



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

## ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

# ODLIŠENÍ POZADÍ A POHYBUJÍCÍCH SE OBJEKTŮ VE VIDEOSEKVENCI

SEPARATION OF BACKGROUND AND MOVING OBJECTS IN VIDEOSEQUENCE

## DIPLOMOVÁ PRÁCE

MASTER'S THESIS

## AUTOR PRÁCE

AUTHOR

Bc. Lucia Komůrková

## VEDOUCÍ PRÁCE

SUPERVISOR

doc. Mgr. Pavel Rajmic, Ph.D.

BRNO 2018

# Zadání diplomové práce

Ústav:	Ústav matematiky
Studentka:	<b>Bc. Lucia Komůrková</b>
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Matematické inženýrství
Vedoucí práce:	<b>doc. Mgr. Pavel Rajmic, Ph.D.</b>
Akademický rok:	2017/18

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

## **Odlišení pozadí a pohybujících se objektů ve videosekvenci**

### **Stručná charakteristika problematiky úkolu:**

Videozáznam je série snímků, v nichž se pohybuje nějaký objekt, ale obvykle pozadí zůstává stále stejné. To je typické např. pro bezpečnostní kamery. Takové video pak můžeme chápat jako posloupnost součtů dvojic snímků, kdy jeden ze dvojice je stále stejný. Při vhodném uspořádání pixelů z jednotlivých snímků lze pozadí automaticky identifikovat (a získat) jako nízkohodnotní strukturu, zatímco pohybující se objekt v datech působí tzv. řídkou změnu. Tuto úlohu je možné aproximativně formulovat jako konvexní optimalizační problém.

### **Cíle diplomové práce:**

Cílem práce je nastudovat uvedenou problematiku, na základě odborných článků navrhnout a implementovat algoritmus pro tuto separaci v Matlabu. Dále ověřit funkčnost na simulovaných i reálných datech a porovnat s jinými metodami separace, a to jak kvalitativně, tak kvantitativně.

### **Seznam doporučené literatury:**

ELAD, M. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. New York: Springer, 2010. ISBN 978-1-4419-7011-4.

HRBÁČEK, R., P. RAJMIC, V. VESELÝ a J. ŠPIŘÍK. Řídké reprezentace signálů: Úvod do problematiky. Elektrotechnika - Internetový časopis, 2011, roč. 2011, č. 50, s. 1-10. ISSN: 1213-1539. URL: <http://elektrotechnika.cz/cz/download/ridke-reprezentace-signalu--uvod-do-problematiky/>

GONZALES, R. a R. WOODS. Digital Image Processing. Prentice Hall, 3rd ed. ISBN 978-0--3-168728-8.

SOLTANOLKOTABI, M., E. ELHAMIFAR a E. J. CANDÈS. Robust subspace clustering. Annals of Statistics 42(2), p. 669-699.

CANDÈS, E. J., X. LI, Y. MA a J. WRIGHT. Robust Principal Component Analysis? Journal of ACM 58(1), p.1-37. 2009.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2017/18

V Brně, dne

L. S.

---

prof. RNDr. Josef Šlapal, CSc.  
ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
děkan fakulty

#### ABSTRAKT

Diplomová práca sa zaoberá odlišením pozadia a pohybujúcich sa objektov vo video-zázname. Videozáznam môžeme reprezentovať ako sériu snímok a každú snímku ako nízkohodnostnú štruktúru - maticu. Táto práca popisuje riedke reprezentácie signálov a robustnú analýzu hlavných komponentov. Ďalej predstavuje a implementuje algoritmy, modely pre rekonštrukciu reálneho videozáznamu.

#### SUMMARY

This diploma thesis deals with separation of background and moving objects in video. Video can be represented as series of frames and each frame represented as low-rank structure - matrix. This thesis describes sparse representation of signals and robust principal component analysis. It also presents and implements algorithms - models for reconstruction of real video.

#### KLÚČOVÉ SLOVÁ

riedke reprezentácie signálov, analýza hlavných komponentov, robustná analýza hlavných komponentov, videozáznam, pozadie, pohybujúce sa objekty, rozšírený Lagrangián, proximálny operátor

#### KEYWORDS

sparse representation of signals, principal component analysis, robust principal component analysis, video, background, moving objects, augmented Lagrange, proximal operator

KOMŮRKOVÁ, L. *Odlišení pozadí a pohybujících se objektů ve videosekvenci*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2018. 43s. Vedoucí doc. Mgr. Pavel Rajmic, Ph.D.

Prehlasujem, že som diplomovú prácu Odlíšenie pozadia a pohybujúcich sa objektov vo videosekvencii vypracovala sama za použitia dostupnej literatúry.

Bc. Lucia Komůrková

Týmto sa chcem poďakovať vedúcemu diplomovej práce doc. Mgr. Pavlovi Rajmicovi, Ph.D. za odbornú pomoc, cenné rady a trpezlivosť pri tvorbe tejto práce.

Bc. Lucia Komůrková



## Obsah

1 Úvod .....	2
2 Motivácia .....	3
3 Riedke reprezentácie signálov .....	4
3.1 Normy vektorov a matíc, riedkosť .....	4
3.2 Riedke riešenia systémov lineárnych rovníc .....	5
3.2.1 Postačujúce podmienky pre jednoznačnosť riešenia .....	5
3.3 $\ell_1$ relaxácia .....	7
3.3.1 Podmienky ekvivalencie riešenia $\ell_0$ - a $\ell_1$ - minimalizácie .....	8
4 Singulárny rozklad a Analýza hlavných komponentov .....	10
4.1 Existencia singulárneho rozkladu .....	10
4.2 Analýza hlavných komponentov .....	12
4.2.1 Výpočet PCA užitím SVD .....	13
4.3 Robustná analýza hlavných komponentov .....	15
4.3.1 Základná úloha .....	15
5 Proximálne algoritmy .....	18
5.1 Proximálny operátor .....	18
5.1.1 Proximálny operátor pre $\ell_1$ -normu .....	18
5.1.2 Proximálny operátor pre nukleárnu normu .....	20
5.2 Proximálna gradientná metóda (dopredné - spätné delenie) .....	20
6 Metóda rozšírených Lagrangeových multiplikátorov .....	22
6.1 Základná Lagrangeova metóda .....	22
6.2 Rozšírená Lagrangeova metóda .....	23
7 Realizácia algoritmov a výsledky .....	24
7.1 PCP model .....	24
7.2 FB model .....	25
7.3 Implementácia modelov v MATLABE .....	27
7.3.1 Reálne video .....	27
7.3.2 Testovacie video .....	34
9 Záver .....	37

# 1 ÚVOD

Ukladanie informácií vo video formáte je v dnešnej dobe celkom bežná záležitosť. Bezpečnostné kamery sú najčastejšie používané v obchodných centrách, na stanicích či už metra, vlaku alebo autobusov. Kamera dnes už nie je výnimkou ani v aute, kde sníma premávku, prípadne záznamy z dopravných nehôd. Takýto videozáznam je len séria snímok, v ktorých sa pohybujú určité objekty ale pozadie zostáva nezmenené. Snímky chápeme ako pixely, ktoré sa dajú reprezentovať konkrétnym číslom, samotná snímka predstavuje maticu. Pri vhodnom usporiadaní pixelov, môžeme z videozáznamu obdržať obe časti - pozadie ako nízkohodnostnú štruktúru a pohybujúce sa objekty ako riedke reprezentácie. Tento problém sa dá reprezentovať ako konvexná optimalizačná úloha. Táto úloha predstavuje jednu z aplikácií robustnej analýzy hlavných komponentov, modernej matematickej teórie, ktorá je v mnohých smeroch účinnejšia ako analýza hlavných komponentov.

Práca sa zaoberá práve separáciou pozadia a pohybujúcich sa objektov z reálneho videozáznamu. Úvodná kapitola 3 sa zaoberá tzv. riedkymi reprezentáciami signálov - zhrňuje základné poznatky a vlastnosti. V ďalších kapitolách 4 a 5 sa snažíme stručne ale výstižne popísať nie práve ľahké matematické pozadie robustnej analýzy hlavných komponentov. Ďalej v kapitolách 6 a 7 popíšeme algoritmy, ktoré budú využité pre reálny videozáznam - konkrétne dopredné-spätné delenie a metóda rozšíreného Lagrangianu. V záverečnej kapitole aplikujeme algoritmy na vlastný videozáznam, reprezentujeme dosiahnuté výsledky a nakoniec porovnáme jednotlivé algoritmy kvalitatívne aj kvantitatívne.

## 2 MOTIVÁCIA

Video sa stalo základnou reprezentáciou zaujímavostí, udalostí a v dnešnej dobe sa používa vo veľkej miere aj v oblastiach života ako je zábava, verejná bezpečnosť, zdravotná starostlivosť. V dôsledku toho hraje video analýza kľúčový význam.

Rozdelenie videa na dve komplementárne zložky tzv. pozadie (background) a popredie (foreground) sa stalo užitočnou technikou video analýzy v mnohých aplikáciách, napr. detekcia pohybu, rozpoznávanie objektov, rozpoznávanie tvári.

Pokroky v tejto oblasti, konkrétne v robustnej analýze hlavných komponentov, sú zásadné. Môžu byť využité na rôzne aplikácie. Mnohí autori porovnávajú a vylepšujú hlavne robustnú analýzu hlavných komponentov použitím PCP, avšak ani dnes ešte neexistuje taký algoritmus, ktorý by dokázal riešiť všetky kľúčové problémy, ktoré sa objavujú pri dnešných videozáznamoch.

V tejto práci zhrňujeme techniky, ktoré umožňujú separovať pozadie a popredie. Hlavnou myšlienkou je zhrnúť všetky informácie o pohybe do jednej matice, ktorú potom budeme rozkladať pomocou algoritmov. Všetky algoritmy budú implementované v prostredí MATLAB.

# 3 RIEDKE REPREZENTÁCIE SIGNÁLOV

## 3.1 NORMY VEKTOROV A MATÍC, RIEDKOSŤ

Najskôr uvedieme základné pojmy, ktoré budeme ďalej používať v texte. Vektory a matice budeme značiť tučne napr.  $\mathbf{x}$ ,  $\mathbf{A}$ , skalárne hodnoty značíme napr.  $\beta$ ,  $\sigma$ .

DEFINÍCIA 3.1. [10] *Nosičom vektoru  $\mathbf{x}$  myslíme množinu indexov, v ktorých má vektor nenulové hodnoty. Túto množinu označíme  $\text{supp}(\mathbf{x})$ ,*

$$\text{supp}(\mathbf{x}) = \{i | x_i \neq 0\}.$$

DEFINÍCIA 3.2. [10] *Nech  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{x} = (x_1, \dots, x_n)^\top$ . Potom  $\ell_p$  norma vektoru  $\mathbf{x} \in \mathbb{R}^n$  je definovaná takto:*

$$\|\mathbf{x}\|_p = \left( \sum_{i=1}^n |x_i|^p \right)^{1/p} \quad \text{pre } 1 \leq p < \infty,$$

$$\|\mathbf{x}\|_p = \sum_{i=1}^n |x_i|^p \quad \text{pre } 0 < p < 1,$$

$$\|\mathbf{x}\|_\infty = \max_i |x_i|,$$

$$\|\mathbf{x}\|_0 = |\text{supp}(\mathbf{x})|.$$

Špeciálne,  $\|\mathbf{x}\|_1$  predstavuje súčet absolútnych hodnôt prvkov vektoru,  $\|\mathbf{x}\|_0$  počet nenulových zložiek vektoru.

DEFINÍCIA 3.3. [10] *Vektor  $\mathbf{x} \in \mathbb{R}^n$  nazveme  $k$ -riedkym ( $k$ -sparse), ak platí*

$$\|\mathbf{x}\|_0 \leq k.$$

To znamená, že  $k$ -riedky vektor má najviac  $k$  nenulových zložiek.

