Detekcija anomalija korištenjem difuzijskih preslikavanja

Doris Đivanović i Karlo Gjogolović

PMF - MO ZG

14. svibnja 2025.

Sadržaj

Uvod - difuzijska preslikavanja

- Analiza podataka - podaci velikih dimenzija $\Longrightarrow$ nepraktično
- $\Longrightarrow$ reprezentirati ih podacima iz prostora puno manje dimenzije
- $\Longrightarrow$ **preslikavanje** u prostor što je moguće manje dimenzije

- Problemi klasteriranja - poseban slučaj: problem **detekcije anomalija** u podacima
- $\Longrightarrow$ ključni koncepti: povezanost, sličnost, bliskost podataka
- $\Longrightarrow$ želimo preslikavanje koje čuva te odnose
- $\Longrightarrow$ tj. takvo da je:

difuzijska udaljenost orig. podataka $=$ euklidska udaljenost njihovih slika

$$\boxed{D(x,y) \quad = \quad ||\Psi(x) - \Psi(y)||_2}$$

- $\Longrightarrow$ tzv. **difuzijsko preslikavanje**

Uvod - difuzijska preslikavanja

- $\Gamma = \{x_1, ..., x_n\}$ skup podataka iz prostora velike dimenzije

- Kako definirati **difuzijsku udaljenost**?

- Kako definirati **difuzijsko preslikavanje**?

Konstrukcija težinskog grafa

- $\Gamma = \{x_1, ..., x_n\}$ skup podataka iz prostora velike dimenzije
- $\Longrightarrow$ težinski graf:
  - čvorovi $=$ podaci x_i
  - težina brida $w(x_i, x_j) =$ **mjera sličnosti (afiniteta)** između x_i i x_j
- $\Longrightarrow$ za težinsku funkciju w želimo da je:
  - simetrična:
  $$w(x_i, x_j) = w(x_j, x_i)$$
  - nenegativna:
  $$w(x_i, x_j) \geq 0$$
  - (često) normalizirana:
  $$w(x_i, x_j) \in [0, 1], \;\; w(x_i, x_i) = 1.$$
- **Kako definirati w?**

Funkcija težine

- Prirodno je mjeru afiniteta povezati s euklidskom udaljenošću između podataka i to:

$$\text{manja udaljenost} \quad \Longleftrightarrow \quad \text{veća sličnost}$$

- $\Longrightarrow$

$$w(x_i, x_j) = \frac{1}{f(||x_i - x_j||_2)}, \qquad f > 0$$

  je simetrična i nenegativna
- $\Longrightarrow$ uzeti da eksponencijalno ovisi o euklidskoj udaljenosti
- $\Longrightarrow$

$$w(x_i, x_j) = \frac{1}{e^{||x_i - x_j||_2^2}} = e^{-||x_i - x_j||_2^2}$$

- $\Longrightarrow$ očito je i:

$$w(x_i, x_j) \in [0, 1], \;\; w(x_i, x_i) = 1.$$

Funkcija težine: Gaussova jezgra

- Bilo bi još dobro težinsku funkciju skalirati nekim faktorom $\sigma > 0$
- varijabilni parametar $\Longrightarrow$ promatrati njegov utjecaj s obzirom na njegove različite vrijednosti
- $\Longrightarrow$

$$w(x_i, x_j) = e^{-\dfrac{\|x_i - x_j\|_2^2}{\sigma}},$$

- odnosno, ako želimo označiti da oba podatka doprinose sa $\sigma > 0$:

$$\boxed{w(x_i, x_j) = e^{-\dfrac{\|x_i - x_j\|_2^2}{\sigma^2}}}$$

- $\Longrightarrow$ tzv. **Gaussova (gaussovska) jezgra**
- **Kako odabrati σ?**

Funkcija težine: Gaussova jezgra

$$\boxed{w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma^2}}} \qquad \sigma > 0$$

