

Matematika pro geometrickou morfometrii (3)

Ján Dupej (jdupej@cgg.mff.cuni.cz)

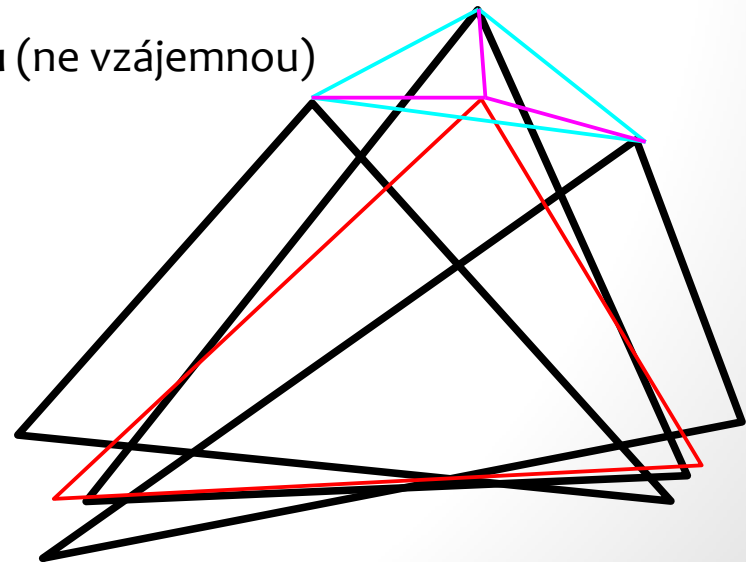
Laboratoř 3D zobrazovacích a analytických metod
Katedra antropologie a genetiky člověka
Přírodovědecká fakulta UK v Praze



Computer
Graphics
Charles
University

Opakování

- Prokrustovská transformace (analýza, registrace)
 - Idea
 - Nalezení rigidní transformace která minimalizuje **vzájemnou** vzdálenost eliminuje rozdílnou polohu a velikost
 - Algoritmus
 - Rozklad na dva problémy (dvojice a množina)
 - Těžiště v počátku
 - Jednotková velikost
 - Minimalizace vzdálenosti **od průměru** (ne vzájemnou)
 - Suboptimální



Opakování

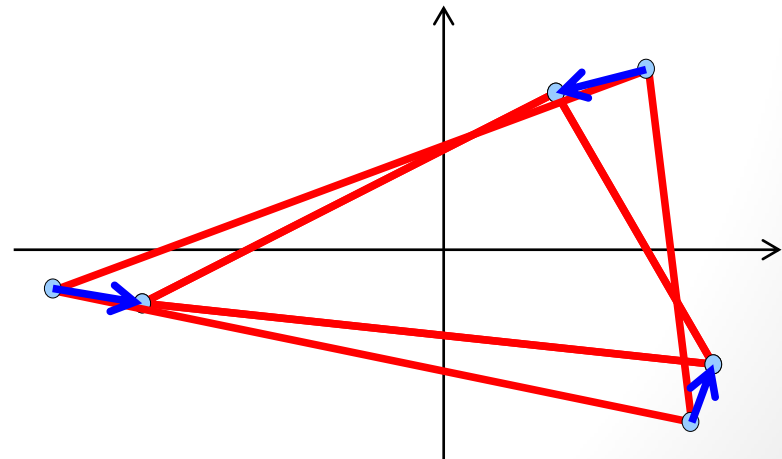
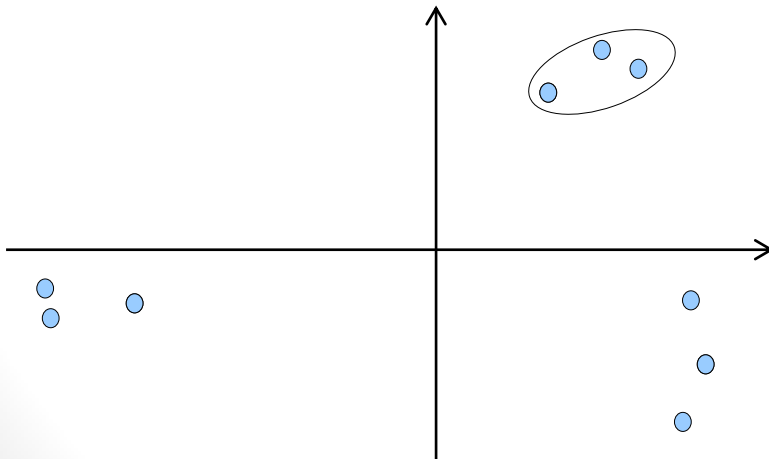
- Prokrustovská transformace
 - Postupná aproximace průměrného tvaru
 - Zarovnání dvojice (jedinec na aproximaci průměrného tvaru)
- Výhody
 - 2D, 3D, rychlá, stabilní (malá změna vstupních dat = malá změna výsledku)
 - Jednotková velikost
- Nevýhody
 - Rozloží chybu mezi landmarky tak aby celková byla nejmenší
 - Všechny landmarky mají stejnou váhu – Pinocchio efekt

Opakování

- Teorie tvaru
 - Stupeň volnosti = minimální počet čísel popisující tvar
 - Může jich být více, pak tvary tvoří zakřivený podprostor v tomto prostoru (např. po Prokrustovské analýze)
 - Statistické metody nefungují na zakřiveném prostoru
 - Projekce do tečné roviny pro malé vzdálenosti
 - Metody odvozené pro Kendallův prostor
- Vzdálenost mezi tvary
 - Proskrustovská
 - Částečná Prokrustovská
 - Úplná Prokrustovská – „indukuje“ Kendallův prostor