DEFINÍCIA 3.4. [14] *Ak je  $T$  lineárny operátor medzi Hilbertovými priestormi (v konečne dimenzionálnom prípade  $T: \mathbb{C}^n \rightarrow \mathbb{C}^m$  ide o maticu rozmerov  $m \times n$ ), jeho operátorová (spektrálna) norma je definovaná ako*

$$\|T\| = \sup_{\mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq \mathbf{0}} \frac{\|T\mathbf{x}\|_2}{\|\mathbf{x}\|_2}.$$

DEFINÍCIA 3.5. [14] *Frobeniovu normu matice  $\mathbf{A}$  definujeme ako*

$$\|\mathbf{A}\|_F = \sqrt{\sum_i \sum_j |a_{ij}|^2}.$$

DEFINÍCIA 3.6. [14] *Nukleárnu normu definujeme ako  $\ell_1$ -normu singulárnych čísiel,*

$$\|\mathbf{A}\|_* = \|\boldsymbol{\sigma}(\mathbf{A})\|_1 = \sum_{i=1}^n \sigma_i.$$

## 3.2 RIEDKE RIEŠENIA SYSTÉMOV LINEÁRNYCH ROVNÍC

Najčastejšou úlohou je riešiť sústavu lineárnych rovníc  $\mathbf{Ax} = \mathbf{y}$  s podmienkou, že hľadaný vektor  $\mathbf{x}$  má byť čo najredší, tzn. mal by obsahovať čo najväčší počet nulových zložiek. Túto úlohu zapíšeme ako:

$$\min_x \|\mathbf{x}\|_0 \text{ vzhľadom k } \mathbf{Ax} = \mathbf{y}, \quad (\text{P0})$$

kde poznáme vektor  $\mathbf{y} \in \mathbb{R}^m$  (pozorovanie, meranie, signál) a maticu  $\mathbf{A} \in \mathbb{R}^{m \times n}$ . Predpokladáme iba prípady, kedy  $m \ll n$ , a  $\mathbf{A}$  je plnej hodnosti [10].

V praxi, pri zašumených signáloch, je povolená malá odchýlka  $\delta$  od presného riešenia :

$$\min_x \|\mathbf{x}\|_0 \text{ vzhľadom k } \|\mathbf{Ax} - \mathbf{y}\|_2 \leq \delta. \quad (1)$$

### 3.2.1 POSTAČUJÚCE PODMIENKY PRE JEDNOZNAČNOSŤ RIEŠENIA

Pre jednoznačnosť riešenia je potrebné zaviesť pojem spark.

DEFINÍCIA 3.7. [19] *Nech  $\varphi: V \rightarrow U$  je lineárne zobrazenie medzi vektorovými priestormi nad poľom  $K$ . Jeho jadrom nazývame množinu*

$$\ker \varphi = \varphi^{-1}(0) = \{\mathbf{x} \in V; \varphi(\mathbf{x}) = 0\}.$$

DEFINÍCIA 3.8. [10] *Číslo  $\text{spark}(\mathbf{A})$  definujeme ako najmenší počet stĺpcov matice  $\mathbf{A}$ , ktoré sú lineárne závislé. Formálne:*

$$\text{spark}(\mathbf{A}) = \min_{\substack{\mathbf{z} \in \ker \mathbf{A}, \\ \mathbf{z} \neq 0}} \|\mathbf{z}\|_0,$$

*kde  $\ker \mathbf{A}$  značí jadro lineárneho zobrazenia určeného maticou  $\mathbf{A}$ .*

Čím menší spark, tým redší musí byť vektor  $\mathbf{x}$ , aby bola zaistená jedinečnosť tohoto riešenia. Ďalšie tvrdenie uvádza postačujúcu podmienku pre jednoznačnosť riešenia.

TVRDENIE 3.9. [10] *Pokiaľ má sústava  $\mathbf{Ax} = \mathbf{y}$  riešenie  $\mathbf{x}$  splňujúce*

$$\|\mathbf{x}\|_0 < \frac{\text{spark}(\mathbf{A})}{2},$$

*potom  $\mathbf{x}$  je nutne najredšie možné riešenie a žiadne iné riešenie s rovnakou riedkosťou neexistuje.*

Avšak nájdenie  $\text{spark}(\mathbf{A})$  je výpočtovo veľmi náročné, preto je nutné hľadať jednoduchší spôsob overenia jedinečnosti riešenia.

DEFINÍCIA 3.10. [10] *Vzájomná koherencia (mutual coherence) matice  $\mathbf{A}$  je definovaná ako najväčší absolútny normalizovaný skalárny súčin dvoch rôznych stĺpcov matice  $\mathbf{A}$ :*

$$\mu(\mathbf{A}) = \max_{\substack{1 \leq j, k \leq n, \\ j \neq k}} \frac{|\mathbf{a}_j^\top \mathbf{a}_k|}{\|\mathbf{a}_j\|_2 \cdot \|\mathbf{a}_k\|_2},$$

*kde  $\mathbf{a}_j$  resp.  $\mathbf{a}_k$  označuje  $j$ -tý resp.  $k$ -tý stĺpec matice  $\mathbf{A}$ .*

Jedine unitárne matice majú nulovú koherenciu, pretože ich stĺpce sú po dvoch ortogonálne (čitateľ je rovný nule).

TVRDENIE 3.11. [10] *Pre ľubovoľnú maticu  $\mathbf{A}$  platí:*

$$\text{spark}(\mathbf{A}) \geq 1 + \frac{1}{\mu(\mathbf{A})}.$$

Táto podmienka nám umožňuje zdola ohraničiť  $\text{spark}(\mathbf{A})$ , a pritom nie je tak výpočtovo náročná.

TVRDENIE 3.12. [10] *Pokiaľ má sústava  $\mathbf{Ax} = \mathbf{y}$  riešenie  $\mathbf{x}$  splňujúce*

$$\|\mathbf{x}\|_0 < \frac{1}{2} \left( 1 + \frac{1}{\mu(\mathbf{A})} \right),$$

*potom  $\mathbf{x}$  je nutne najredšie možné a je jediné. Navyac toto riešenie môžeme dosiahnuť  $\ell_1$ -minimalizáciou. (ďalej vysvetlená v časti (3.3))*

### 3.3 $\ell_1$ RELAXÁCIA

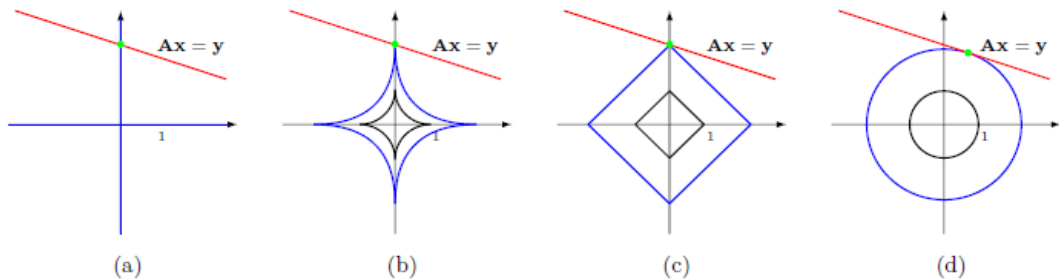
Keďže norma  $\ell_0$  nie je konvexná funkcia, nie je možné pre problém (P0) použiť akýkoľvek algoritmus konvexného programovania. Avšak normy  $\ell_p$  sú konvexné pre  $p \geq 1$ , viz Obrázok 1. Preto sa naskytá otázka použiť najbližšiu konvexnú normu, teda  $\ell_1$  normu, pre získanie aspoň približného riešenia našej NP ťažkej úlohy. Nová úloha by vyzerala takto:

$$\min_x \|x\|_1 \text{ vzhľadom k } Ax = y \quad (\text{P1})$$

Riešenia úloh (P0) a (P1) sa prekvapivo v niektorých prípadoch zhodujú a v ďalšej kapitole uvedieme za akých podmienok sa dá použiť táto aproximácia. V prípade zašumených dát riešime úlohu

$$\min_x \|x\|_1 \text{ vzhľadom k } \|Ax - y\|_2 \leq \delta. \quad (2)$$

Obrázok 1 ilustruje riešenie úlohy  $\min_x \|x\|_p$  vzhľadom k  $Ax = y$  v priestore  $\mathbb{R}^2$ , postupne pre  $p = 0; 0,5; 1; 2$ . Prípustné riešenia reprezentuje červená priamka. Z obrázku vidíme, že riešenie oboch problémov je zhodné. V prípade euklidovskej normy,  $p = 2$ , je riešenie odlišné. Jedná sa o riešenie s minimálnou energiou [14].



Obrázok 1: "Nafukujúce sa" gule v normách (a)  $\ell_0$ , (b)  $\ell_{0,5}$ , (c)  $\ell_1$ , (d)  $\ell_2$  a ich dotyk s nadrovinou určenou sústavou  $Ax = y$  [14].

### 3.3.1 PODMIENKY EKVIVALENCIE RIEŠENIA $\ell_0$ - A $\ell_1$ - MINIMALIZÁCIE

#### VLASTNOSŤ NULOVÉHO PRIESTORU

Pre  $T \subset \{1, \dots, n\}$  označíme  $\mathbf{x}_T \in \mathbb{R}^n$  vektor odvodený z  $\mathbf{x} \in \mathbb{R}^n$  tak, že prvky na pozíciách patriacich do množiny  $T$  zachováme a ostatné vynulujeme. Komplement  $T$  označíme ako  $T^c = \{1, \dots, n\} \setminus T$ . Teraz zadefinujeme vlastnosť nulového priestoru:

DEFINÍCIA 3.13. [10] *Povieme, že matica  $A \in \mathbb{R}^{m \times n}$  splňuje vlastnosť nulového priestoru  $k$ -tého stupňa s konštantou  $\rho \in (0,1)$ , ak platí*

$$\|\boldsymbol{\eta}_T\|_1 \leq \rho \|\boldsymbol{\eta}_{T^c}\|_1,$$

pre všetky množiny  $T \subset \{1, \dots, n\}$ ,  $|T| \leq k$  a pre všetky vektory  $\boldsymbol{\eta} \in \ker(A)$ .

Vlastnosť nulového priestoru zaisťuje jednoznačné  $k$ -riedke riešenie, ktoré nájdeme pomocou  $\ell_1$  minimalizácie a zaručuje jeho jednoznačnosť.

DEFINÍCIA 3.14. [10] *Chyba najlepšej aproximácie vektoru  $\mathbf{x} \in \mathbb{R}^n$   $k$ -riedkym vektorom  $\mathbf{z}$  v norme  $\ell_p$  je definovaná ako*

$$\sigma_k(\mathbf{x})_p = \inf_{\mathbf{z} \in \Sigma_k^n} \|\mathbf{x} - \mathbf{z}\|_p,$$

kde  $\Sigma_k^n = \{\mathbf{x} \in \mathbb{R}^n: \|\mathbf{x}\|_0 \leq k\}$ .

Nasledujúce tvrdenie dáva horný odhad chyby aj v ostatných prípadoch:

TVRDENIE 3.15. [10] *Nech matica  $A \in \mathbb{R}^{m \times n}$  splňuje vlastnosť nulového priestoru  $k$ -tého stupňa s konštantou  $\rho \in (0,1)$ . Nech  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{y} = A\mathbf{x}$  a  $\mathbf{x}^* \in \mathbb{R}^n$  je riešenie  $\ell_1$ -minimalizácie. Potom*

$$\|\mathbf{x} - \mathbf{x}^*\|_1 \leq \frac{2(1+\rho)}{1-\rho} \sigma_k(\mathbf{x})_1.$$

Ak teda existuje nejaké  $k$ -riedke riešenie sústavy  $\mathbf{y} = A\mathbf{x}$ , potom  $\sigma_k(\mathbf{x})_1 = 0$  a na pravej strane sa vynúti  $\mathbf{x} = \mathbf{x}^*$ . Môžeme konštatovať, že riedke riešenie nájdeme aj  $\ell_1$  minimalizáciou.

## VLASTNOST' ZOSLABENEJ IZOMETRIE

DEFINÍCIA 3.16. [10] Konštanta obmedzenej izometrie  $\delta_k$  matice  $\mathbf{A} \in \mathbb{R}^{m \times n}$  je najmenšie číslo také, že platí

$$(1 - \delta_k) \leq \frac{\|\mathbf{A}\mathbf{z}\|_2^2}{\|\mathbf{z}\|_2^2} \leq (1 + \delta_k)$$

Pre všetky vektory  $\mathbf{z} \in \Sigma_k^n$ . Povieme, že matica  $\mathbf{A}$  splňuje vlastnosť zoslabenej izometrie  $k$ -tého stupňa s konštantou  $\delta_k$ , pokiaľ  $\delta_k \in (0,1)$ .