- Vrijednost w proporcionalna vrijednosti σ
- σ jako mali $\Longrightarrow$ većina težina u grafu jako mala, tj. graf jako nepovezan $\Longrightarrow$ podaci bliski samo sebi
- σ jako velik $\Longrightarrow$ većina težina u grafu jako velika $\Longrightarrow$ gotovo svi podaci međusobno povezani, jako bliski
- $\Longrightarrow$ **OVO JE LOŠE ZA PRIMJENE!**

- Parametar $\sigma > 0$ očito kontrolira (povećava ili umanjuje) utjecaj euklidske udaljenosti dvaju podataka na mjeru njihove sličnosti
- $\Longrightarrow$ odrediti ga empirijski tako da umanjuje ili uvećava onda kada je prikladno

Funkcija težine: Gaussova jezgra

$$\boxed{w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma^2}}} \qquad \sigma > 0$$

Još jedan problem:

- ovakva težinska funkcija ovisi isključivo o međusobnoj udaljenosti odgovarajućih dvaju podataka, nikako o ostatku skupa podataka

Funkcija težine: Gaussova jezgra

$$\boxed{w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma^2}}} \qquad \sigma > 0$$

- 1. rješenje: "usrednjenje"

  Npr. $\sigma =$
  - medijan
  - standardna devijacija
  - ...

  svih međusobnih udaljenosti u grafu.

- **Prednost**: "globalan" σ ...
- **Mana**: ...

Funkcija težine: Gaussova jezgra

- 2. rješenje: skalirati težinu između dva podatka nekim faktorom koji ovisi o bas tim konkretnim podacima
- Želimo "difuzijsku udaljenost" $\Longrightarrow$ želimo da mjera sličnosti oslikava lokalnu geometriju
- $\Longrightarrow$ definirati preslikavanje koje svakom podatku pridružuje faktor koji ovisi samo o njegovoj maloj okolini, a ne cijelom skupu podataka:

$$x_i \longrightarrow \sigma_i$$

- $\Longrightarrow$ svaki od dva podatka doprinosi međusobnoj težini svojim faktorom:

$$\boxed{w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma_i \sigma_j}}} \qquad \sigma_i, \sigma_j > 0$$

Gaussova jezgra za detekciju anomalije

- Primjer - problem **detekcije anomalije**
- Pretpostavljamo:
- Podaci koji nisu anomalija, odnosno u tom smislu "normalni" podaci - nalaze se u "gusto naseljenom" području, tj. nalaze se blizu većine ostalih podataka iz skupa podataka
- Podaci koji jesu anomalija - nalaze se u "rijetko naseljenom" području, odnosno daleko od gotovo svih ostalih podataka iz skupa podataka

Gaussova jezgra za detekciju anomalije

- Primjer - posebno prikladan za problem **detekcije anomalije**:

$$w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma_i \sigma_j}}$$

$$\boxed{\sigma_i = \|x_i - x_K\|_2^2}$$

- x_K - K-ti najbliži susjed podatka x_i

- Moguća interpretacija ...

- Težinska funkcija $w : \Gamma \times \Gamma \to \mathbb{R}^+$, $\Gamma = \{x_q, \ldots, x_n\}$, definira kvadratnu simetričnu **matricu sličnosti** W dimenzije n s jedinicama na dijagonali

$$w_{ij} = w(x_i, x_j)$$

Slučajna šetnja i matrica prijelaza

Na temelju težina $w(x_i, x_j)$ definira se slučajna šetnja na skupu podataka Γ.

- Normalizacijom težinske funkcije w definiramo novu funkciju:

$$p(x_i, x_j) = \frac{w(x_i, x_j)}{d(x_i)}, \quad \text{gdje je } d(x_i) = \sum_{x_j \in \Gamma} w(x_i, x_j).$$

- $\Longrightarrow$ očito vrijedi:

$$\sum_{x_j \in \Gamma} p(x_i, x_j) = 1, \qquad p(x_i, x_j) \geq 0$$

- $\Longrightarrow$ vrijednost $p(x_i, x_j)$ možemo shvatiti kao vjerojatnost prijelaza iz podatka x_i u podatak x_j u jednom koraku
- $\Longrightarrow$ matrica P s elementima $p_{ij} = p(x_i, x_j)$ je retčano stohastička matrica prijelaza Markovljeva lanca
- $\Longrightarrow$ matrica P^t je matrica prijelaza u t koraka, ima elemente $p_{ij}^t = p_t(x_i, x_j)$ - predstavljaju vjerojatnosti prijelaza iz x_i u x_j u t koraka