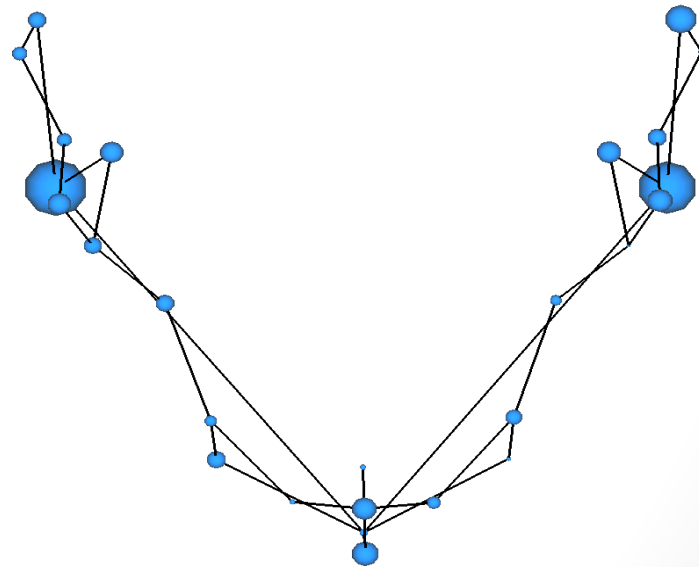
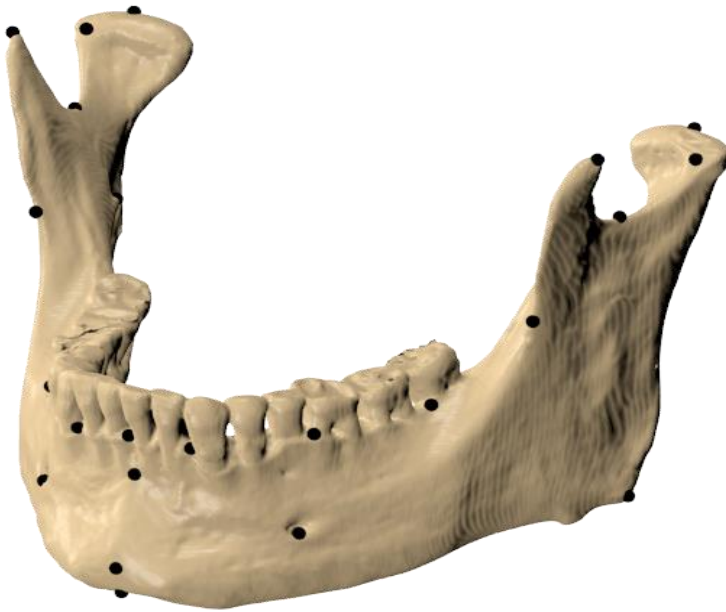
Zobrazení rozdílů

- Skupina nebo dvojice
- Pro jednoduché konfigurace mohou stačit překryté exempláře
- Scatter plot



Zobrazení rozdílů

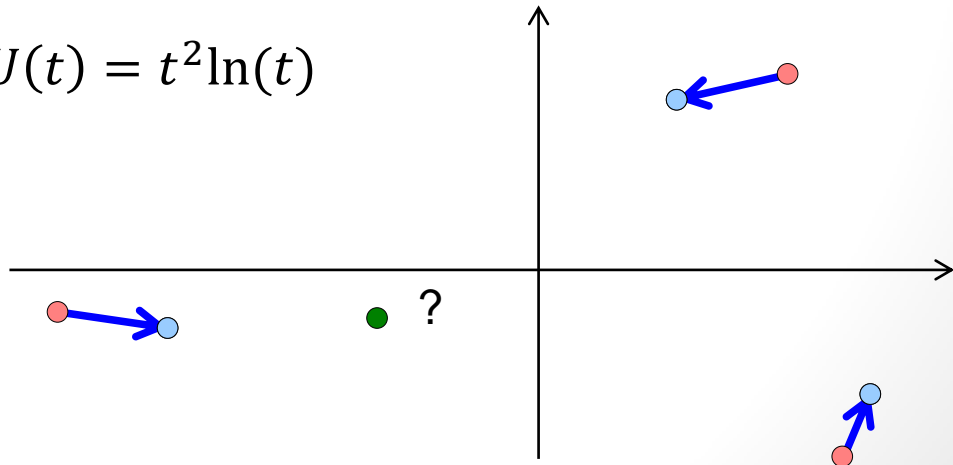
- Velikost, barva, orientace symbolů
- Složitější ve 3D



Thin Plate Spline

- Větší počet landmarků – mnoho šipek, nepřehledné
- Zobrazení pomocí deformované mřížky
- Vytvoření spojitého prostoru pomocí interpolace
 - Znáám posun landmarků
 - Co je mezi, spočítám interpolací

- $TPS(\mathbf{x}) = \mathbf{a}_0 + \mathbf{a}_1x + \mathbf{a}_2y + \sum_{i=1}^n \mathbf{w}_i U(\|\mathbf{p}_i - \mathbf{x}\|)$
 - $\mathbf{x} = (x, y)$
 - Radial basis function: $U(t) = t^2 \ln(t)$



TPS odvození

$$TPS_x(\mathbf{x}) = \underbrace{a_{0x} + a_{1x}x + a_{2x}y}_{\text{Afinní část}} + \underbrace{\sum_{i=1}^n w_{ix} U(\|\mathbf{p}_i - \mathbf{x}\|)}_{\text{Elastická část}}$$

Neznámé

- Známe:

$$TPS_x(\mathbf{p}_j) = \mathbf{q}_{jx} \quad j = 1, \dots, n$$

- $3 + n$ neznámých, n rovnic – chybí tři rovnice abychom měli jednoznačné řešení

$$\sum_{i=1}^n w_{ix} = 0$$

$$\sum_{i=1}^n w_{ix} p_{ix} = 0$$

$$\sum_{i=1}^n w_{ix} p_{iy} = 0$$

TPS odvození

- Soustava rovnic

$$(1, p_{jx}, p_{jy}, U_{1j}, \dots, U_{nj})$$

$$TPS_x(\mathbf{p}_j) = a_{0x}1 + a_{1x}p_{jx} + a_{2x}p_{jy} + \sum_{i=1}^n w_{ix} U(\|\mathbf{p}_i - \mathbf{p}_j\|) = q_{jx}$$

$$\mathbf{w} = (a_{0x}, a_{1x}, a_{2x}, w_{1x}, \dots, w_{nx})$$

$$\begin{array}{l}
 n \text{ daných} \\
 \text{podmínek} \\
 \\
 3 \text{ další} \\
 \text{podmínky}
 \end{array}
 \left\{ \begin{array}{l}
 \left[\begin{array}{cccccc}
 1 & p_{1x} & p_{1y} & U_{1j} & \dots & U_{nj} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 1 & p_{nx} & p_{ny} & U_{nj} & \dots & U_{nn}
 \end{array} \right] \begin{bmatrix} a_{0x} \\ a_{1x} \\ a_{2x} \\ w_{1x} \\ \vdots \\ w_{nx} \end{bmatrix} \\
 \left[\begin{array}{cccccc}
 0 & 0 & 0 & 1 & \dots & 1 \\
 0 & 0 & 0 & p_{1x} & \dots & p_{nx} \\
 0 & 0 & 0 & p_{1y} & \dots & p_{ny}
 \end{array} \right] \begin{bmatrix} w_{1x} \\ \vdots \\ w_{nx} \end{bmatrix}
 \end{array} \right\} = \begin{bmatrix} q_{0x} \\ \vdots \\ q_{nx} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