Izometria znamená, že lineárne zobrazenie zachováva dĺžku vektorov, tzn. je unitárne. Uvedieme ešte vzťah medzi vlastnosťou nulového priestoru a vlastnosťou zoslabenej izometrie.

TVRDENIE 3.17. [10] Nech matica  $\mathbf{A} \in \mathbb{R}^{m \times n}$  splňuje vlastnosť zoslabenej izometrie stupňa  $K = k + h$  s konštantou  $\delta_K \in (0,1)$ . Potom  $\mathbf{A}$  splňuje vlastnosť nulového priestoru stupňa  $k$  s konštantou

$$\rho = \sqrt{\frac{k(1 + \delta_K)}{h(1 - \delta_K)}}.$$

# 4 SINGULÁRNY ROZKLAD A ANALÝZA HLAVNÝCH KOMPONENTOV

Analýza hlavných komponentov je pravdepodobne jeden z najrozšírenejších štatistických nástrojov pre analýzu vysoko rozmerných dát a zníženie ich dimenzie. Ďalej v texte budeme využívať už len skratku PCA z anglického *Principal component analysis*.

## 4.1 EXISTENCIA SINGULÁRNEHO ROZKLADU

Singulárny rozklad matice je podobný diagonalizácii normálnej matice. Diagonalizácia matice rozkladá maticu na faktory pomocou vlastných hodnôt a vlastných vektorov. Diagonalizácia matice  $\mathbf{A}$  má formu  $\mathbf{A} = \mathbf{VDV}^T$ , kde stĺpce matice  $\mathbf{V}$  sú vlastné vektory matice  $\mathbf{A}$  a vytvárajú ortonormálnu bázu v  $\mathbb{R}^n$ , a  $\mathbf{D}$  je diagonálna matica s diagonálnymi prvkami pozostávajúcimi z vlastných hodnôt. Na druhej strane, singulárny rozklad má formu  $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{W}^T$ . Stĺpce matíc  $\mathbf{U}$  a  $\mathbf{W}$  sa nazývajú ľavé a pravé singulárne vektory matice  $\mathbf{A}$ , a matica  $\mathbf{\Sigma}$  je diagonálna matica s diagonálnymi prvkami pozostávajúca zo singulárnych hodnôt matice  $\mathbf{A}$ .

Singulárny rozklad má mnoho aplikácií pri spracovávaní signálov. Napríklad ak chceme nájsť maticu s nízkou hodnotou, ktorá nám dobre aproximuje maticu pozorovaní  $\mathbf{A}$ . V ďalších kapitolách sa budeme na singulárny rozklad odvolávať skratkou SVD, čo pochádza z anglického *Singular value decomposition*. Túto kapitolu spracovávame podľa [6],[15],[19].

DEFINÍCIA 4.1. [19] *Hermitovsky adjungovaná matica  $\mathbf{A}^*$  ku matici  $\mathbf{A} \in \mathbb{C}^{m \times n}$  je definovaná ako  $\mathbf{A}^* = \overline{\mathbf{A}^T}$ , pričom  $\overline{\mathbf{A}^T}$  označuje maticu transponovanú a komplexne združenú k matici  $\mathbf{A}$ . Potom matica je samoadjungovaná alebo Hermitovská ak  $\mathbf{A}^* = \mathbf{A}$ .*

DEFINÍCIA 4.2. *Matica  $\mathbf{U} \in \mathbb{C}^{n \times n}$  nazýva unitárna ak  $\mathbf{U}^* \mathbf{U} = \mathbf{U} \mathbf{U}^* = \mathbf{I}_n$ , kde  $\mathbf{I}_n$  je jednotková matica typu  $n \times n$ . Ak  $\mathbf{U} \in \mathbb{R}^n$ ,  $\mathbf{U}$  je reálne ortogonálna.*

DEFINÍCIA 4.3. *Matica  $\mathbf{A} \in \mathbb{C}^{n \times n}$  je normálna matica ak  $\mathbf{A}^* \mathbf{A} = \mathbf{A} \mathbf{A}^*$ .*

Potom aj matica  $\mathbf{A} \in \mathbb{R}^{n \times n}$  je normálna práve vtedy, keď  $\mathbf{A}^T \mathbf{A} = \mathbf{A} \mathbf{A}^T$ .

DEFINÍCIA 4.4. [19] *Nech  $U, V$  sú vektorové priestory nad tým istým poľom  $K$ . Hovoríme, že  $\varphi: V \rightarrow U$  je lineárne zobrazenie, ak  $\varphi$  zachováva operácie vektorového súčtu a skalárneho násobku, t.j. ak pre ľubovoľné  $\mathbf{x}, \mathbf{y} \in V, c \in K$  platí*

$$\varphi(\mathbf{x} + \mathbf{y}) = \varphi(\mathbf{x}) + \varphi(\mathbf{y}),$$

$$\varphi(c\mathbf{x}) = c\varphi(\mathbf{x}).$$

Prvá z uvedených vlastností sa nazýva aj aditivita a druhá homogenita.

DEFINÍCIA 4.5. *Lineárne zobrazenie  $\varphi: V \rightarrow V$  vektorového priestoru  $V$  do seba nazývame lineárnym operátorom alebo lineárnou transformáciou.*

DEFINÍCIA 4.6. [19] *Komplexný vektorový priestor  $V$  so skalárnym súčinom nazývame unitárny priestor. Skalárny súčin na  $V$  môžeme definovať ako binárnu operáciu  $V \times V \rightarrow \mathbb{C}$ , ktorá každej dvojici vektorov  $(\mathbf{x}, \mathbf{y})$  z  $V$  priradí komplexné číslo  $\langle \mathbf{x}, \mathbf{y} \rangle$  také, že pre všetky  $\mathbf{x}, \mathbf{y}, \mathbf{x}_1, \mathbf{x}_2 \in V$  a ľubovoľné  $c \in \mathbb{C}$  platí:*

$$\langle \mathbf{x}_1 + \mathbf{x}_2, \mathbf{y} \rangle = \langle \mathbf{x}_1, \mathbf{y} \rangle + \langle \mathbf{x}_2, \mathbf{y} \rangle \quad (\text{aditivita})$$

$$\langle c\mathbf{x}, \mathbf{y} \rangle = c\langle \mathbf{x}, \mathbf{y} \rangle \quad (\text{homogenita})$$

$$\langle \mathbf{x}, \mathbf{y} \rangle = \overline{\langle \mathbf{y}, \mathbf{x} \rangle} \quad (\text{kosá symetria})$$

$$\mathbf{x} \neq \mathbf{0} \Rightarrow \langle \mathbf{x}, \mathbf{x} \rangle > 0 \quad (\text{kladná definitnosť}).$$

Ďalej uvedieme základnú vetu o singulárnom rozklade. Pre jednoduchosť budeme častejšie pracovať s lineárnymi transformáciami namiesto s maticami samotnými. Akýkoľvek s nasledujúcich výsledkov pre lineárne transformácie platí tiež pre matice.

VETA 4.7. VETA O SINGULÁRNOM ROZKLADE

*Nech  $V$  a  $W$  sú konečne-dimenziálne unitárne priestory, a nech  $T: V \rightarrow W$  je lineárna transformácia hodnosti  $r$ . Potom existujú ortonormálne bázy  $\{v_1, v_2, \dots, v_n\}$  pre  $V$  a  $\{u_1, u_2, \dots, u_m\}$  pre  $W$  a pozitívne čísla  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$  také, že*

$$T(v_i) = \begin{cases} \sigma_i u_i, & \text{ak } 1 \leq i \leq r \\ 0, & \text{ak } i > r. \end{cases}$$

*Predpokladajme, že predchádzajúce podmienky sú splnené. Potom pre  $1 \leq i \leq n$ ,  $v_i$  je vlastný vektor  $T^*T$  s príslušnou vlastnou hodnotou  $\sigma_i^2$ , keď  $1 \leq i \leq r$  a 0 ak  $i > r$ . Preto skaláry  $\sigma_1, \sigma_2, \dots, \sigma_r$  sú jednoznačne určené  $T$ .*

DEFINÍCIA 4.8. Skaláry  $\sigma_1, \sigma_2, \dots, \sigma_r$  z Vety 4.7. sa nazývajú *singulárne hodnoty*  $T$ . Ak  $r$  je menšie ako oba rozmery  $m$  a  $n$ , potom termín *singulárne hodnoty* môžeme rozšíriť aj na  $\sigma_{r+1} = \dots = \sigma_k = 0$ , kde  $k$  je minimum z  $m$  a  $n$ .

VETA 4.9. Ak matica  $\mathbf{A} \in \mathbb{C}^{m \times n}$  je hodnosti  $r$  s pozitívnymi singulárnymi hodnotami  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ , potom môžeme maticu napísať aj vo forme

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{W}^*$$

kde  $\mathbf{U} \in \mathbb{C}^{m \times m}$  a  $\mathbf{W} \in \mathbb{C}^{n \times n}$  sú unitárne matice a matica  $\mathbf{\Sigma} = |\Sigma_{ij}| \in \mathbb{C}^{m \times n}$  je daná ako

$$\Sigma_{ij} = \begin{cases} \sigma_i, & \text{pre } i = j \leq r \\ 0, & \text{inak.} \end{cases}$$

DEFINÍCIA 4.10. Ak matica  $\mathbf{A} \in \mathbb{C}^{m \times n}$  je hodnosti  $r$  s pozitívnymi singulárnymi hodnotami  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r$ , potom *singulárny rozklad matice*  $\mathbf{A}$  je faktorizácia  $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{W}^*$ , kde  $\mathbf{U}$  a  $\mathbf{W}$  sú unitárne a  $\mathbf{\Sigma} \in \mathbb{C}^{m \times n}$  je definovaná podľa Vety 4.9.

## 4.2 ANALÝZA HLAVNÝCH KOMPONENTOV

Analýza hlavných komponentov (PCA z angličtiny Principal Component Analysis) je v dnešnej dobe štandardný nástroj k dátovým analýzám v rôznych oblastiach vedy. Je to jednoduchá, neparametrická metóda na získavanie znalostí a informácií v rôznych dátových súboroch.

PCA poskytuje návod ako jednoducho znížiť dimenziu komplexných údajov pri zachovaní trendov a vzorov. PCA je metóda podobná zhlukovaniu (clustering) – nachádza vzory bez akýchkoľvek predchádzajúcich znalostí. PCA transformuje lineárne závislé premenné na lineárne nezávislé premenné nazývané hlavné komponenty. Hlavné komponenty sú zoradené podľa dôležitosti učenej mierou variability, ktorú popisujú. Prvé komponenty zachytávajú vlastnosti dát, ktoré majú najväčší vplyv na skúmané veličiny.

Existuje viacero možností ako získať hlavné komponenty. Najznámejšie z nich sú dekompozícia kovariačnej (alebo korelačnej) matice priamo na vlastné vektory a vlastné čísla a dekompozícia pomocou SVD. My si uvedieme práve druhý spôsob výpočtu PCA.

## 4.2.1 VÝPOČET PCA UŽITÍM SVD

Majme maticu vstupných dát  $A \in \mathbb{R}^{m \times n}$ , kde  $m$  označuje počet meraní a  $n$  počet premenných. Hlavné komponenty sú popísané vlastnými číslami a im prislúchajúcimi vlastnými vektormi. Vlastný vektor určuje smer hlavnej osi hlavného komponentu.

PCA užitím SVD nerozkladá kovariančnú maticu ale priamo maticu vstupných dát. V tom spočíva veľká výhoda jej použitia, pretože je oveľa menej náročná na pamäť počítača a pri samotnom počítaní kovariančnej matice môže dôjsť ku strate presnosti a numerickej stability.