Slučajna šetnja i matrica prijelaza

- Znamo da retčano stohastička matrica P ima potpun skup svojstvenih vrijednosti, od kojih je jedna i najveća $\lambda = 1$, čiji je svojstveni potprostor skup svih višekratnika jediničnog vektora.
- Spektar matrice P brzo opada, dakle vrijedi:

$$1 = |\lambda_0| \geq |\lambda_1| \geq \ldots$$

- Ako s D označimo dijagonalnu matricu čiji su dijagonalni elementi

$$d_{ii} = d(x_i) = \sum_{x_j \in \Gamma} w(x_i, x_j), \quad \forall\ i \in \{1, \ldots, n\},$$

matricu P očito možemo prikazati kao:

$$P = D^{-1} W$$

Spektar matrice prijelaza i difuzijsko preslikavanje

- Matrica $P = D^{-1}W$ je slična simetričnoj matrici
$$P_s = D^{-1/2}WD^{-1/2},$$
gdje je
$$D^{-1/2}D^{-1/2} = D^{-1},$$
pa te matrice imaju iste svojstvene vrijednosti, a svojstveni vektori im se razlikuju za faktor $D^{-1/2}$.
- Kako je matrica P_s simetrična, ima ortonormirani skup lijevih svojstvenih vektora
- Možemo ga dobiti spektralnom dekompozicijom matrice P_s:
$$P_s = U\Lambda U^T.$$
- Za ortonormirani skup lijevih i desnih svojstvenih vektora, Markovljev lanac može se predstaviti ovako:
$$p_t(x_i, x_j) = \sum_{l \geq 0} \lambda_l^t u_l(x_i) v_l(x_j)$$

Spektar matrice prijelaza i difuzijsko preslikavanje

- $\Longrightarrow$ za ortonormirani skup lijevih svojstvenih vektora, očito bismo svakom podatku x_i mogli pridružiti sljedeće:

$$x_i \longrightarrow \left(\lambda_0 u_0(x_i), \lambda_1 u_1(x_i), \lambda_2 u_2(x_i), \ldots, \lambda_n u_n(x_i)\right)^T .$$

- Kako je u_0 konstantan, možemo ga izostaviti
- Zbog brzog opadanja spektra svojstvenih vrijednosti (λ_l brzo teže k nuli), za određenu točnost gornje sume dovoljno je uzeti određeni broj vodećih svojstvenih parova, neki $l < n$
- $\Longrightarrow$ svakom podatku x_i mogli bismo pridružiti sljedeće:

$$x_i \in \Gamma \longrightarrow \left(\lambda_1 u_1(x_i), \lambda_2 u_2(x_i), \ldots, \lambda_\ell u_\ell(x_i)\right)^T \in \mathbb{R}^\ell$$

odnosno:

$$x_i \in \Gamma \longrightarrow \left(\lambda_1^t u_1(x_i), \lambda_2^t u_2(x_i), \ldots, \lambda_\ell^t u_\ell(x_i)\right)^T \in \mathbb{R}^\ell$$

za odabrani broj koraka t.

- $\Longrightarrow$ ovo je dobar **kandidat za difuzijsko preslikavanje**!

Difuzijska udaljenost

Difuzijska udaljenost u prostoru Γ može se definirati ovako:

$$D_t^2(x_i, x_j) = \sum_{x_j \in \Gamma} \frac{(p_t(x_i, x_j) - p_t(x_i, x_j))^2}{u_0(x_j)}$$

Zašto je ovo dobro uzeti kao difuzijsku udaljenost, dakle za **mjeru bliskosti** u prostoru Γ?