TPS odvození

- Soustava rovnic

$$M\mathbf{w} = \mathbf{x}$$

$$M^{-1}M\mathbf{w} = M^{-1}\mathbf{x}$$

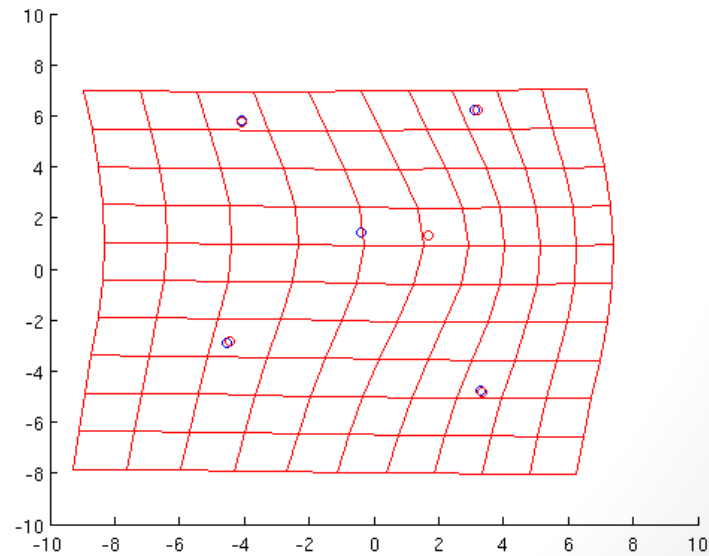
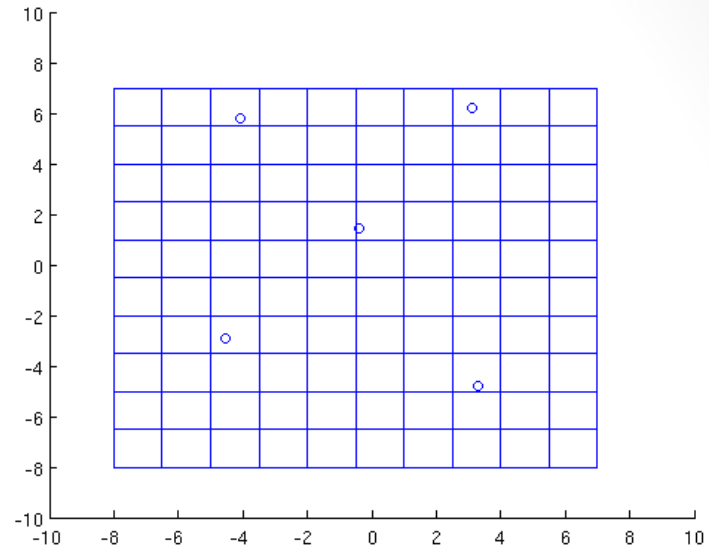
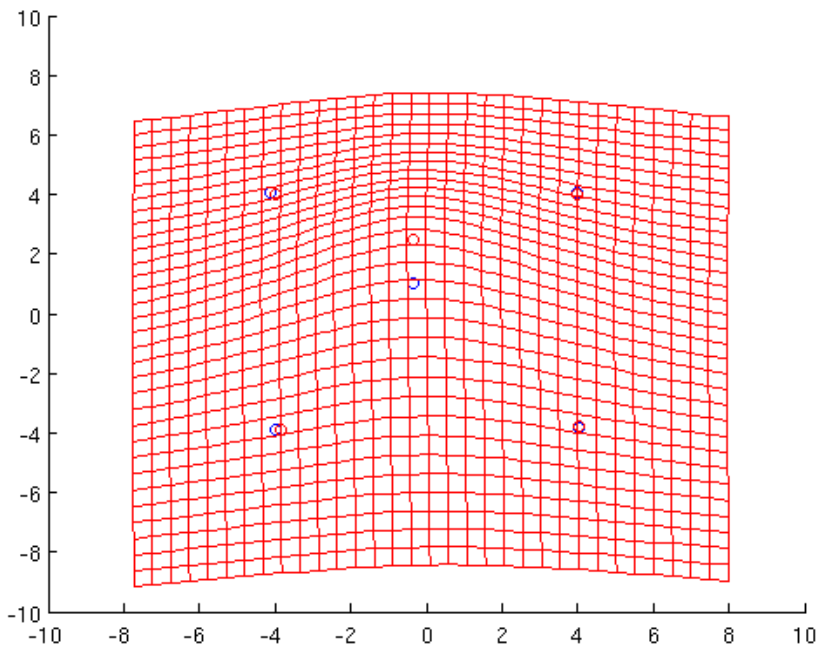
$$\mathbf{w} = M^{-1}\mathbf{x}$$

- Získám potřebné parametry \mathbf{w} pro transformační funkci
- Provedu vykreslení
- Levá dolní část M^{-1} velikosti $n \times n$ se nazývá matice ohybové energie (Bending energy matrix)

TPS ukázka

- Morphome3cs

TPS ukázka



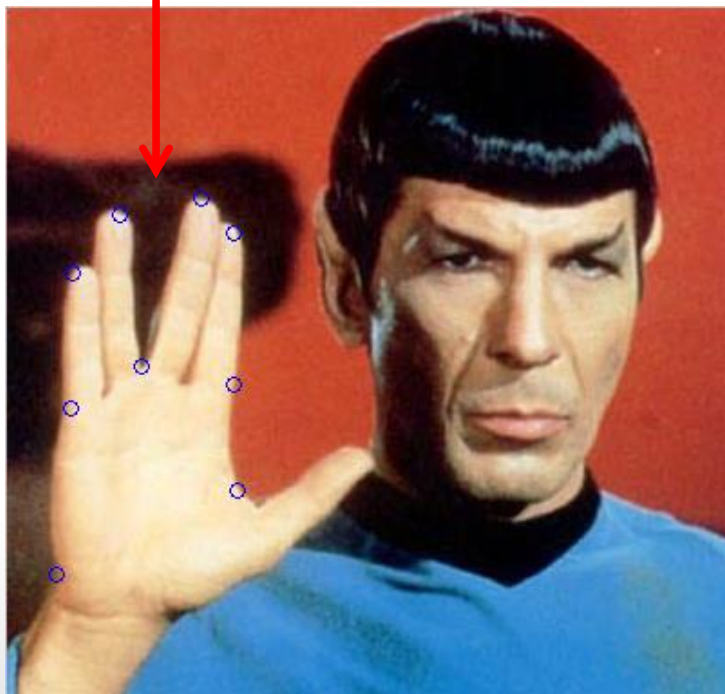
TPS - shrnutí

- První použití vizualizace **D'Arcy Thompson**, ruční kreslení
- Inspirace v matematice strojního inženýrství
 - Nekonečně tenký kovový plátek
- Interpolace **definovaná všude**
 - Může vést k falešným závěrům
- Důležité je husté pokrytí landmarky
- Vizualizace **rozdílů** jako deformací ve 2D
 - Lze snadno ve 3D, problém se zobrazováním
 - 2D řezy, 3D mřížky
- Landmarky se nesmějí „křížit“ – nestabilita

