

# Užití diskrétní wavelet transformace v potlačování rušivých složek biomedicínských obrazů

Zdeněk Měřínský, Andrea Gavlasová, Eva Hošťálková

Vysoká škola chemicko-technologická v Praze, Ústav počítačové a řídicí techniky

## Abstrakt

Diskrétní wavelet transformace (DWT) nalézá využití při potlačování šumu v obrazu. Proces transformace obsahuje dekompozici, kdy z původního obrazu vzniká sada aproximačních a detailních koeficientů. Opačný postup se nazývá rekonstrukce, kdy při nezměněných hodnotách koeficientů vznikne původní obraz. Tato práce popisuje dvě konkrétní wavelet transformace – Haarovu a Daubechies. Potlačení šumu je realizováno prahováním detailních wavelet koeficientů před následnou rekonstrukcí. Druhá část práce je proto zaměřena na studium prahování – rozsah, tvar prahovací funkce a metody výpočtu prahu. Kvalita potlačení šumu byla posouzena pomocí PSNR kritéria (Peak signal to noise ratio). Ke studiu metod potlačování šumu byly použity biomedicínské obrazy MR.

## Poděkování

Biomedicínské obrazy MR byly laskavě poskytnuty MUDr. Oldřichem Vyšatou z neurologického centra v Rychnově nad Kněžnou. Tato práce byla podpořena výzkumným grantem č. MSM 6046137306.

## 1 Diskrétní wavelet transformace

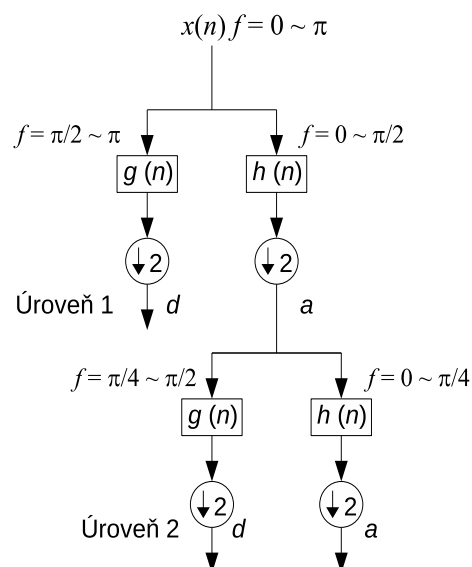
Wavelet transformace vznikla pro potřebu analýzy nestacionárních signálů, kde nestačí použití *Fourierovy transformace*. Tato transformace využívá různě dlouhá okénka tvaru wavelet funkce, která nám poskytují informace o tom, kdy a kde se vyskytují různé frekvenční komponenty (složky) signálu. Okno se posouvá podél signálu a pro každou pozici se vypočítá korelace s dilatovanou vlnkou. Tento proces se opakuje vždy s nepatrně delším okénkem. Výsledkem je sada časově-frekvenčních reprezentací signálu v různých rozlišeních [3]. Výše popsaným postupem získáme koeficienty tzv. **spojité wavelet transformace**, která je poměrně výpočetně náročná, protože obsahuje *redundantní* (nadbytečnou) informaci o dekomponovaném signálu. Výpočetní náročnost je ale v některých aplikacích vyvážena invariantností této transformace vůči posunu.

Kritickým navzorkováním spojité wavelet transformace se dospělo k **diskrétní wavelet transformaci** (DWT), která je vhodná pro aplikace ve výpočetní technice. DWT využívá techniku dekompozice diskrétního signálu zvanou *subband coding* [6].