- Mjeri sličnost distribucija vjerojatnosti podataka x_i i x_j nakon t koraka
- Intuitivno: udaljenost između x_i i x_j je mala, tj. bliskost je velika, ako postoji puno kratkih puteva koji povezuju x_i i x_j
- Robusna na šum jer ovisi o svim putevima duljine t.
- Ovisi o cijelom skupu podataka, za razliku od euklidske

Difuzijsko preslikavanje i difuzijska udaljenost

- Izračun pomoću svojstvenih vektora:

$$D_t^2(x_i, x_j) = \sum_{x_j \in \Gamma} \frac{(p_t(x_i, x_j) - p_t(x_i, x_j))^2}{u_0(x_j)} = \sum_{j \geq 1} \lambda_j^{2t} (u_j(x_i) - u_j(x_j))^2$$

- $\Longrightarrow$ aproksimacija koristeći prvih ℓ svojstvenih vektora je računski efikasna
- Ako **difuzijsko preslikavanje** definiramo ovako:

$$\Psi_t : x_i \in \Gamma \longrightarrow \left(\lambda_1^t u_1(x_i), \lambda_2^t u_2(x_i), ..., \lambda_\ell^t u_\ell(x_i)\right)^T \in \mathbb{R}^\ell$$

  ovakva **difuzijska udaljenost** uistinu je jednaka Euklidskoj udaljenosti u prostoru difuzijskog preslikavanja:

$$D_t^2(x_i, x_j) = \|\Psi_t(x_i) - \Psi_t(x_j)\|^2$$

Proširenje funkcija izvan uzorka

- Za velike skupove podataka (npr. slike), računanje difuzijskog preslikavanja za sve podatke je nepraktično
- $\Longrightarrow$ izračunaj preslikavanje za podskup uzoraka $\Gamma' \subseteq \Gamma$, a zatim proširi ulaganje na sve podatke u Γ.
- Poznate metode: Nyströmova ekstenzija, geometrijske harmonike.
- **Višerezolucijska Laplaceova piramida**:
- Aproksimacija funkcije na svakoj razini piramide l pomoću Gaussove jezgre

$$W_l(x_i, x_j) = \exp(-\|x_i - x_j\|^2/\sigma_l)$$

  s progresivno manjim σ_l.
- Normalizirana jezgra:

$$K_l = q_l^{-1} W_l, \qquad q_l(x_i) = \sum_j W_l(x_i, x_j).$$

Višerezolucijska Laplaceova piramida

- Laplaceova piramidalna reprezentacija funkcije f definirana je iterativno komponentama s_l:

$$s_0(x_k) = \sum_{i=1}^{n_{subset}} K_0(x_i, x_k) f(x_i)$$

$$s_l(x_k) = \sum_{i=1}^{n_{subset}} K_l(x_i, x_k) d_l(x_i), \quad l \geq 1$$

$$\text{gdje je } d_l(x_k) = f(x_k) - \sum_{m=0}^{l-1} s_m(x_k), \quad l \geq 1$$

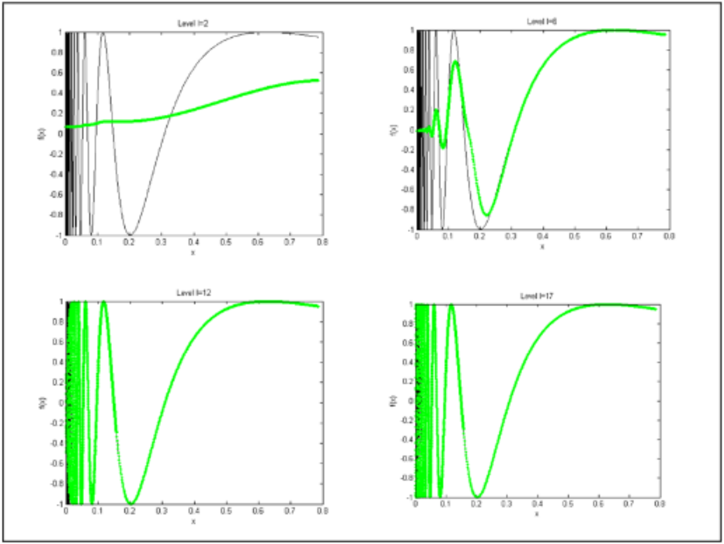
- Iterira se dok norma reziduala ne padne ispod zadanog praga / tolerancije
- Proširenje funkcije f na novu točku $\bar{x}_k \in \Gamma \setminus \Gamma'$:

$$f(\bar{x}_k) = \sum_{l \geq 0} s_l(\bar{x}_k)$$

gdje se $s_l(\bar{x}_k)$ računaju analogno gornjim formulama, ali s $\bar{x}_k$ umjesto x_k, tj. $K_l(x_i, \bar{x}_k)$ umjesto $K_l(x_i, x_k)$