Ukázka 1

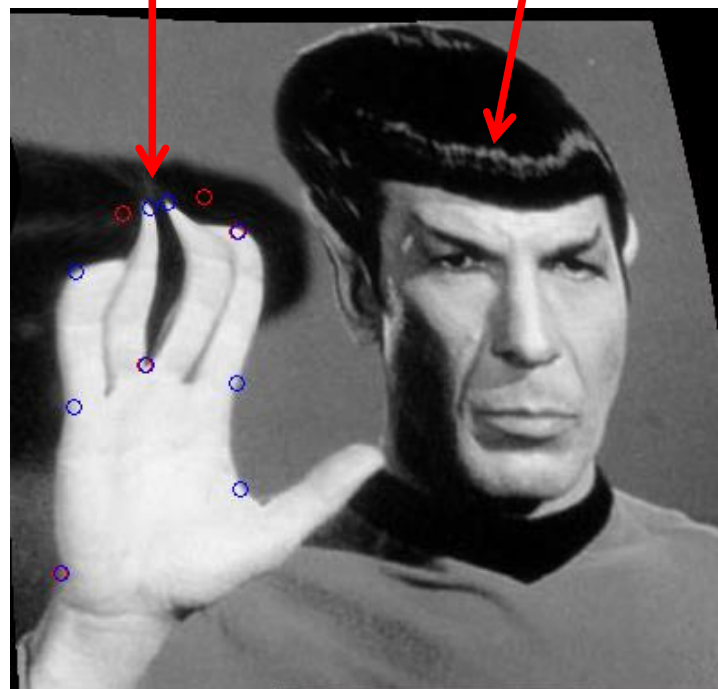
- Jiné použití TPS – rektifikace obrazu
 - Lokální deformace má globální efekt

Přitáhnout prsty k sobě

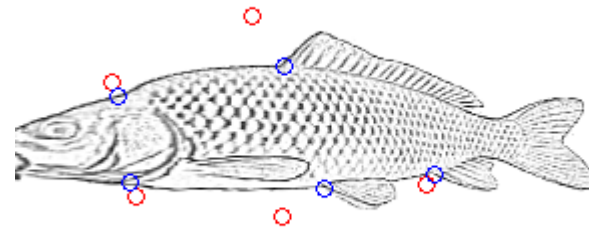
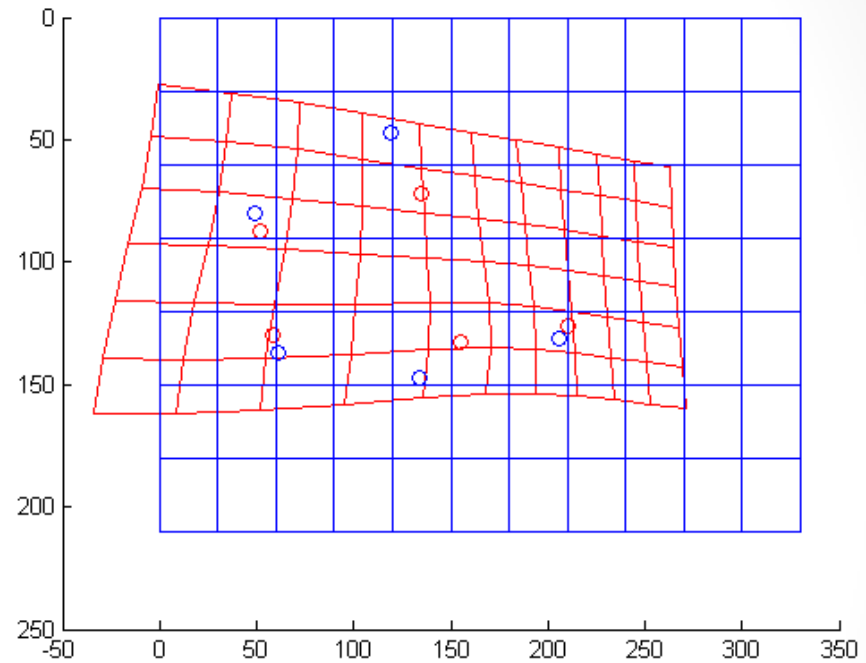
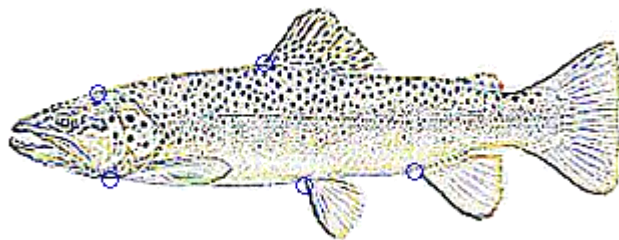
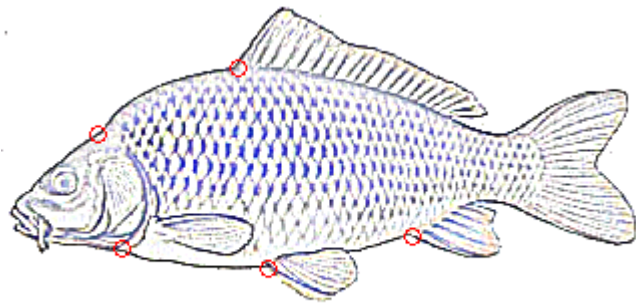


Vznikající nestabilita

Vzdálená deformace



Ukázka 2

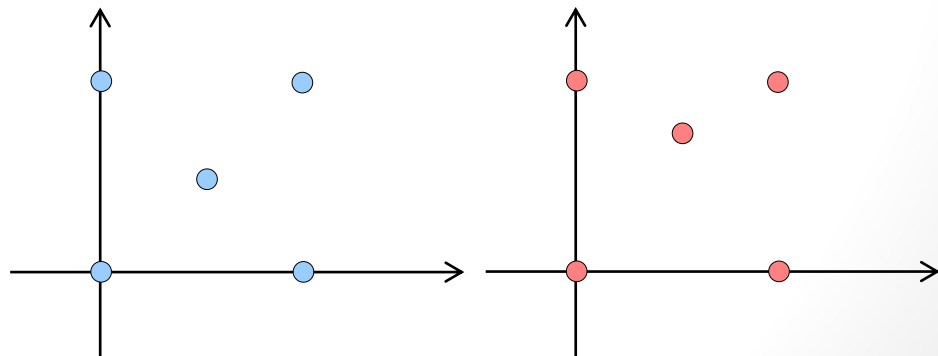


Warps

- Parametrický model tvaru odvozený z TPS
- Principal warps, Partial warps
 - Vlastní vektory bending energy matrix E
 - Setříděné nejvýraznější po dvou kolmé směry lokálních deformací
- Příklad

- $P = \begin{pmatrix} 0 & 0 & 1 & 1 & 0.5 \\ 0 & 1 & 1 & 0 & 0.5 \end{pmatrix}^T$

- $Q = \begin{pmatrix} 0 & 0 & 1 & 1 & 0.5 \\ 0 & 1 & 1 & 0 & 0.75 \end{pmatrix}^T$