Nech  $A \in \mathbb{R}^{m \times n}$ , potom môžeme SVD rozklad matice  $A$  napísať ako

$$A = U\Sigma W^T.$$

PCA počíta rozklad na hlavné komponenty z kovariančnej matice

$$C = \frac{A^T A}{(n-1)}.$$

Do rovnice dosadíme

$$C = \frac{W\Sigma U^T U\Sigma W^T}{(n-1)}$$

ďalej dostaneme

$$C = \frac{W\Sigma^2 W^T}{(n-1)}.$$

Ďalej nám to umožní odvodiť vzťah pre výpočet vlastných čísiel zo singulárnych hodnôt matice  $\Sigma$ . SVD umožňuje výpočtovo efektívnu metódu na získanie hlavných komponentov. Po rozklade vstupnej matice  $A$  na matice  $U, \Sigma, W$  sú  $\Sigma$  a  $U$  získané vlastné čísla a im prislúchajúce vektory. Stĺpce matice  $U$  sú vlastné vektory hlavných komponentov kovariančnej matice  $C$ . Hodnoty na diagonále matice  $\Sigma$  sú druhými odmocninami vlastných vektorov.

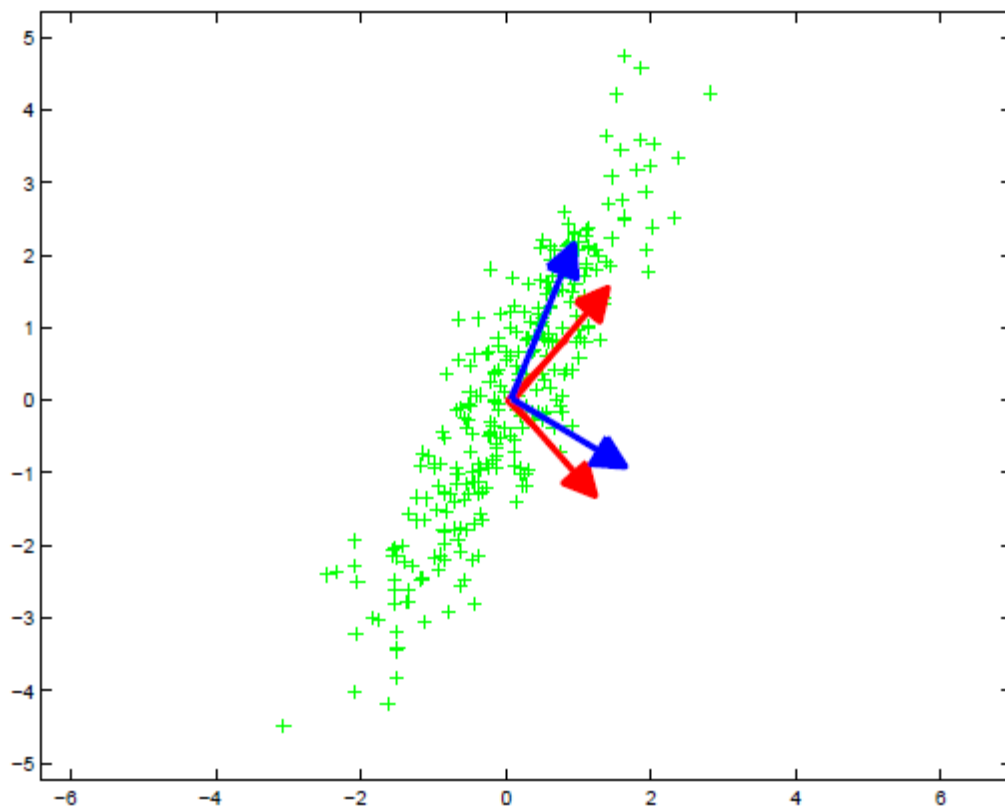
Metódou aplikácie SVD a pri hľadaní hlavných komponentov je získaných maximálne  $r$  vlastných čísel a vektorov. Ako je spomenuté vyššie,  $r$  je hodnota matice  $A$ , čo je určené počtom lineárne nezávislých stĺpcov. Avšak  $r$  môže byť maximálne

$$r \leq \min(m, n).$$

V prípade, keď do PCA vstupujú dáta s veľkým počtom vzájomne korelovaných premenných, sú výsledkom analýzy kovariančnej matice okrem pozitívnych aj negatívne vlastné čísla. Takéto vlastné čísla majú malé hodnoty a preto sú

zanedbateľné. Aplikácia SVD nájde iba komponenty s kladnými hodnotami [6], [15], [18].

Veľkou nevýhodou PCA je citlivosť na poškodené vstupy alebo na hodnoty ďaleko vzdialené, vychýlené od pôvodných dát (anglicky outliers). Na obrázku 2 sú znázornené náhodne vygenerované hodnoty z viacrozmerného normálneho rozdelenia s vektorom stredných hodnôt  $[0; 0]$  a kovariančnou maticou  $[1; 1:5; 1:5; 3]$ . Modré vektory znázorňujú prvé 2 hlavné komponenty. Do týchto dát sme pridali jeden outlier  $[6;-15]$  ako ďalšie pozorovanie (na obrázku ho nie je vidieť). Červené vektory ukazujú, ako sa zmenia hlavné komponenty pridaním len jedného porušeného pozorovania.



Obrázok 2. PCA, vygenerovaných 300 dát

## 4.3 ROBUSTNÁ ANALÝZA HLAVNÝCH KOMPONENTOV

Aj keď PCA je silný matematický nástroj, bohužiaľ v dnešnej dobe už nie dost postačujúci. Hrubé chyby pri pozorovaní vznikajú všade, napr. pri spracovaní obrazu, videa, v bioinformatike apod. Klasická PCA je na poškodenie dát veľmi citlivá. Len jedno, ale za to hrubé poškodenie na vstupe môže výrazne skresliť výsledok. V tejto kapitole sa preto zameriame na ideálnu robustnú PCA, ďalej v texte RPCA, ktorej cieľom je obnovenie matice  $L$  s nízkou hodnotou z hrubo porušených dát, pozorovaní  $M$ , tak že  $M = L + S$ . Chyby  $S$  môžu byť ľubovoľnej veľkosti, ale požadujeme ich riedkosť.

### 4.3.1 ZÁKLADNÁ ÚLOHA

Predpokladáme, že matica pozorovaní  $M \in \mathbb{R}^{m \times n}$  vznikla porušením niektorých vstupov matice s nízkou hodnotou  $L \in \mathbb{R}^{m \times n}$ . Tieto chyby reprezentujeme maticou  $S \in \mathbb{R}^{m \times n}$ , tak že  $M = L + S$ . Vzhľadom k tomu, že chyby sa vyskytujú len na časti vstupov  $M$ ,  $S$  je riedka matica. Ideálnu (bez šumu) robustnú PCA úlohu formulujeme takto:

**Úloha 1:** Je daná  $M = L + S$ , kde  $L$  a  $S$  sú neznáme matice. Vieme však, že  $L$  je matica s nízkou hodnotou a  $S$  riedka matica. Potom môžeme dostatočne obnoviť  $L$ .

Riešením tejto úlohy je zjavne hľadať  $L$  s najnižšou hodnotou, ktorá by ešte vyhovovala podmienke, že  $S$  je riedka:  $\|S\|_0 \leq k$ . Ak úlohu preformulujeme do optimalizačného problému, dostávame

$$\min_{L, S} \quad \text{rank}(L) + \gamma \|S\|_0 \tag{3}$$

za podmienky  $M = L + S$

Ak by sa nám aj podarilo vyriešiť tento problém pre konkrétnu hodnotu, pravdepodobne by sme dostali presné vyjadrenie  $L, S$ . Táto úloha je však NP optimalizačná úloha a jej riešenie zatiaľ nepoznáme. Môžeme však obdržať konvexný optimalizačný problém a to  $\ell_1$  relaxáciou úlohy (3) a nahradením  $\text{rank}(L)$  nukleárnou normou. Potom

$$\min_{L, S} \quad \|L\|_* + \lambda \|S\|_1 \quad (4)$$

$$\text{za podmienky} \quad M = L + S$$

V jednom z našich hlavných zdrojov Candes a Wright [5] dokázali, že za pomerne slabých predpokladov má úloha riešenie. Táto úloha nesie názov Principal Component Pursuit, v skratke PCP, a presne obnovuje nízko-hodnotnú maticu  $L$  a riedku  $S$ .

DEFINÍCIA 4.10. [5] *Predpokladáme SVD rozklad matice  $L_0 \in \mathbb{R}^{m \times n}$*

$$L_0 = U \Sigma V^T = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^T,$$

kde  $r$  je hodnota matice,  $\sigma_1, \dots, \sigma_r$  sú pozitívne singulárne hodnoty a  $U, V$  sú matice ľavých a pravých singulárnych vektorov. Potom podmienka inkohorentnosti s parametrom  $\mu$  je

$$\max_i \|U^T e_i\|^2 \leq \frac{\mu r}{m}$$

$$\max_i \|V^T e_i\|^2 \leq \frac{\mu r}{n}$$

$$\|UV^T\|_\infty \leq \sqrt{\frac{\mu r}{nm}},$$

kde  $\|M\|_\infty = \max_{ij} |M_{ij}|$ .

VETA 4.11. [5] *Predpokladajme  $L_0$  je typu  $n \times n$ , splňujúca podmienky z definície 4.10. Potom existuje numerická konštanta  $c$  taká, že s pravdepodobnosťou najmenej  $1 - cn^{-10}$ , Principal Component Pursuit (4) s parametrom  $\lambda = \frac{1}{\sqrt{n}}$  je presné riešenie, tzn.  $L = L_0$  a  $S = S_0$ , za predpokladu*

$$\text{rank}(L_0) \leq \rho_r n \mu^{-1} (\log n)^{-2} \quad \text{a} \quad k \leq \rho_s n^2,$$

kde  $\rho_r, \rho_s$  sú pozitívne numerické konštanty. V obecnom prípade, kedy  $L_0$  je matica typu  $m \times n$ , PCP s parametrom  $\lambda = \frac{1}{\sqrt{m}}$  splňuje s pravdepodobnosťou najmenej  $1 - cm^{-10}$ , za predpokladu  $\text{rank}(L_0) \leq \rho_r n \mu^{-1} (\log m)^{-2}$  a  $k \leq \rho_s nm$ .

Inak povedané, matica  $L_0$ , ktorej singulárne vektory alebo hlavné komponenty, sú rozumne rozložené, môže byť obnovená s pravdepodobnosťou blízkou k jednej z úplne neznámych alebo porušených vstupov. Dôkaz vety a ďalšie teóremy a lemmata k nemu potrebné čitateľ nájde v časti 2 a 3 článku [5].

RPCA má veľa iných aplikácií, v ktorých sa dá využiť PCP model:

- Kmitanie kamery: Vo veterných podmienkach sa môže stať, že sa kamera rozkmitá vo frekvencii. To na popredí spôsobí veľa chybných záznamov.
- Automatické úpravy fotoaparátu: Mnoho moderných kamier má automatické zaostrovanie, automatické vyváženie bielej farby a pod.. Tieto vplyvy môžu spôsobiť zmeny v dynamike farieb medzi jednotlivými snímkami.
- Zmeny svetla: Môžu byť postupné (napr. zmena deň, noc) alebo náhle (zapnutie svetla).
- Kamufláž: Vlastnosti pixelov predného obrazu môžu byť zmenené modelovaním zadného obrazu.
- Pohybujúce sa objekty v pozadí: Objekty sa na pozadí môžu presúvať. Tieto objekty by sa nemalo považovať za súčasť pozadia.
- Novo vložené objekty do pozadia.
- Dynamické pozadie
- Začínajúci pohyb objektu: Objekt sa začne na pozadí zrazu pohybovať, hovoríme o detekcii „ducha“.
- Tiene: V popredí môžeme detekovať tiene, ktoré pochádzajú z objektov na pozadí.

# 5 PROXIMÁLNE ALGORITMY

## 5.1 PROXIMÁLNY OPERÁTOR

DEFINÍCIA 5.1. [7] Funkcia  $f(\mathbf{x})$  je zdola polospojité na podmnožine  $X$  metrického priestoru, ak pre ľubovoľné  $\mathbf{x}_0 \in X$  platí

$$\liminf_{\mathbf{x} \rightarrow \mathbf{x}_0} f(\mathbf{x}) \geq f(\mathbf{x}_0).$$

DEFINÍCIA 5.2. [8] Nech  $f: \mathbb{R}^n \rightarrow (-\infty, \infty)$  je zdola polospojité konvexná funkcia s neprázdny definíčným oborom. Potom pre každé  $\mathbf{x} \in \mathbb{R}^n$ , minimalizačný problém

$$\min_{\mathbf{y} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 + f(\mathbf{y})$$

má jediné riešenie, ktoré označíme ako  $\text{prox}_f(\mathbf{x})$ . Takto definovaný operátor  $\text{prox}_f: \mathbb{R}^n \rightarrow \mathbb{R}^n$  nazveme proximálny operátor  $f$ .