Višerezolucijska Laplaceova piramida

- Primjer: $f(x) = \sin\left(\frac{1}{x+0.1}\right)$



Višerezolucijska Laplaceova piramida

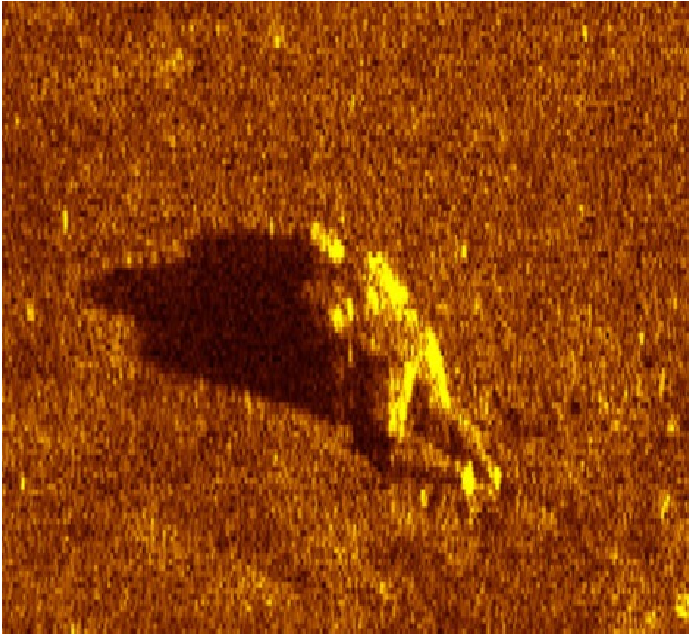
- Provodi se za svaku difuzijsku koordinatu Ψ_j, tj. komponentnu funkciju preslikavanja Ψ, zasebno.

Primjena: Detekcija anomalija na digitalnoj slici

- Cilj: Detektirati dijelove digitalne slike koji odskaču od ostatka (tzv. pozadine), tj. anomalije
- Problem se svodi na klasteriranje visokodimenzionalnih podataka (u dva klastera)
- Riješit ćemo ga ulaganjem podataka u prostor puno manje dimenzije

- Ulazni podaci: digitalna slika (npr. 'grayscale') - podatak tipa *Image*
- $\Longrightarrow$ MATLAB: matrica piksela kao cjelobrojnih vrijednosti intenziteta sive boje $\in [0, 255]$, tj. decimalnih vrijednosti $\in [0, 1]$ nakon normalizacije

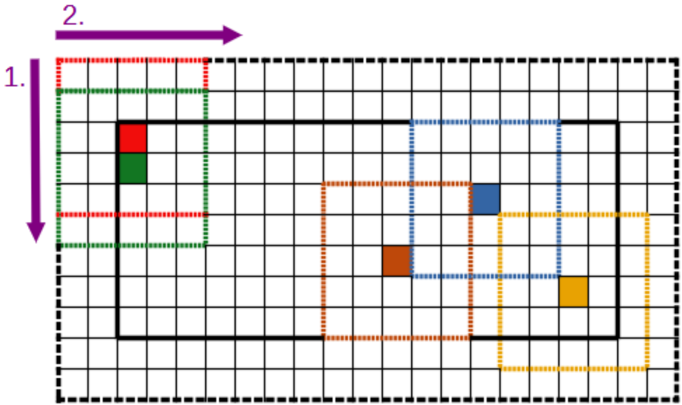
Primjena: Detekcija anomalija na digitalnoj slici