Warps - příklad

- Na spočtení M^{-1} potřebuji pouze souřadnice vstupních landmarků, P

$$E = \begin{bmatrix} 0.4809 & -0.2404 & 0.4809 & -0.2404 & -0.4809 \\ -0.2404 & 0.4809 & -0.2404 & 0.4809 & -0.4809 \\ 0.4809 & -0.2404 & 0.4809 & -0.2404 & -0.4809 \\ -0.2404 & 0.4809 & -0.2404 & 0.4809 & -0.4809 \\ -0.4809 & -0.4809 & -0.4809 & -0.4809 & 1.9236 \end{bmatrix}$$

- Principal warps tvoří komponenty deformační složky bez ohledu na cílové landmarky

$$v_1 = [-0.5000 \quad 0.5000 \quad -0.5000 \quad 0.5000 \quad 0.0000]$$

$$v_2 = [0.2236 \quad 0.2236 \quad 0.2236 \quad 0.2236 \quad -0.8944]$$

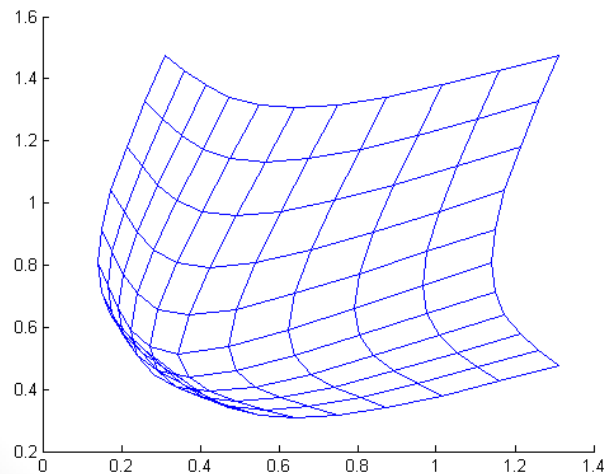
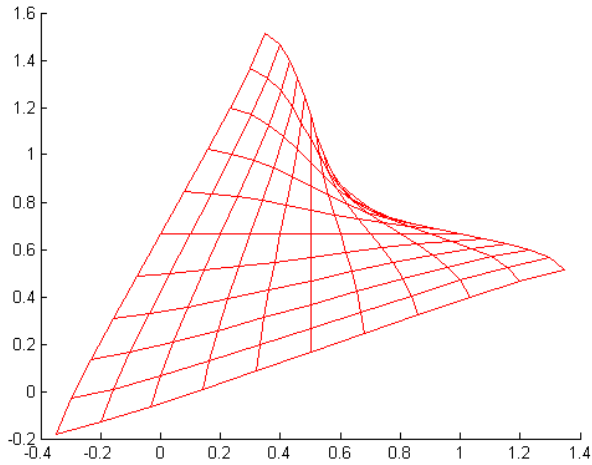
Warps - model

- Konkrétní exemplář získáme z:
 - **Afinní** transformace vstupních landmarků
 - Lineární **kombinace warpů**

$$Q = PA + \sum (v_1 k_1^T + v_2 k_2^T)_i U(\| \mathbf{p}_i - \mathbf{x} \|)$$

- Koeficienty k se spočítají ze znalosti vstupních P a výstupních Q landmarků – z TPS koeficientů
 - Nazývají se **partial warp scores** a tvoří tvarové proměnné
- Je možné vizualizovat vliv jednotlivých warpů na tvar odděleně (nastavením ostatních $k_i = 0$)

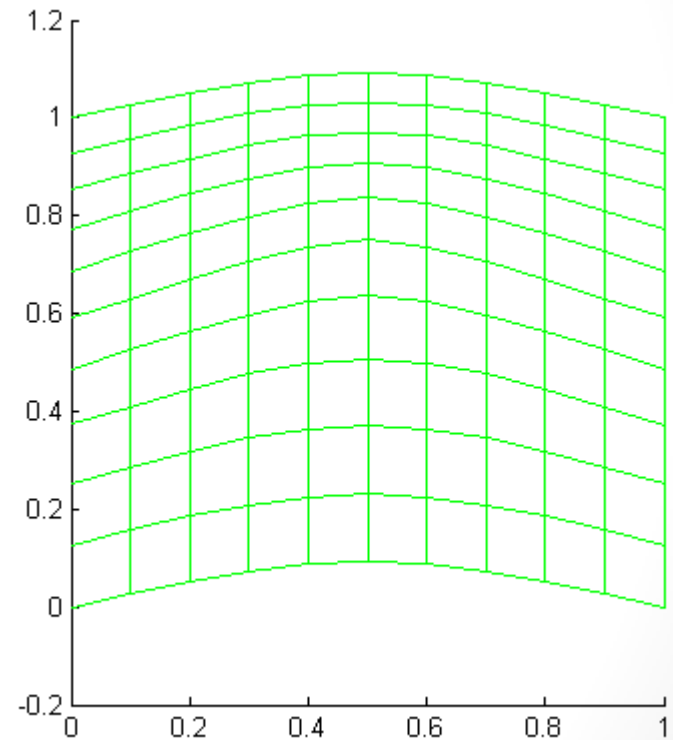
Warps - dekompozice



$$\cdot [k_{1x}, k_{1y}]$$

+

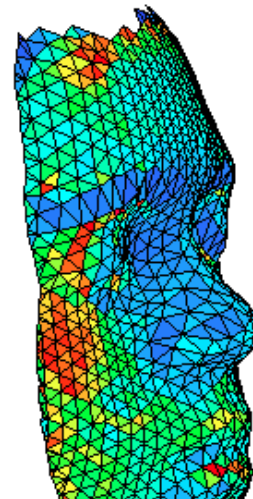
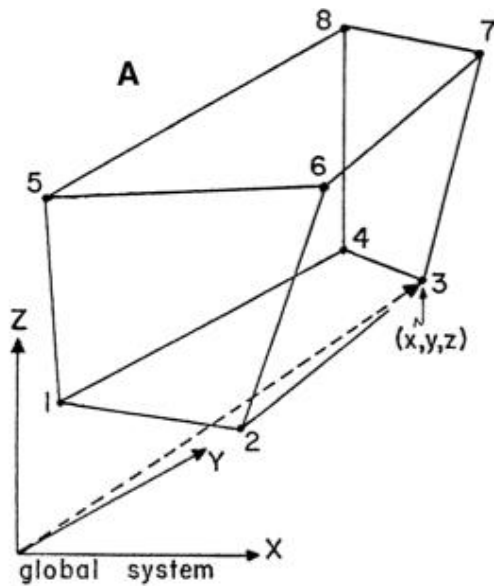
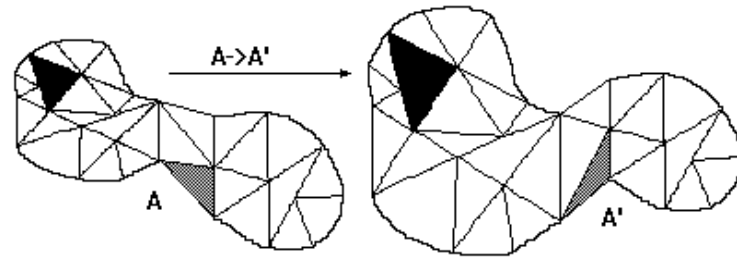
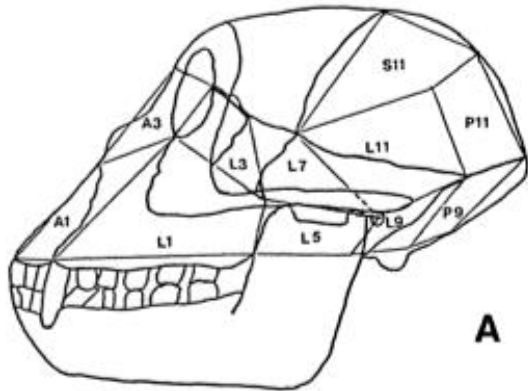
$$\cdot [k_{2x}, k_{2y}]$$