Proximálny operátor môžeme charakterizovať inklúziou

$$(\forall (\mathbf{x}, \mathbf{p}) \in \mathbb{R}^n \times \mathbb{R}^n) \quad \mathbf{p} = \text{prox}_f(\mathbf{x}) \Leftrightarrow \mathbf{x} - \mathbf{p} = \nabla f(\mathbf{p}),$$

ak  $f$  je diferencovateľná.

Ďalej sa bližšie pozrieme na konkrétne proximálne operátory pre  $\ell_1$ -normu a nukleárnu normu.

### 5.1.1 PROXIMÁLNY OPERÁTOR PRE $\ell_1$ -NORMU

Pre  $\ell_1$ -normu definujeme  $f(\mathbf{y}) = \lambda \|\mathbf{y}\|_1$ , kde  $\lambda$  je zvolený parameter nazývaný tiež prah. Z definície dostávame

$$\text{prox}_{\lambda \|\cdot\|_1}(\mathbf{x}) = \arg \min_{\mathbf{y}} \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{y}\|_1, \quad (5)$$

čo rozpíšeme ako

$$\arg \min_{\mathbf{y}} \underbrace{\frac{1}{2} \sum_{i=1}^n (x_i - y_i)^2 + \lambda \sum_{i=1}^n |y_i|}_g \quad (6)$$

Toto minimum budeme hľadať pomocou derivácie postupne po jednotlivých intervaloch:

$$1. y_i > 0$$

$$\frac{\partial g}{\partial y_i} = y_i - x_i + \lambda = 0$$

$$y_i = x_i - \lambda > 0 \Rightarrow x_i > \lambda$$

$$2. y_i < 0$$

$$\frac{\partial g}{\partial y_i} = y_i - x_i - \lambda = 0$$

$$y_i = x_i + \lambda < 0 \Rightarrow x_i < -\lambda$$

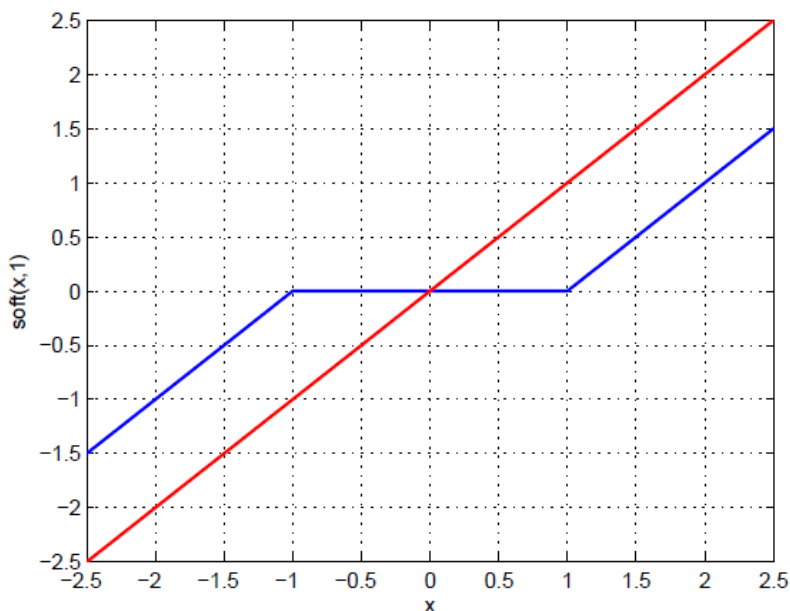
$$3. y_i = 0$$

$$-\lambda \leq x_i \leq \lambda$$

Celkovo zapíšeme proximálny operátor  $\ell_1$ - normy ako

$$y_i = \frac{x_i}{|x_i|} \max(|x_i| - \lambda, 0). \quad (7)$$

Táto funkcia, znázornená aj na obrázku 2, sa nazýva *mäkké prahovanie*, anglicky *soft thresholding*, a budeme ju značiť  $\mathcal{y} = \text{soft}_\lambda(\mathbf{x})$ .



Obrázok 2 Funkcia mäkkého prahovania. Červeným znázornená identita a modrým jej soft funkcia pre  $\lambda = 1$  [14].

## 5.1.2 PROXIMÁLNY OPERÁTOR PRE NUKLEÁRNU NORMU

Z definície nukleárnej normy  $\|\mathbf{A}\|_* = \|\sigma(\mathbf{A})\|_1 = \sum_{i=1}^n \sigma_i$  vidíme, že proximálny operátor pre nukleárnu normu je len mäkké prahovanie singulárnych čísiel. Zapišeme ako

$$\text{prox}_{\lambda\|\cdot\|_*}(\mathbf{A}) = \sum_{i=1}^n \text{soft}_{\lambda}(\sigma_i) \mathbf{u}_i \mathbf{v}_i^{\top}. \quad (8)$$

## 5.2 PROXIMÁLNA GRADIENTNÁ METÓDA (DOPREDNÉ - SPÄTNÉ DELENIE)

Znalosť proximálneho operátora nám umožní definovať jednotlivé algoritmy, ktoré nám pomôžu vyriešiť pôvodný minimalizačný problém.

Uvažujme konvexnú úlohu

$$\min_{\mathbf{x}} f_1(\mathbf{x}) + f_2(\mathbf{x}), \quad (9)$$

pričom  $f_1: \mathbb{R}^n \rightarrow \mathbb{R}$  je zdola polospojité konvexná funkcia,  $f_2: \mathbb{R}^n \rightarrow \mathbb{R}$  je konvexná a diferencovateľná s  $\beta$ -Lipschitzovským spojitým gradientom  $\nabla f_2$ , tj

$$(\forall(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n) \quad \|\nabla f_2(\mathbf{x}) - \nabla f_2(\mathbf{y})\| \leq \beta \|\mathbf{x} - \mathbf{y}\|,$$

kde  $\beta \in (0, \infty)$ .

Za predpokladu, že

$$\lim_{\|\mathbf{x} \rightarrow \infty\|} f_1(\mathbf{x}) + f_2(\mathbf{x}) = \infty,$$

má táto úloha aspoň jedno riešenie.

**TVRDENIE 5.3.** *Nech  $\mathbf{x} \in \mathbb{R}^n$  a  $\gamma \in (0, \infty)$ . Potom riešenie úlohy môžeme charakterizovať ako pevný bod*

$$\mathbf{x} = \text{prox}_{\gamma f_1}(\mathbf{x} - \gamma \nabla f_2(\mathbf{x})).$$

Tento pevný bod nám dáva možnosť riešiť úlohu iteratívne ako

$$x_{n+1} = \overbrace{\text{prox}_{\gamma_n f_1}}^{\text{spätný krok}} \underbrace{(x_n - \gamma_n \nabla f_2(x_n))}_{\text{dopredný krok}}.$$

Táto metóda je známa ako *Forward-backward splitting*, čo voľne preložíme ako dopredné spätné delenie. Ďalej v teste budeme užívať skratku FB. Obecný algoritmus je uvedený v tabuľke ako algoritmus 1. Skladá sa z dvoch krokov:

- dopredný (explicitný) gradientný krok, ktorý využíva funkciu  $f_2$
- spätný (implicitný) krok, ktorý využíva funkciu  $f_1$ .

---

ALGORITMUS 1 – FORWARD-BACKWARD SPLITTING

---

- 1) zvolíme:  $\epsilon \in ]0, \min\{1, 1/\beta\}[$ , štartovací bod  $x_0 \in \mathbb{R}^n$
  - 2) pre  $n = 0, 1, \dots$  opakujeme
    - zvolíme  $\gamma_n \in [\epsilon, 2/\beta - \epsilon]$
    - $y_n = x_n - \gamma_n \nabla f_2(x_n)$
    - $\lambda_n \in [\epsilon, 1]$
    - $x_{n+1} = x_n + \lambda_n (\text{prox}_{\gamma_n f_1} y_n - x_n)$
-

# 6 METÓDA ROZŠÍRENÝCH LAGRANGEOVÝCH MULTIPLIKÁTOROV

Túto kapitolu spracovávame podľa [1], [2], [4], [9] a [11].

## 6.1 ZÁKLADNÁ LAGRANGEOVA METÓDA

DEFINÍCIA 6.1. [17] *Funkcia  $f$  je diferencovateľná na množine  $X$ , pokiaľ pre každé  $\mathbf{x} \in \mathbf{M}$  existuje jej diferenciál  $df(\mathbf{x})$ .*

DEFINÍCIA 6.2. [17] *Funkcia  $f$  je spojitاً diferencovateľná, pokiaľ diferenciál  $df(\mathbf{x})$  je spojitý. Funkcia definovaná na otvorenej množine  $U$  je  $k$ -krát spojitاً diferencovateľná, pokiaľ má všetky parciálne derivácie  $k$ -tého stupňa spojitاً.*

Základnú úlohu pre metódu Lagrangeových multiplikátorov uvidíme na obmedzenom optimalizačnom probléme

$$\begin{aligned} \min f(\mathbf{x}) \\ \text{za podmienky } h(\mathbf{x}) = 0, \end{aligned} \tag{10}$$

kde  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  a  $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ . Predpokladáme, že problém číslo (10) má aspoň jedno prípustné riešenie. Na funkciu  $f$  kladieme podmienku hladkosti.

Potom Lagrangián

$$l(\mathbf{x}, y) = f(\mathbf{x}) + y h(\mathbf{x}), \tag{11}$$

rozpíšeme ako

$$l(\mathbf{x}, y) = f(\mathbf{x}) + \sum_i^n y_i h_i, \tag{12}$$

kde  $y$  je konštanta úmernosti. Môže to byť akékoľvek reálne číslo. Túto konštantu nazveme Lagrangeovým multiplikátorom.

Myšlienka nájdenia minima spočíva v nájdení kritických bodov cez  $\mathbf{x}$  aj  $y$ . Sice Lagrangeova funkcia je neobmedzená optimalizačná úloha, jej stacionárne body dostaneme jednoducho, a to položením prvých derivácií nule:

$$\frac{\partial l}{\partial x_j} = \frac{\partial f}{\partial x_j} + \sum_{i=1}^n y_i \frac{\partial h_i}{\partial x_j} = 0, \quad j = 1, 2, \dots, n \tag{13}$$

$$\frac{\partial l}{\partial y_i} = h_i = 0, \quad i = 1, \dots, m. \quad (14)$$

Tieto rovnice zvyčajne nazývame podmienky optimality prvého stupňa [16].

## 6.2 ROZŠÍRENÁ LAGRANGEOVA METÓDA

Metóda Augmented Lagrange multiplier, ďalej v texte budeme používať skratku ALM metóda, pracuje s tzv. rozšíreným Lagrangiánom (augmented Lagrangian). Táto metóda je jedna z najpoužívanejších tzv. algoritmov založených na pokutovej (penalizačnej) funkcii (penalty function-based algorithm). Základná myšlienka penalizačnej funkcie je eliminovať podmienky a pridať ich do účelovej funkcie (objective function) ako penalizujúci člen, ktorý nám pomôže v riešení úlohy 11.

DEFINÍCIA 6.3. [2] *Pre akýkoľvek skalár  $\mu$  definujeme rozšírený Lagrangián (augmented Lagrangian)  $l_\mu: \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}$*

$$l(x, y) = f(x) + yh(x) + \frac{1}{2} \mu |h(x)|^2. \quad (15)$$

Číslo  $\mu$  nazývame penalizujúci parameter (penalty parameter) a  $\mathbf{y}$  sú vektorové multiplikátory (multiplier vector). V texte ich budeme nazývať len multiplikátory. Za podmienok, že  $\mu_k$  je rastúca postupnosť obe funkcie  $f$  aj  $h$  sú spojitاً diferencovateľné, je dokázané [2], že Lagrangeove multiplikátory konvergujú Q-lineárne k optimálnemu riešeniu ak  $\mu_k$  je ohraničená.

