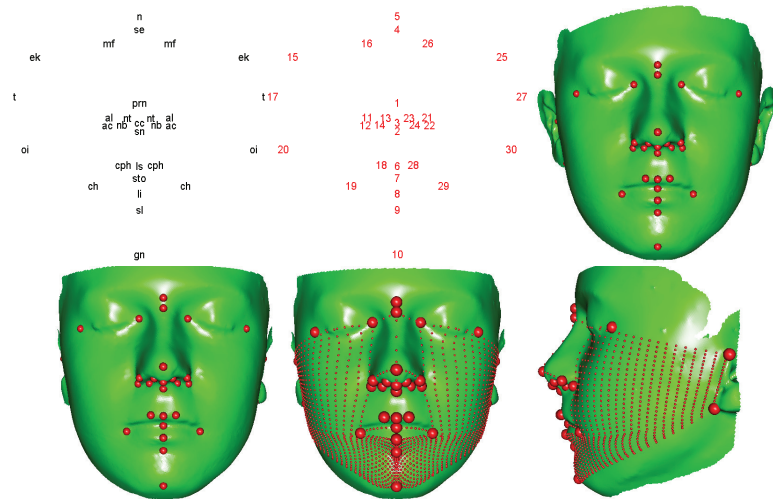
- 1 **nasnímanie dát stereo-kamerovým systémom alebo laserovým skanerom** – protokol snímania (dotazník, kalibrácia snímacieho systému, participanti a časový harmonogram)
- 2 **extrakcia 3D súradníc, normál bodov na ploche, trinagulácie a RGB farieb** – z .obj, .wrl a .jpeg súborov do .dmp súborov čitateľných v 
- 3 **validačná štúdia (štúdia reliability)** – vhodná/optimálna orientácia tváre v anatomickom súradnicovom systéme, opakovateľnosť presnosti snímania/merania (Technical Measurement Error, TEM), lineárny regresný model so zmiešanými efektami
- 4 **(polo)automatické meranie/extrakcia súradníc (semi)landmarkov, kriviek a plôch** $[1664 \times 3 = 4992, 4992 \times 42 = 209664$ bodov]
- 5 iteratívny výpočet **súradníc geometricky homologických semilandmarkov na krivkách/plochách a bodov na ploche** použitím **TPS warpingu** $[59242 \times 3 = 177726; 177726 \times 42 = 7464492]$
- 6 **výpočet symetrizovaného priemerného tvaru (template)** (semi)landmarky na referenčnej tvári a preznačenej a zrkadlovo súmernej tvári musia byť spriemerované; plocha tiež symetrizovaná

Stanislav Katina

Statistická analýza tvaru a obrazu

3D tvár – snímok z laserového skaneru

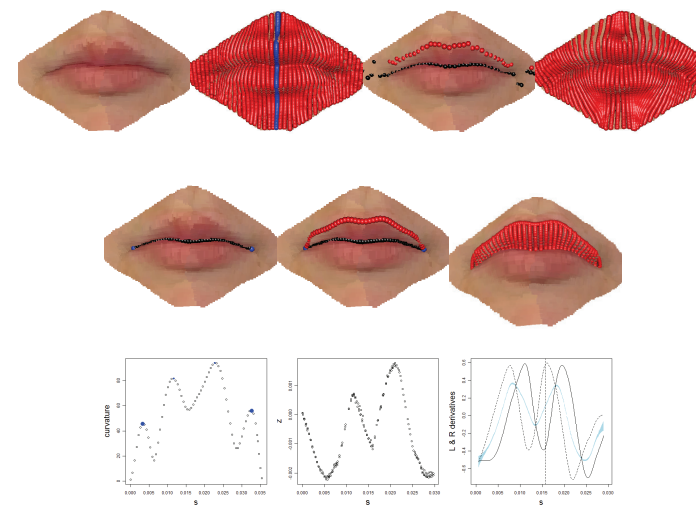
Royal College of Surgeons in Ireland, Dublin; Face 3D data



Obrázok: Symetrizovaná vzorová tvár (template) a (semi)landmarky

Budúcnosť analýzy tvaru

Automatická extrakcia diferenciálno-geometrických štruktúr z biologických objektov



Obrázok: Sekcencia automatickej extrakcie ľudských pier

Budúcnosť analýzy tvaru

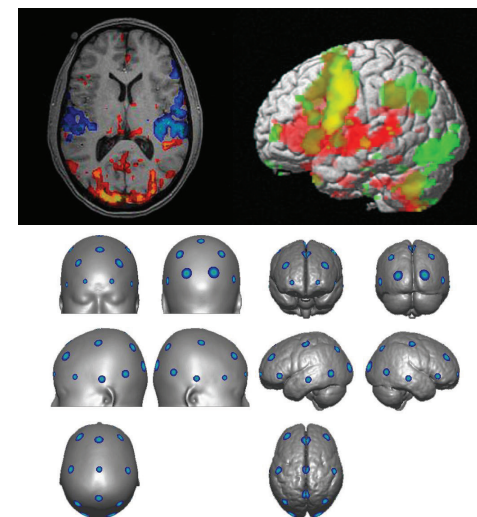
Automatická extrakcia diferenciálno-geometrických štruktúr z biologických objektov



Obrázok: Automatická extrakcia landmarkov, kriviek a anatomických plôch

Budúcnosť analýzy tvaru

Analýza tvaru EEG – časo-priestorové modelovanie



Obrázok: Fúzia analýzy tvaru, EEG a zobrazovacích techník mozgu