Analýza konečných prvků

- Spojitá deformace (TPS nebo jiná) nenabízí přímo žádnou míru (číslo) hodnotící deformaci nebo lokální změnu velikosti
- Motivace v mechanice kontinua
 - Popis oblasti mezi landmarky jako celku – konečné uzavřené prvky většinou trojúhelníkového tvaru
 - Přirazení hodnoty popisující rozdíly v **tvaru** nebo **velikosti**
- FESA – Finite Element Scaling Analysis

Konečné prvky



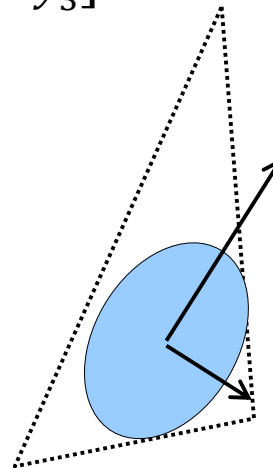
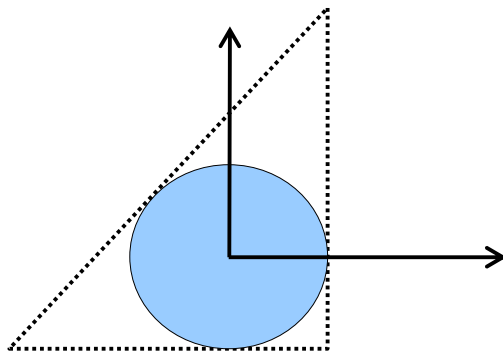
Mechanika kontinua

- Je možné dodefinovat spojitou transformační funkci a analyzovat ji prostředky mechaniky kontinua (Cheverud and Richtmeier 1986)
- Strain tensor
 - $F = (\epsilon_{i,j}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$
- Rozklad na dvojici tenzorů velikosti a deformace
 - $F = S + T$
- Redukce každého tensoru na jediný skalár (lokální rozdíl velikostí/tvarů)

FESA na trojúhelníkových sítích

- Velikost je dána poměrem ploch odpovídajících si trojúhelníků, nebo kružnice a elipsy
- Tvar je dán poměrem velikostí vlastních čísel deformační matice

$$P = QA = \begin{bmatrix} \bar{x}_1 & \bar{y}_1 \\ \bar{x}_2 & \bar{y}_2 \\ \bar{x}_3 & \bar{y}_3 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$



Lineární model tvaru

- Model – matematický popis jevu z reálného světa
 - Reálné příklady ho mohou více nebo méně splňovat
- Příklad
 - Model tvaru lebky může být jeden „vhodný“ reprezentant
 - Splňuje ho pouze on sám, extrémy jsou daleko
 - Může být průměrný reprezentant ze zkoumané populace
 - Nesplňuje ho nikdo, ale všichni jsou blízko
 - Průměr + tolerance
 - Může popisovat i nemožné exempláře
- **Parametrický model**
 - Každý exemplář odpovídá číselné konfiguraci

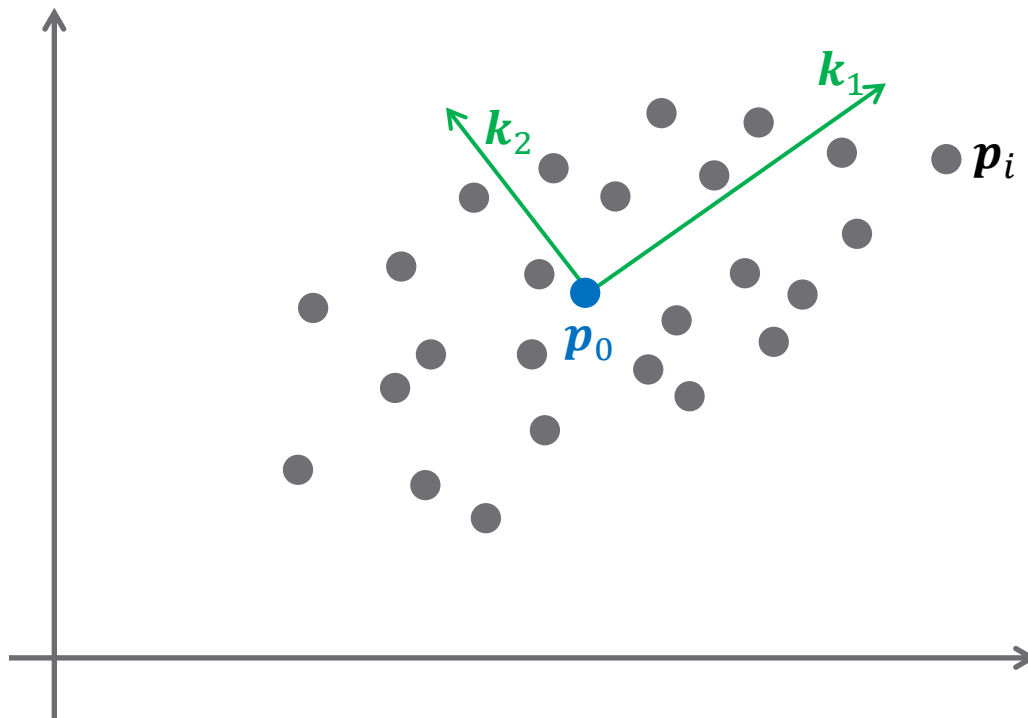
Lineární model tvaru

- Zohledňuje tendence vyskytující se v pozorované množině jedinců, vzorku
 - $\{P_1, P_2, \dots, P_n\} \rightarrow F(k_1, k_2, \dots, k_m)$
- Takový model dokáže popsat nekonečně velkou populaci – generování jedinců
- Matematický nástroj – analýza hlavních komponent
 - Principal component analysis (PCA)
 - Jeden exemplář se skládá z příspěvků různých komponent
 - Hledáme komponenty které přispívají nejvíc