Obecný algoritmus je uvedený v tabuľke ako algoritmus 2.

---

### ALGORITMUS 2 – METÓDA ROZŠÍRENÉHO LAGRANGIÁNU

---

- 1) Zvolíme:  $\mu > 0$
  - 2) pokiaľ nie je splnená podmienka konvergencie
  - 3)  $x_{k+1} = \operatorname{argmin} l(x, y_k, \mu_k)$
  - 4)  $y_k = y_k - \mu_k h(x_{k+1})$
  - 5) end while
  - 6)  $x_k$
-

# 7 REALIZÁCIA ALGORITMOV A VÝSLEDKY

## 7.1 PCP MODEL

V tejto kapitole pracujeme už len s maticami, preto upustíme z hrubého značenia matic, hlavne kvôli lepšej priehľadnosti.

### ODVODENIE ALGORITMU

Pripomeňme si PCP úlohu

$$\min_{L,S} \|L\|_* + \lambda \|S\|_1$$

za podmienky  $M = L + S$ .

Pre túto úlohu definujeme rozšírený Lagrangián takto:

$$l_\mu(L,S,Y) = \|L\|_* + \lambda \|S\|_1 + \langle Y, M - L - S \rangle + \frac{\mu}{2} \|M - L - S\|_F^2. \quad (16)$$

Pre náš rozklad môžeme efektívne vyriešiť úlohu (12) po častiach, to znamená, rozdeliť ju na dve jednoduchšie konvexné časti. Práve teraz využijeme proximálne operátory a rozpíšeme (12) ako

$$\arg \min_S l(L,S,Y) = \text{prox}_{\lambda\mu^{-1}\|\cdot\|_1}(M - L + \mu^{-1}Y) \quad (17)$$

a

$$\arg \min_L l(L,S,Y) = \text{prox}_{\mu^{-1}\|\cdot\|_*}(M - S + \mu^{-1}Y). \quad (18)$$

Oveľa praktickejšie je prvotne minimalizovať  $l$  vzhľadom k  $L$  (zafixujeme  $S$ ), potom minimalizovať  $l$  vzhľadom k  $S$  (zafixovaním  $L$ ), a nakoniec aktualizovať maticu multiplikátorov  $Y$ . Tento postup je uvedený v algoritme 3.

---

#### ALGORITMUS 3 – PCP S POUŽITÍM ALM

---

- 1) nastavíme  $S_0 = Y_0 = 0, \mu > 0$
  - 2) pokiaľ nie je splnená podmienka konvergencie
  - 3)  $L_{k+1} = \text{prox}_{\mu^{-1}\|\cdot\|_*}(M - S_k + \mu^{-1}Y_k)$
  - 4)  $S_{k+1} = \text{prox}_{\lambda\mu^{-1}\|\cdot\|_1}(M - L_{k+1} + \mu^{-1}Y_k)$
  - 5)  $Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$
-

---

6) Splnenie podmienky, výsledok L, S

---

Pre výpočet je najdôležitejšie správne nastavenia hodnôt  $\mu, \lambda$ . V tomto prípade sme použili nastavenia hodnôt podľa článku [5], a to nasledovne

$$\mu = \frac{nm}{4\|M\|_1},$$

$$\lambda = \sqrt{\frac{1}{m}}$$

a podmienku konvergencie s  $\delta = 10^{-7}$

$$\|M - L - S\|_F \leq \delta \|M\|_F.$$

## 7.2 FB MODEL

### ODVODENIE ALGORITMU

PCP úlohu rozpíšeme do tvaru

$$\arg \min_{L,S} \underbrace{\frac{1}{2} \|M - (L + S)\|_2^2}_{f_2(L,S)} + \underbrace{\lambda_L \|L\|_* + \lambda_S \|S\|_1}_{f_1(L,S)}. \quad (19)$$

Spočítame gradient  $\nabla f_2(L,S)$  ako

$$\nabla f_2(L, S) = \begin{pmatrix} -(M - (L + S)) \\ -(M - (L + S)) \end{pmatrix}.$$

Ďalej odvodíme konštantu  $\beta$  potrebnú k implementácii algoritmu:

$$\frac{1}{2} \|M - (L + S)\|_2^2 = \frac{1}{2} \left\| M - \begin{bmatrix} I & I \end{bmatrix} \begin{bmatrix} L \\ S \end{bmatrix} \right\|_2^2.$$

Ďalej

$$\| \begin{bmatrix} I & I \end{bmatrix} \| = \left\| \begin{bmatrix} I & I \end{bmatrix} \begin{bmatrix} I \\ I \end{bmatrix} \right\| = \|2I\| = 2\|I\| = 2.$$

Pre algoritmus použijeme  $\beta = 2$ . Podľa Algoritmu 1 zvolíme štartovací bod:

$$x_0 = \begin{bmatrix} M \\ 0 \end{bmatrix} = \begin{bmatrix} L_0 \\ S_0 \end{bmatrix},$$

ďalej zvolíme  $\epsilon = \frac{1}{2}$ ,  $\gamma = \frac{1}{2}$ .

Ďalej podľa štvrtého bodu Algoritmu 1 rozpišeme:

$$Y_L^n = L^n - \gamma(M - (L + S))$$

$$Y_S^n = S^n - \gamma(M - (L + S)).$$

V ďalšom kroku budú operovať proximálne operátory paralelne vedľa seba:

$$\begin{bmatrix} L^{n+1} \\ S^{n+1} \end{bmatrix} = \begin{bmatrix} L^n \\ S^n \end{bmatrix} + \gamma \left[ \text{prox}_{\gamma(\lambda_L \|\cdot\|_* + \lambda_S \|\cdot\|_1)} \begin{bmatrix} Y_L^n \\ Y_S^n \end{bmatrix} - \begin{bmatrix} L^n \\ S^n \end{bmatrix} \right],$$

pričom

$$\text{prox}_{\gamma(\lambda_L \|\cdot\|_* + \lambda_S \|\cdot\|_1)} \begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} \text{prox}_{\|\cdot\|_*}(v, \lambda_L) \\ \text{prox}_{\|\cdot\|_1}(w, \lambda_S) \end{bmatrix}.$$

Ako podmienku konvergencie sme si zvolili

$$\frac{\|(L_{k+1} + S_{k+1}) - (L_k + S_k)\|_F}{\|L_k + S_k\|_F} \leq \epsilon,$$

kde  $\epsilon = 10^{-7}$ .

## 7.3 IMPLEMENTÁCIA MODELOV V MATLABE

### 7.3.1 REÁLNE VIDEO

V tejto časti krátko vysvetlíme popis funkcií použitých v prostredí MATLAB, ktoré sa nachádzajú na priloženom CD.

Hlavný program je pomenovaný ako *hlavny\_program.m*.

#### 1. Načítanie videa

V prvom kroku je potrebné načítať video. V prostredí MATLAB 2012, na to slúži funkcia VideoReader. Načítať môžeme rovno zo zložky alebo definujeme cestu. Ďalej pracujeme s video ako so 4-D objektom. Parametre nami použitého videa sú:

šírka x výška = 144 x 176 pixelov

počet framov (snímok) = 210

2. V druhom kroku potrebujeme transformovať 4-D objekt do 3-D matice a konvertovať do formátu grayscale. V tomto formáte a o jednu dimenziu menej sa nám bude lepšie pracovať. Takáto výsledná matica je typu (144×176×210).
3. V treťom kroku potrebujeme transformovať maticu a hodnoty musia mať formát double. Tento formát je potrebný pre všetky výpočetné funkcie v Matlabe. K tomuto použijeme funkciu Reshape a dostaneme M - 2-D maticu typu (144×176, 210) = (25 344, 210).
4. Teraz už môžeme použiť jeden z implementovaných algoritmov. Tieto algoritmy sme vytvorili podľa hore uvedených odvodení a výpočtov, za použitia niektorých matlabovských funkcií, napr. pre výpočet SVD rozkladu.
5. Po skončení výpočtu je potreba zase výsledné matice L, S transformovať do 3-D matice, aby sme mohli vytvoriť a zapísať video.

## GRAFICKÉ VÝSLEDKY

Metóda	Minimalizácia	Podmienka konvergencie
ALM	$l_{\mu}(L,S,Y) = \ L\ _* + \lambda \ S\ _1 + \langle Y, M - L - S \rangle + \frac{\mu}{2} \ M - L - S\ _F^2$	$\ M - L - S\ _F \leq \delta \ M\ _F$
FB	$\operatorname{argmin} \frac{1}{2} \ M - (L + S)\ _2^2 + \lambda_L \ L\ _* + \lambda_S \ S\ _1$	$\frac{\ (L_{k+1} + S_{k+1}) - (L_k + S_k)\ _F}{\ L_k + S_k\ _F} \leq \epsilon$

Tabuľka 1. RPCA riešená PCP pomocou dvoch algoritmov

V tejto časti uvedieme a zhrnieme výsledky použitých algoritmov. Bohužiaľ kvalita videa o veľkosti 144×176 pixelov nie je príliš dobrá na prezentáciu .jpg obrázkov. Lepšie je rozdiel vidieť vo videu. Tieto videá sa nachádzajú na priloženom CD. Ako prvý sme použili ALM algoritmus s parametrom  $\lambda = \sqrt{\frac{1}{m}}$ . Tento výsledok je uložený v zložke PCP. V matici L sa ešte vyskytujú tiene, ktoré sme odstránili znížením parametru o polovicu. Tento výsledok je uložený v zložke PCP1.

Síce separácia bola úspešná a aj pozadie je už kvalitnejšie a čistejšie, táto separácia bola dvojnásobne náročnejšia počtom iterácií aj časovým intervalom. Je to spôsobené práve výpočtovo náročným SVD rozkladom. Tento rozklad sme sa snažili zrýchliť dvomi spôsobmi:

- Zobratím len  $k$  potrebných stĺpcov:

```
sigma_pom = diag(Sigma);
```

```
k = length(find(sigma_pom > 1/mu));
```

```
U = U(:,1:k); V = V(:,1:k); Sigma = diag(soft(sigma_pom(1:k),1/mu));
```

```
L = U*Sigma*V';
```

- Ďalej sme sa snažili zrýchliť výpočet zmenšením matice M a to tak, že sme rozdelili video na menšie časti, a počítali sme SVD rozklad a algoritmy na týchto pod častiach, na konci výpočtu sme zase matice spojili a vytvorili video. Táto funkcia je na CD pomenovaná ako *ExtractSubvideoFrames.m*. Tento výpočet má zmysel pri ALM metóde, ale aj pri väčšom rozlíšení videa. Skúsili sme separovať aj video o rozlíšení 240×320 pixelov. Takýto rozklad bez použitia rozdelenia videa trval viac ako 6 hodín, pri použití rozdelenia sa tento čas zmenšil na 4 hodiny.

FB algoritmus taktiež separoval video. Parametre boli zistené experimentálne. Hlavnou výhodou tohto algoritmu bol celkový čas len 607 sekúnd a 1122 iterácií. Výsledky sú ukázané na obrázku 4 a uložené na CD v zložke FB. Ďalej sme skúšali aj tu zmenšiť parameter  $\lambda_S$  na 0,1. Toto zníženie trochu viac zosvetlilo maticu S, prenikajú v nej aj iné objekty ako autá, ale zvýšilo čas výpočtu aj počet iterácií.

Tento algoritmus sme použili aj na separovanie zašumených dát. Do videa sme pridali Gaussovský šum so smerodajnou odchýlkou =20. Táto separácia taktiež bola úspešná pre L, avšak v matici S sa objavuje šum. Pochopiteľne sa zvýšil aj čas výpočtu a to až na 2 hodiny.

Metóda	ALM		FB	
Parametre	$\lambda = 1/\sqrt{m}$		$\lambda_L = 100 \lambda_S = 0,5$	
	Počet iterácií	Čas	Počet iterácií	Čas
Hodnoty	1995	1263 s	1122	607 s
Parametre	$\lambda = 1/2\sqrt{m}$		$\lambda_L = 100 \lambda_S = 0,1$	
	Počet iterácií	Čas	Počet iterácií	Čas
Hodnoty	3951	2453 s	3893	2102 s

Tabuľka 2. Porovnanie algoritmov

Z tabuľky 2 vidíme, že časovo je oveľa rýchlejší FB algoritmus pre zvolené parametre  $\lambda_L = 100, \lambda_S = 0,5$ . Aj prvotný rozklad videa bol kvalitnejší u FB algoritmu ako pri ALM so zvolenými parametrami podľa článku [5].