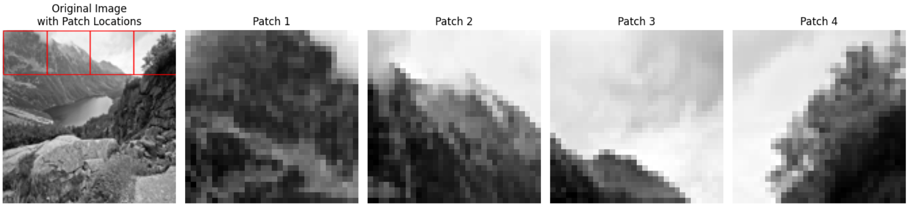
Podaci velike dimenzije - *patchevi* digitalne slike

- Cilj: za svaki piksel slike odrediti pripada li anomaliji
- Difuzijsko preeslikavanje ne računamo na pikselima, nego na skupu kvadratnih podmatrica odabrane dimenzije - *patchevi*
- $\Longrightarrow$ parametar **patchDim**
- $\Longrightarrow$ svakom pikselu pridružujemo različiti *patch* unutar kojeg se nalazi
- $\Longrightarrow$ uzimati patcheve npr. t.d. da im je element centar ili gornji lijevi vrh
- Može se napraviti *padding* matrice kako bi se mogli izdvojiti *patchevi* centrirani i na rubnim elementima (npr. MATLAB funckija **padarray(...)** s opcijom 'symmetric').
- $\Longrightarrow$ **vrijednost dif. preslikavanja za piksel ne ovisi samo o tom pikselu, nego o cijeloj njegovoj kvadratnoj okolini !**

Podaci velike dimenzije - *patchevi* digitalne slike



Podaci velike dimenzije - *patchevi* digitalne slike



Podaci velike dimenzije - *patchevi* digitalne slike

- *patchevi* $\longrightarrow$ vektorizacija
- $\Longrightarrow$ podaci su vektori u prostoru $\mathbb{R}^m$, gdje je $m = \text{patchDim}^2$.
- **dimenziju** m smatramo **velikom**

- Za svaki patch, tj. odgovarajuće podatke, pamte se indeksi retka i stupca njegovog centra (ili gornjeg lijevog kuta) na originalnoj slici
- dif. preslikavanje - MATLAB funkcije **knnsearch(...)**, **pdist2(...)** $\Longrightarrow$ *patcheve*-vektore spremamo kao retke jedne matrice

Računanje dif. preslikavanja - MATLAB implementacija

$\Longrightarrow$ *patcheve*-vektore spremamo kao retke jedne matrice

Podskup skupa podataka

- računska složenost $\Longrightarrow$ dif. preslikavanje računamo na nasumično odabranom podskupu određene veličine
- MATLAB funkcije: **round()**, **randperm(...)**

- $\Longrightarrow$ potencijalni problemi premalenog ili nerelevantog uzorka

Računanje dif. preslikavanja - MATLAB implementacija

Matrica sličnosti W za podskup

$$w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|_2^2}{\sigma_i \sigma_j}}$$

$$\boxed{\sigma_i = ||x_i - x_K||_2^2}$$

- **rijetka matrica** $\Longrightarrow$ za svaki podatak x_i pronalazi se kNN **najbližih susjeda**
- $\Longrightarrow$ parametri K i kNN
- težine se računaju samo za te parove
- zadovoljava se uvjet simetričnosti težina
- ostale težine su 0
- $\Longrightarrow$ računati i samog sebe uz kNN najbližih susjeda, jer ne želimo da je sličnost sa samim sobom 0, nego 1

Računanje dif. preslikavanja - MATLAB implementacija

Matrica sličnosti W za podskup

- MATLAB funkcija **knnsearch(...)**
- MATLAB funkcija **sparse(...)**
- $\Longrightarrow$ za konstrukciju rijetke matrice dovoljno je konstruirati:
  - vektor indeksa redaka
  - vektor indeksa stupaca
  - vektor pripadnih vrijednosti

  za sve elemente koji nisu $= 0$
- $\Longrightarrow$ spremiti i rijetku matricu (kvadrata) međusobnih udaljenosti podataka u podskupu
- da bismo je mogli iskoristiti kasnije u Laplacevoj piramidi