PCA

- Data $\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{in})$
- Rozklad na $\mathbf{p}_0, \mathbf{k}_i$
 - Průměr \mathbf{p}_0
 - Hlavní komponenty (PC) \mathbf{k}_i
- Komponenty jsou seřazeny dle míry přispění (zachycené variability)
- Každého jedince lze zapsat jako součet průměru a lineární kombinace komponent
 - $\mathbf{p}_i = \mathbf{p}_0 + \sum_{j=1}^n v_j \mathbf{k}_j$
 - v_j skóre

PCA



PCA

- Jak najít množinu hlavních komponent?
 - Komplikovaná teorie, vynecháme (tl, dr)
 - Odpovídá vlastním vektorům kovariační matice
- Kovariance popisuje „závislost“ dvou náhodných proměnných
- Kovariační matice popisuje vzájemnou „závislost“ všech dvojic

$$c_{ij} = \frac{1}{N} \sum_k (P_{ki} - \bar{P}_i)(P_{kj} - \bar{P}_j)$$

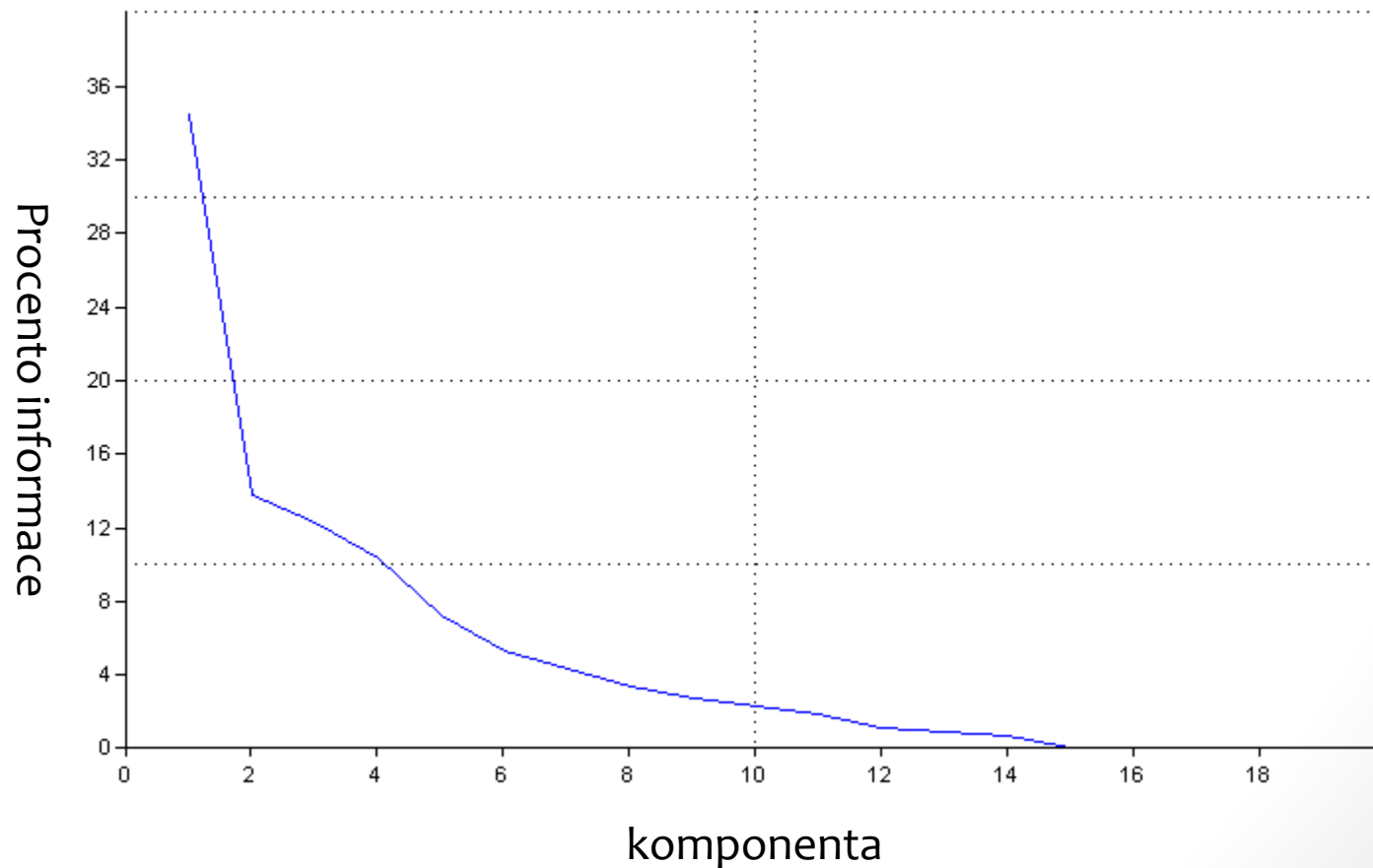
$$C = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{1n} & \cdots & c_{nn} \end{bmatrix}$$

PCA výpočet

- Rozklad kovariační matice na vlastní vektory
 - Dvojic vlastní vektor \mathbf{x} , číslo λ je stejně jako souřadnic
 - Vlastní číslo určuje důležitost komponenty
 - $C\mathbf{x} = \lambda\mathbf{x}$
- Můžu se rozhodnout kolik komponent chci zahrnout do svého modelu
 - Víc komponent – víc (potenciálně zbytečných) detailů, parametrů
 - Poměr součtu vlastních čísel vybraných komponent ku celkovému součtu odpovídá množství informace

PCA – váha komponent

- Nejvíc informace je typicky obsaženo v prvních několika komponentách – vzorek není náhodný



PCA – příklad

- Data

$$P = \begin{bmatrix} 1 & 2 & 1.5 & 4 \\ 1.1 & 1.2 & 2.5 & 3.8 \\ 1.1 & 1.2 & 1.5 & 3.5 \\ 1.3 & 1.1 & 1.2 & 3.7 \end{bmatrix}$$

- Kovariační matice

$$\text{cov}(P) = \begin{bmatrix} 0.016 & -0.039 & -0.026 & -0.012 \\ -0.039 & 0.18 & -0.031 & 0.068 \\ -0.026 & -0.031 & 0.32 & 0.022 \\ -0.012 & 0.068 & 0.022 & 0.043 \end{bmatrix}$$