Nevýhodou práve FB algoritmu je experimentálne nastavenie parametrov, ktoré je potrebné najprv skúšať na menšej časti videa, zatiaľ čo pri ALM metóde už prvotné nastavenie poznáme a pri horšom výsledku stačí, ak budeme pomaly parameter znižovať.

Všetky výpočty boli počítané na desktopovom počítači s parametrami:

- Procesor Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz, 4200 MHz,
- počet jadier: 4
- počet logických procesorov: 8
- Nainštalovaná fyzická pamäť (RAM): 16,0 GB

## ALM ALGORITMUS



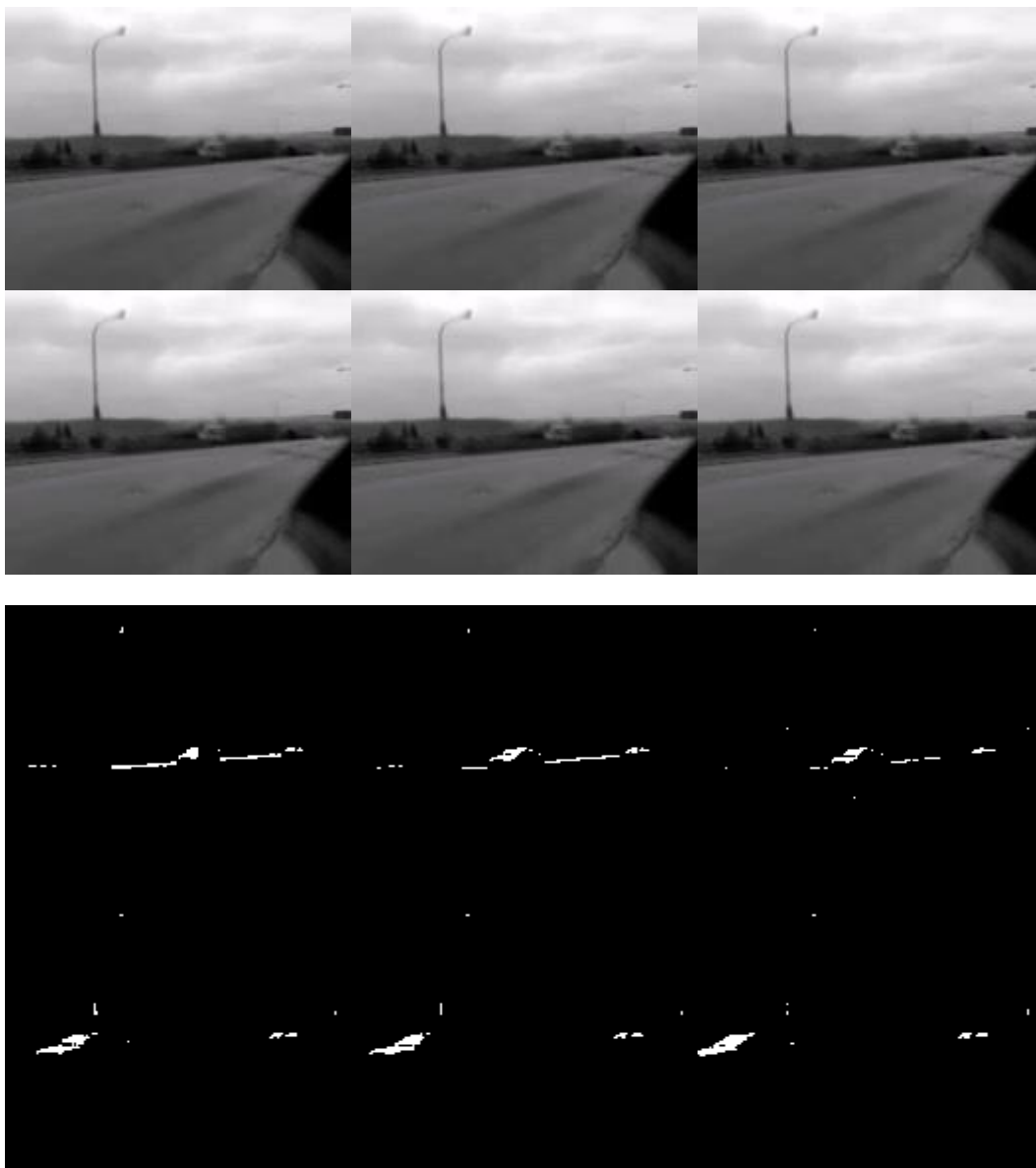
Obrázok 3. Požitie ALM algoritmu, prvé je originálne video, ktoré sa konvertujú do grayscale formátu. Matica  $L$ , dole  $S$ . Použité parametre  $\lambda = \sqrt{\frac{1}{m}}$ . Počet iterácií 1995, celkový čas 1263 sekúnd.

## FB ALGORITMUS



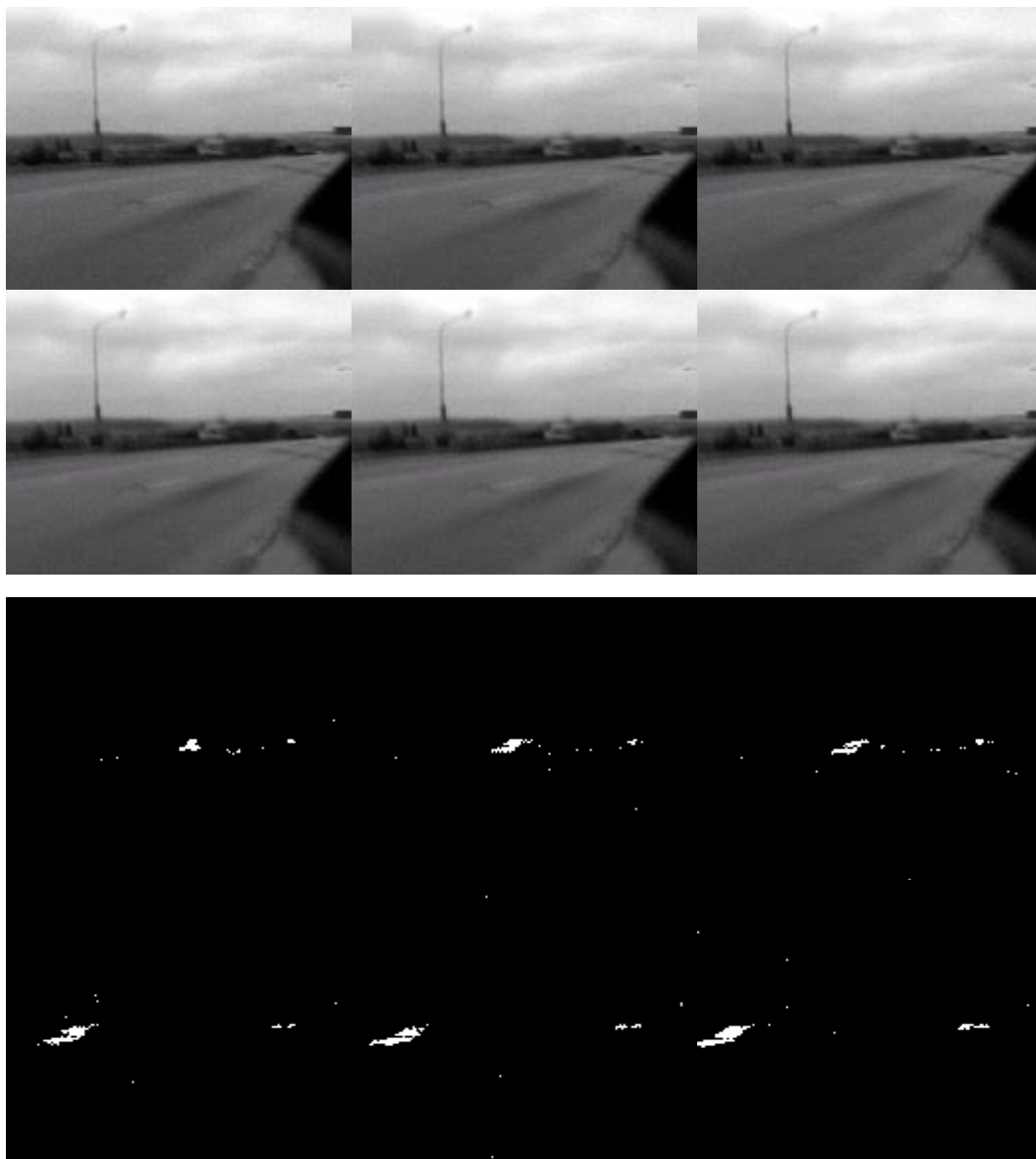
Obrázok 4. Požitie FB algoritmu, hore matica  $L$ , dole  $S$ . Použité parametre  $\lambda_L = 100$ ,  $\lambda_S = 0,5$ . Počet iterácií 1122, celkový čas 607 sekúnd.

## FB ALGORITHMUS



Obrázok 5. Požitie FB algoritmu, hore matica L, dole S. Použité parametre  $\lambda_L = 100$ ,  $\lambda_S = 0,1$ . Počet iterácií 3893, celkový čas 2101 sekúnd.

## FB ALGORITMUS S PRIDANÍM ŠUMU



Obrázok 6. Požitie FB algoritmu s pridaním šumu so smerodajnou odchýlkou 20, hore matica  $L$ , dole  $S$ . Použité parametre  $\lambda_L = 100$ ,  $\lambda_S = 0,5$ . Počet iterácií 1256, celkový čas 2261 sekúnd.

### 7.3.2 TESTOVACIE VIDEO

Testovacie video bolo vytvorené takisto v Matlabe. Pre tento účel je na CD uložený skript s názvom *test\_video.m*. Video je vytvorené jednoducho vložením jedného obrázku, ktorý bude stále ako statické pozadie a na ňom sa pohybuje červený bod.