Računanje dif. preslikavanja - MATLAB implementacija

Difuzijsko preslikavanje

- Računanje matrice $D^{-1/2}$
- $\Longrightarrow D^{-1/2} * D^{-1/2} = D^{-1}$
- $\Longrightarrow$ dijagonalna matrica s $\dfrac{1}{\sqrt{d(i,i)}}$ na dijagonali
- želimo rijetku matricu (sparse)
- $\Longrightarrow$ MATLAB funkcija **spdiags(...)**

- Simetrična matrica $P_s = D^{-1/2} * W * D^{-1/2}$

Računanje dif. preslikavanja - MATLAB implementacija

Difuzijsko preslikavanje

- Spektralna dekompozicija P_s: funkcijom **eigs(...)** dobiva se željeni broj vodećih svojstvenih parova (željena niska dimenzija postavljena kao parametar **numEigs** $\Longrightarrow$ matrice U i Λ
- odgovarajući skup svojstvenih vektora matrice P dobivamo kao

$$\Phi = D^{-1/2} U$$

- Koordinate difuzijskog preslikavanja - egzaktne za podskup: skaliranjem svojstvenih vektora matrice P odgovarajućim svojstvenim vrijednostima
- Proširenje na sve originalne podatke (*patcheve*): izravno pomoću gore opisane Laplaceove piramide, ali također s **rijetkim matricama težina** koristeći kNN **najbližih susjeda**, i unaprijed izračunate potrebne euklidske udaljenosti

Primjena difuzijskog preslikavanja

Kako koristimo dobiveno difuzijsko preslikavanje za detekciju anomalije među podacima?

- Ako je piksel sličan drugim pikselima - nije anomalija
- Ako je piksel "daleko" od ostalih - anomalija je

- Računa se sličnost $\bar{w}(i,j)$ između x_i i x_j, za sve piksele x_j u kvadratnom "prozoru" slike oko x_i, koristeći **difuzijsku udaljenost** i Gaussovu jezgru s globalnim faktorom $\bar{\sigma}$:

$$\bar{w}(i,j) = e^{-\frac{\|\Psi(x_i) - \Psi(x_j)\|^2}{\bar{\sigma}}}$$

Primjena difuzijskog preslikavanja

- Računa se afinitet $\bar{w}(i,j)$ između x_i i x_j, za sve x_j u kvadratnom "prozoru" slike oko x_i, koristeći **difuzijsku udaljenost** i Gaussovu jezgru s globalnim faktorom $\bar{\sigma}$:

$$\bar{w}(i,j) = e^{-\frac{\|\Psi(x_i) - \Psi(x_j)\|^2}{\bar{\sigma}}}$$

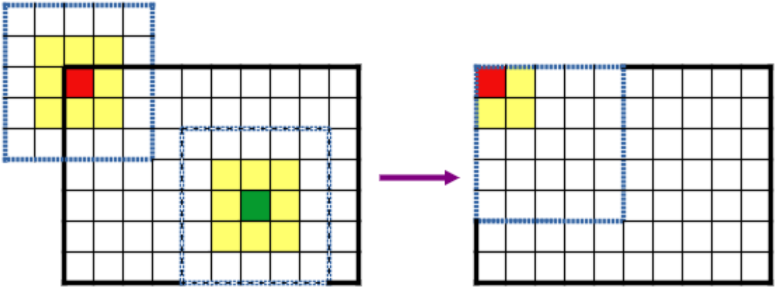
- Globalni faktor $\bar{\sigma}$ se određuje empirijski, npr.

$$\bar{\sigma} = r \cdot \sigma^2_{pair},$$

- σ^2_{pair} varijanca difuzijskih udaljenosti nasumično odabranih n_{pair} parova podataka
- faktor skaliranja udaljenosti (npr. $r = 20$).