- Vlastní vektory

$$\mathbf{v}_1 = [-0.8117 \quad -0.3713 \quad -0.1295 \quad 0.4318]$$

$$\mathbf{v}_2 = [0.5388 \quad -0.2217 \quad -0.0343 \quad 0.8120]$$

$$\mathbf{v}_3 = [0.2167 \quad -0.8866 \quad -0.1194 \quad -0.3909]$$

$$\mathbf{v}_4 = [0.0618 \quad 0.1642 \quad -0.9838 \quad -0.0377]$$

- Model

$$\lambda_1 = 0$$

$$\lambda_2 = 0.016$$

$$\lambda_3 = 0.2114$$

$$\lambda_4 = 0.3301$$

0% informace

2.87% informace

37.92% informace

59.21% informace

- Průměr

$$\mathbf{P}_0 = [1.1250 \quad 1.3750 \quad 1.6750 \quad 3.7500]$$

PCA – příklad

- Zbývá dopočítat koeficienty (souřadnice, skóre) pro namodelování původních dat

$$\mathbf{k}_1 = [0.4158 \quad 0.1040 \quad -0.2303 \quad 0.5125]$$

$$P_0 = [1.1250 \quad 1.3750 \quad 1.6750 \quad 3.7500]$$

$$P_0 + \mathbf{v}_4 k_{1,4} = [1.3463 \quad 1.7911 \quad 1.4746 \quad 3.7307]$$

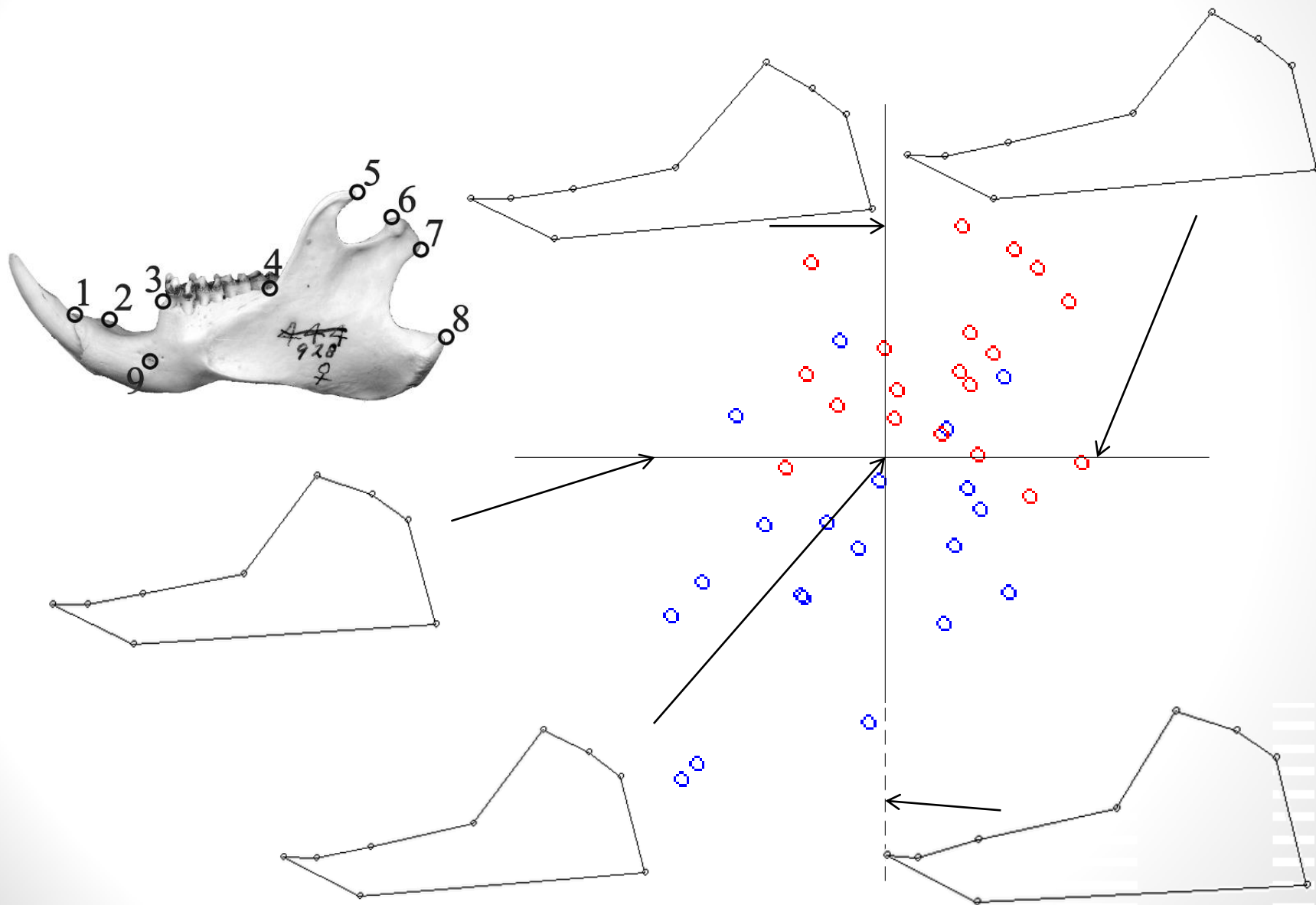
$$P_0 + \mathbf{v}_3 k_{1,3} + \mathbf{v}_4 k_{1,4} = [1.3761 \quad 1.7990 \quad 1.5021 \quad 3.9572]$$

$$P_0 + \mathbf{v}_2 k_{1,2} + \mathbf{v}_3 k_{1,3} + \mathbf{v}_4 k_{1,4} = [1.3375 \quad 1.7760 \quad 1.4099 \quad 3.9743]$$

$$P_0 + \mathbf{v}_1 k_{1,1} + \mathbf{v}_2 k_{1,2} + \mathbf{v}_3 k_{1,3} + \mathbf{v}_4 k_{1,4} = [1.0000 \quad 2.0000 \quad 1.5000 \quad 4.0000] = P_1$$

- Zobrazení vybraných 2 až 3 koeficientů do grafu pro všechny exempláře (scatter plot)

PCA – demonstrace



PCA aplikace – Eigenfaces (1)

- Yale faces dataset
 - Registrované fotografie lidských tváří
 - 2D data převedeny na 1D „sploštěním“



PCA aplikace – Eigenfaces (2)

- PCA na vektorech tváří produkuje hlavní komponenty – „eigenfaces“
 - Každou tvář je možné vyjádřit jako součet průměrné tváře a lineární kombinace „eigenfaců“



PCA aplikace – Eigenfaces (3)

- Kolik hlavních komponent je potřeba sečíst?
 - Ideálně všechny (někdy není možné)
 - Čím více, tím přesnější aproximace
 - Nejvýraznější příspěvky od komponent s nízkým indexem



PCA aplikace – Eigenfaces (4)

- Do báze určené PC je možné převést data \mathbf{q} které nebyly zahrnuty do analýzy
 - Takové data nemusí být přesně rekonstruovatelné
 - $\mathbf{v}_j = (\mathbf{q} - \mathbf{p}_0) \cdot \mathbf{k}_j$



PCA – závěr

- PCA modeluje vztahy mezi landmarky **pouze lineárně**
 - Existují metody které dokáží zachytit nelineární vztahy
- Další užitečné použití
 - **Dobře separuje třídy** – odlišnost se projevuje v prvních komponentách (shluková analýza)
 - **Redukce dimenze** – vytvoří stejný počet „nových“ proměnných, ale poslední nesou jen minimum informace, ty je možné zanedbat
 - **Dopočítání chybějících dat**