DWT využívá dvě sady funkcí, jedna se nazývá *scaling* (měřítková) a druhá *wavelet* (vlnková). Dohromady tvoří **banku filtru**. Wavelet funkce souvisí s vysokofrekvenčním filtrem  $g(n)$  a scaling s nízkofrekvenčním filtrem  $h(n)$ . Konvolucí signálu s těmito filtry získáme aproximační (*scaling*) koeficienty  $a$  a detailní (*wavelet*) koeficienty  $d$ . Došlo k podvzorkování dvěma, měřítko je tedy zdvojené, ale rozlišení se nezměnilo.

$$a_1(k) = \sum_n x(n)h(2k - n) \quad (1a)$$

$$d_1(k) = \sum_n x(n)g(2k - n) \quad (1b)$$



Obrázek 1: Znázornění wavelet dekompo-

Tento krok se dá se stejně navrženými filtry opět aplikovat na získané aproximační koeficienty  $a$ , provede se další úroveň dekompozice. Se zvyšující se úrovní dekompozice roste také frekvenční rozlišení a snižuje se časové rozlišení.

## 1.1 Haarova transformace

### 1.1.1 Dekompozice

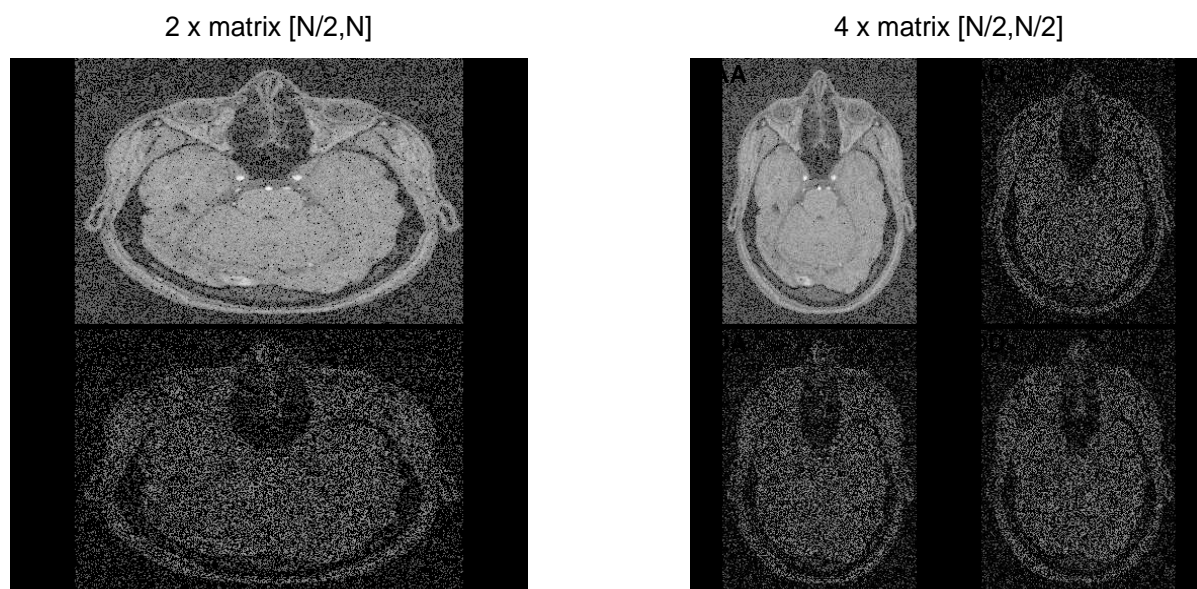
Haarova transformace je nejjednodušší typ diskretní wavelet transformace. Vstupní data jsou tvořeny posloupností  $\{x(i)\}_{i=0}^{N-1}$ . V každém kroku jsou počítány aproximační a detailní koeficienty ze dvojice po sobě jdoucích čísel posloupnosti [4].

$$a_i = \frac{x_{2i} + x_{2i+1}}{\sqrt{2}}, \quad i = 0, 1, \dots, \frac{N}{2} - 1 \quad (2a)$$

$$d_i = \frac{x_{2i} - x_{2i+1}}{\sqrt{2}}, \quad i = 0, 1, \dots, \frac{N}{2} - 1 \quad (2b)$$