- **Maskiranje**: unutarnji dio prozora N_i se često isključuje da se izbjegne usporedba s neposrednim susjedima (anomalija je veća od jednog piksela)
- Veličina prozora **winDim** i maske **maskDim** ovise o očekivanoj veličini anomalije

Primjena difuzijskog preslikavanja



Stupanj anomalije (Anomaly Score)

- N_i skup m susjeda u prozoru oko x_i
- $\Longrightarrow$ sličnost s okolinom:

$$\frac{1}{m}\sum_{j\in N_i}\bar{w}(i,j)$$

- $\Longrightarrow$ **stupanj anomalije** (*anomaly score*) za piksel x_i

$$C(i) = 1 - \frac{1}{m}\sum_{j\in N_i}\bar{w}(i,j)$$

- $\Longrightarrow$ **mala sličnost** s okolinom $\Longrightarrow$ **visok** anomaly score (blizu 1) $\Longrightarrow$ **potencijalna anomalija**

"Jednorezolucijski pristup" u analizi digitalnih slika

- Problemi ...

Višerezolucijski pristup - Generalna ideja

Umjesto primjene algoritma na samo jednoj rezoluciji slike, koristi se **Gaussova piramida** slike: $\{G_l\}_{l=0}^{L}$, gdje je G_0 originalna slika (najfinija rezolucija), a G_L slika najgrublje rezolucije.

- Algoritam se odvija iterativno po razinama piramide, počevši od najgrublje ($l = L$) prema najfinijoj ($l = 0$).
- **Ključna ideja**: Informacije o "sumnjivim" podacima s grubljih razina koriste se za usmjeravanje uzorkovanja i analize na finijim razinama

Višerezolucijski pristup - Generalna ideja

Prednosti

- Povećava vjerojatnost da će anomalija biti prikladno zastupljena u uzorku na finijim razinama
- Dozvoljava postavljanje nižih pragova za detekciju sumnjivih područja na grubljim razinama, jer konačna odluka pada na najfinijoj razini
- Računski efikasniji jer se na grubljim razinama koriste manji patchevi

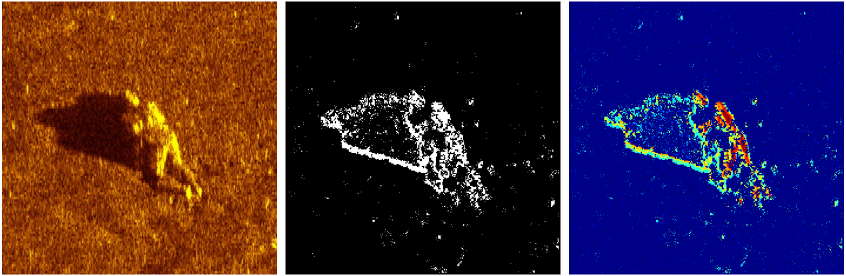
Gaussova piramida

Umjesto primjene algoritma na samo jednoj rezoluciji slike, koristi se **Gaussova piramida** slike: $\{G_l\}_{l=0}^{L}$, gdje je G_0 originalna slika (najfinija rezolucija), a G_L slika najgrublje rezolucije.

- Gaussov filter
- "downsampling"
- impyramid(...)

Gaussova piramida

Rezultat



Zaključak

- Difuzijska preslikavanja su moćan alat za nelinearnu redukciju dimenzionalnosti i analizu strukture podataka.
- Korištenjem patcheva umjesto pojedinačnih piksela omogućuje se hvatanje lokalnog konteksta i strukture, što je bitno za detekciju anomalija u slikama.
- Proširenje funkcija (npr. Laplaceovom piramidom) omogućuje primjenu na velike skupove podataka uzorkovanjem manjeg podskupa za inicijalnu konstrukciju preslikavanja.
- Višerezolucijski (multiscale) pristup dodatno poboljšava robusnost i efikasnost detekcije:
  - Usmjerava proces uzorkovanja na finijim skalama prema područjima koja su se pokazala sumnjivima na grubljim skalama.
  - Povećava šanse za detekciju anomalija različitih veličina.
  - Može biti računski efikasniji od nekih jednorezolucijskih metoda.
- Mjera anomalije temeljena na gustoći susjedstva u prostoru difuzijskih koordinata omogućuje kvantifikaciju "čudnosti" svakog podatkovnog objekta (patcha).