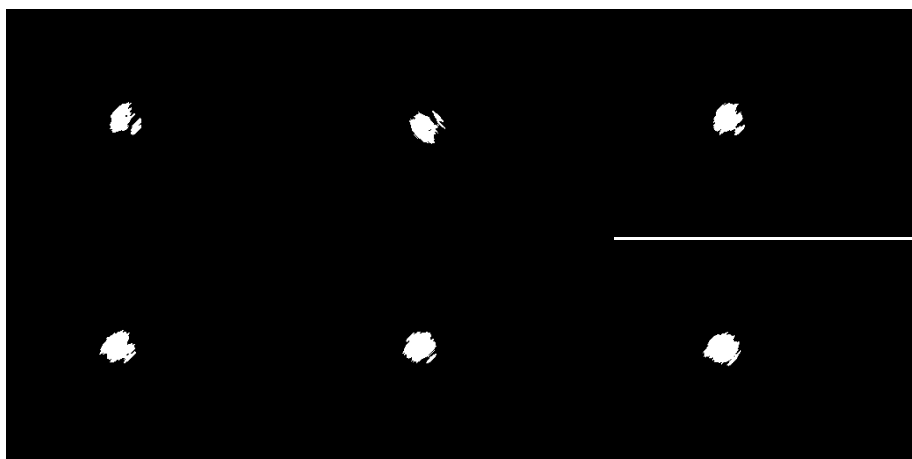
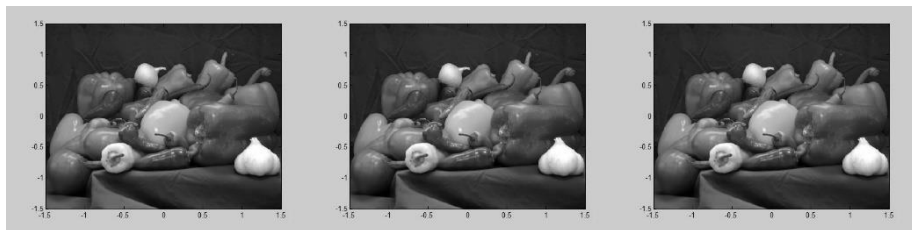
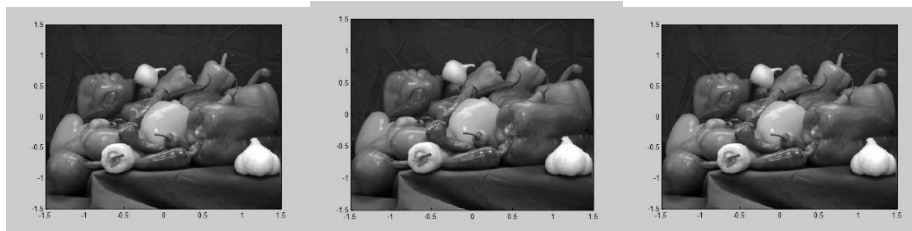
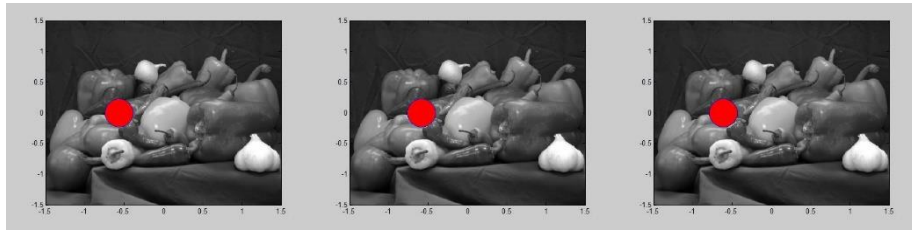
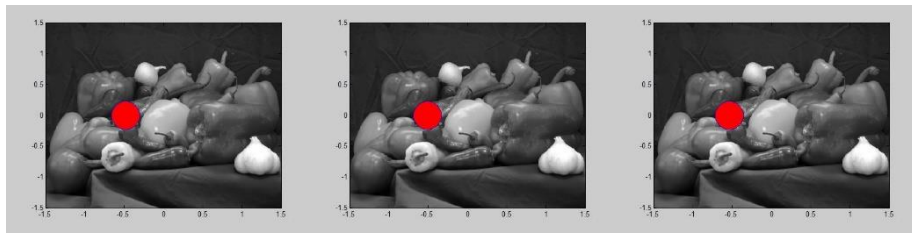
Avšak už pri vytváraní videa funkciou VideoWriter sme narazili na problém, a to nemožnosť zmeniť výšku a šírku videa v Matlabe. V tejto funkcii je prednastavená šírka x výška = 560 x 420 pixelov, počet framov (snímok) = 200, čo je pre náš značný problém. Ako sme vyššie spomínali, naše reálne video je veľkosti len 144 x 176 pixelov. Znamená to, že výpočet sa značne predĺžil.

Pre FB algoritmus s nastavením parametrov  $\lambda_L = 100$  ,  $\lambda_S = 0,5$  trval výpočet 5,5 hodiny a počet iterácií bol 3265. Samotná separácia sa vydarila, ale pohybujúci sa červený bod sa neukáže po celej trajektórii.

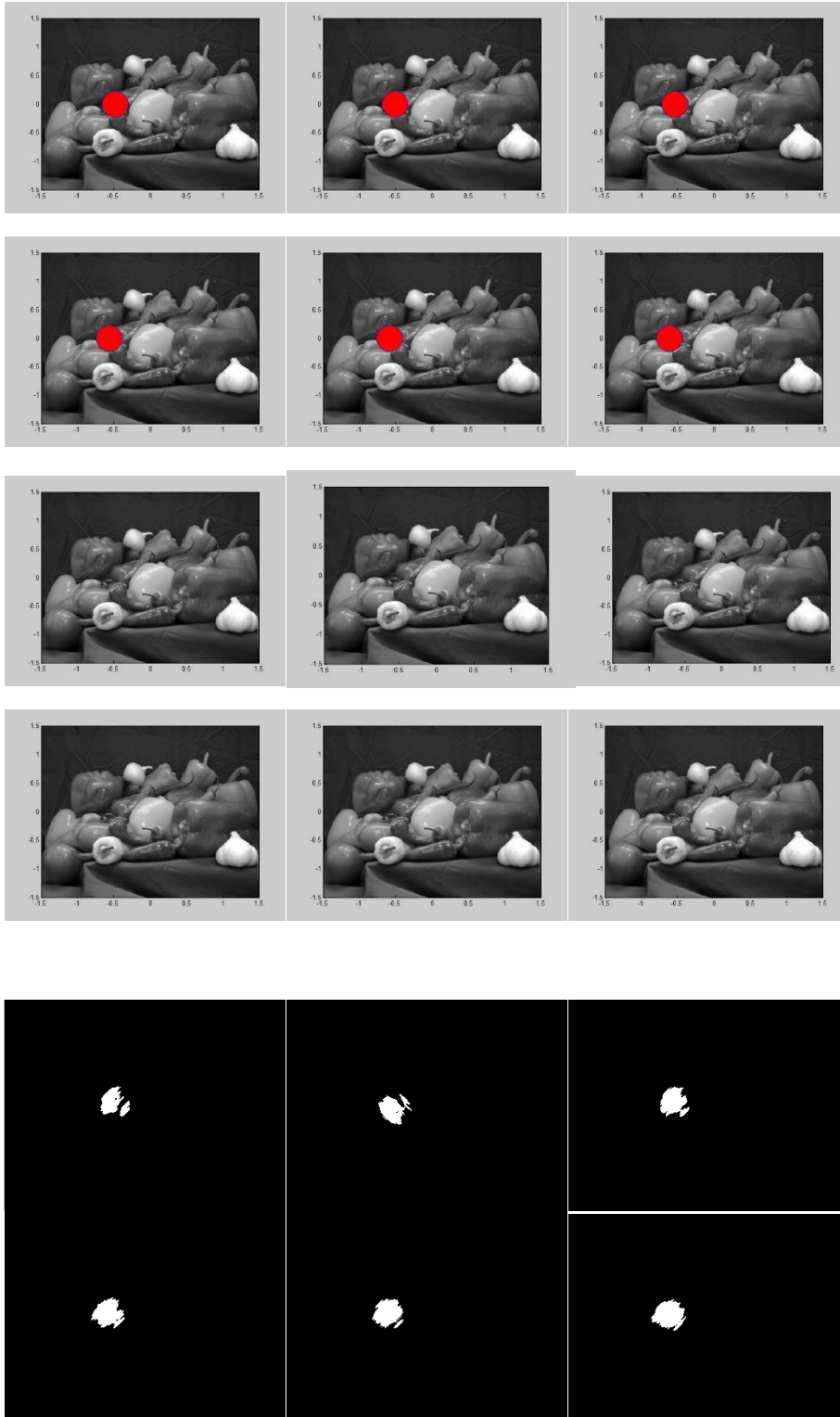
Pri ALM algoritme bol výsledok skoro rovnaký, okom snád' ani nie je rozoznateľný. Zaujímavé však je, že výpočet trval „len“ 3 hodiny 20 minút a počet iterácií dosiahol len 1786, pri skoro rovnakom výsledku.

Samozrejme by sa možno dosiahol lepší výsledok pri FB algoritme, pri experimentálnej úprave parametrov, ale za cenu zdĺhavého výpočtu.

# FB ALGORITHMUS



# ALM ALGORITHMUS



## 9 ZÁVER

Cieľom tejto diplomovej práce bolo popísanie a hlavne implementácia algoritmov na oddelenie pozadia a pohybujúcich sa objektov vo videosekvencii. V prvej kapitole sme sa zaoberali riedkymi reprezentáciami signálov. Tieto poznatky sme ďalej využili pri odvodení algoritmov. Druhá kapitola stručne popisuje analýzu hlavných komponentov. Na túto kapitolu ďalej nadväzuje robustná analýza hlavných komponentov, čo je pre nás najdôležitejší základ pre ďalší postup v práci.

Práve RPCA má veľa aplikácií v teórii signálov. My sme sa zaoberali oddelením pozadia a pohybujúcich sa objektov vo videosekvencii. Túto úlohu môžeme formulovať ako konvexný optimalizačný problém a nasadiť na ňu niekoľko algoritmov.

V prvom rade sme sa zaoberali proximálnymi algoritmami, konkrétne dopredne-spätným delením. Tento algoritmus sme implementovali v prostredí MATLAB. Ďalším algoritmom bola metóda rozšíreného Lagrangianu. Táto metóda je najznámejšou metódou spomedzi algoritmov založených na penalizačnej funkcii.

V poslednej kapitole sme zhrnuli poznatky a uviedli pár obrázkov získaných výpočtom. Celé videá sú dostupné v prílohe alebo na CD. Zaoberali sme sa aj zrýchlením výpočtu alebo zašumením dát.

## Literatúra

- [1] Bazaraa, M. S.; Sherali, H. D.; Shetty, C. M.: *Nonlinear programming: Theory and Algorithms*. 3rd ed. John Wiley & Sons, 2006, ISBN 04-714-8600-0.
- [2] Bertsekas, D. P.: *Constrained Optimization and Lagrange Multiplier Methods*. Massachusetts Institute of Technology, 1996, ISBN 1.886529-04-3.
- [3] Bouwmans, T.; Zahzah, El Hadi.: *Robust PCA via Principal Component Pursuit: A review for a comparative evaluation in video surveillance*. Elsevier, 2013, DOI: 10.1016/j.cviu.2013.11.009.
- [4] Boyd, S.; Vandenberghe, L.: *Convex Optimization*. Cambridge University Press, 2004, ISBN 978-0-521-83378-3.
- [5] Candes, E.J.; Li, X.; Ma, Y.; et al.: *Robust Principal Component Analysis?* J. ACM, vol. 58, č. 3, 2011, DOI: 10.1145/1970392.1970395. Dostupné z: <http://doi.acm.org/10.1145/1970392.1970395>
- [6] Čermák, L.: *Vybrané statě z numerických metod*. [elektronické skriptum], FSI VUT, Brno, 2012.
- [7] Daňková, M.: *Komprimované snímání v perfuzním zobrazování pomocí magnetické rezonance*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2014. 56 s. Vedoucí Mgr. Pavel Rajmic, Ph.D.
- [8] Combettes, P. L.; Pesquet, J.-C.: *Proximal Splitting Methods in Signal Processing*. Fixed-Point Algorithms for Inverse Problems in Science and Engineering. Springer, 2011.
- [9] Ganes, A.; Lin, Z.; Wright, J.: *Fast Algorithms for Recovering a Corrupted Low-Rank Matrix*. IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, 2009
- [10] Hrbáček, R.; Rajmic, P.; Veselý, V.; Špiřík, J.: *Řídké reprezentace signálů: úvod do problematiky*. *Elektrorevue - Internetový časopis*, 2011, č. 50, s. 1-10. ISSN:1213- 1539.
- [11] Lin, Z.; Chen, M.; Ma, Y.: *The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices*. arXiv:1009.5055 [math.OA], 2010, DOI:10.1016/j.jsb.2012.10.010.
- [12] Parikh, N.; Boyd, S.: *Proximal Algorithms*. *Foundations and Trends in Optimization*, 2013, č.3, s. 123{231.
- [13] Popela, P.: *Nonlinear Programming*. 2002.
- [14] Rajmic, P.: *Řídké a nízkohodnostní reprezentace signálů s aplikacemi*. Habilitační práce, Vysoké učení technické v Brně, 2014.
- [15] Shlens, J.: *A Tutorial on Principal Component Analysis*. 2014

- [16] Vanderbei, R.J.: *Linear Programming: Foundations and Extensions*. 2nd ed., 2001.
- [17] Wikipedie - internetová encyklopedie: Differentiable function 2016 [online]. 2014  
[cit. 2016-05-22]. Dostupné z: [https://en.wikipedia.org/wiki/Differentiable\\_function](https://en.wikipedia.org/wiki/Differentiable_function)
- [18] Wright, J.; Peng, Y.; Ma, Y.: *Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices by Convex Optimization*.
- [19] Zlatoš, P.: *Lineárna algebra a geometria*. Bratislava : Marenčin PT, 2011.  
Dostupné z: [http://thales.doa.fmph.uniba.sk/zlatos/la/LAG\\_A4.pdf](http://thales.doa.fmph.uniba.sk/zlatos/la/LAG_A4.pdf)