Pro návrh algoritmu je vhodnější si výše uvedenou definici dekompozice přepsat v maticovém tvaru. Signál je zpracováván jako vektor vždy po dvou hodnotách. Matice  $N \times N$  obsahuje transformační koeficienty. Umístění ve sloupcích odpovídá příslušným vzorkům signálu, které jsou zpracovány. Po dvou řádcích se koeficienty posouvají o dva sloupce, zbylé hodnoty jsou nuly.

$$\begin{bmatrix} a_0 \\ d_0 \\ a_1 \\ d_1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (3)$$



Obrázek 2: Haarova dekompozice do první úrovně

## 1.1.2 Rekonstrukce

Rekonstrukce je zpětná transformace, při které lze získat nezměněná vstupní data. Perfektní rekonstrukce je provedena, pokud aproximační ani detailní koeficienty nejsou modifikovány.

$$x_{2i} = \frac{a_i + d_i}{\sqrt{2}} \quad (4a)$$

$$x_{2i+1} = \frac{a_i - d_i}{\sqrt{2}} \quad (4b)$$

Obdobně jako dekompozice i rekonstrukce lze provést v maticovém tvaru.

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} a_0 \\ d_0 \\ a_1 \\ d_1 \end{bmatrix} \quad (5)$$

## 1.2 Daubechies transformace

### 1.2.1 Dekompozice

Daubechies transformace zpracovává čtyři hodnoty signálu při každém posunutí filtru. Tomu odpovídají čtyři transformační scaling a wavelet koeficienty.

$$h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}}, \quad h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}}, \quad h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}}, \quad h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}} \quad (6)$$

$$g_0 = h_3, \quad g_1 = -h_2, \quad g_2 = h_1, \quad g_3 = -h_0$$

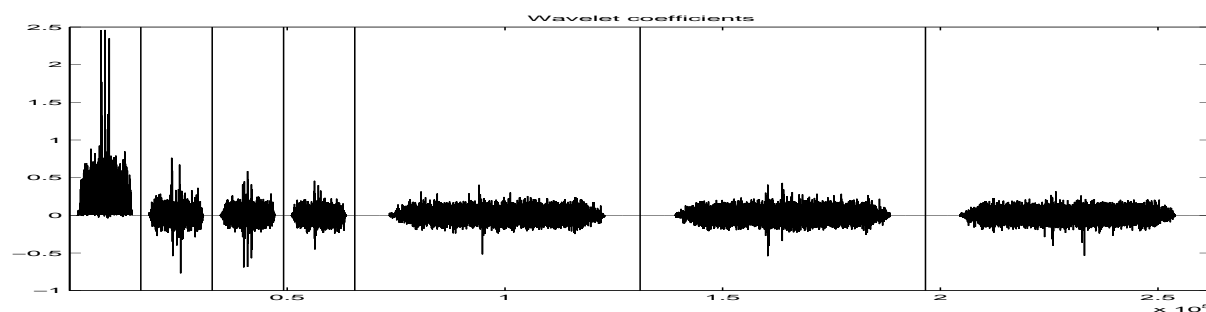
Nyní je možno vypočítat aproximační a detailní koeficienty.

$$a_i = h_0 x_{2i} + h_1 x_{2i+1} + h_2 x_{2i+2} + h_3 x_{2i+3} \quad (7a)$$

$$d_i = g_0 x_{2i} + g_1 x_{2i+1} + g_2 x_{2i+2} + g_3 x_{2i+3} \quad (7b)$$

Tento postup je možno zapsat v maticovém tvaru obdobně jako u Haarovy transformace.

$$\begin{bmatrix} a_0 \\ d_0 \\ a_1 \\ d_1 \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 \\ g_0 & g_1 & g_2 & g_3 \\ 0 & 0 & h_0 & h_1 \\ 0 & 0 & g_0 & g_1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (8)$$



Obrázek 3: Znáznornění aproximačních a detailních koeficientů po Daubechies dekompozici do 2. úrovně

## 1.2.2 Rekonstrukce

Na rekonstrukci budou použity stejné koeficienty  $h$  a  $g$  jako při dekompozici.

$$x_{2i} = h_2 a_i + g_2 d_i + h_0 a_{i+1} + g_0 d_{i+1} \quad (9a)$$

$$x_{2i+1} = h_3 a_i + g_3 d_i + h_1 a_{i+1} + g_1 d_{i+1} \quad (9b)$$

Tento vztah lze také zapsat v maticovém tvaru.

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} h_2 & g_2 & h_0 & g_0 \\ h_3 & g_3 & h_1 & g_1 \\ 0 & 0 & h_2 & g_2 \\ 0 & 0 & h_3 & g_3 \end{bmatrix} \begin{bmatrix} a_0 \\ d_0 \\ a_1 \\ d_1 \end{bmatrix} \quad (10)$$

## 2 Prahování wavelet koeficientů

Prahování jsou matematické úpravy, které se využívají pro segmentaci obrazů či klasifikaci časových řad [2]. Prahování má také využití ve spojení s wavelet transformací. Signál či obraz je dekomponován do určité úrovně a poté jsou všechny detailní koeficienty vhodně prahovány. Po následné zpětné rekonstrukci může dojít k potlačení šumu. Nevýhodou ovšem může být ztráta informací v obraze (např. hran).

### 2.1 Rozsah prahování

Před provedením samotného prahování je třeba určit, které wavelet koeficienty budou prahovány. Prahují se detailní  $d$  koeficienty ze všech úrovní rozkladu. Z hlediska rozsahu prahovaných hodnot rozlišujeme několik typů prahování. V této práci byly použity dva typy prahování.

- **Lokální:** Každá skupina (*subband*) detailních koeficientů je prahována samostatně s různě velkou hodnotou prahu.
- **Globalní:** Všechny koeficienty určené k prahování se prahují najednou s jednou hodnotou prahu.

### 2.2 Prahovací funkce

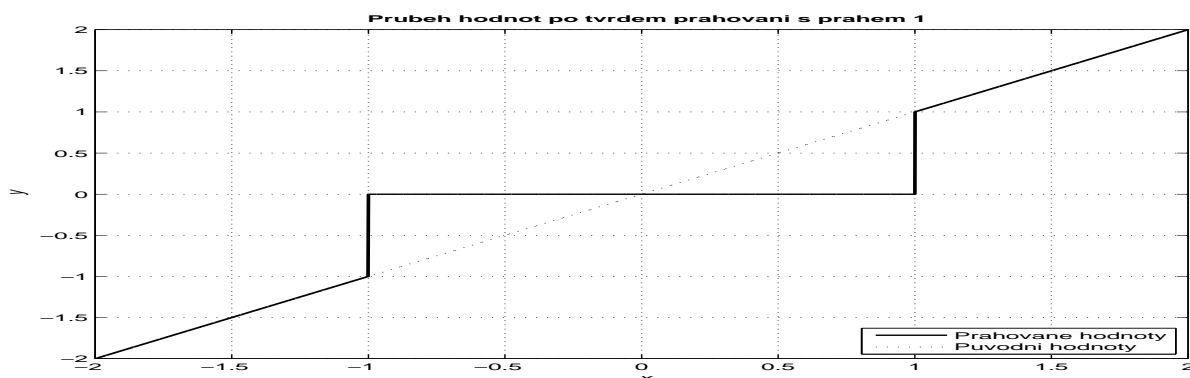
Prahovací funkce  $y = f(x)$  provádí vlastní prahování předem vybraných hodnot. Nové hodnoty  $y$  jsou následně použity k wavelet rekonstrukci. Rozlišujeme dvě základní prahovací funkce [7].

- **Hard thresholding:** Tzv. tvrdé prahování, hodnoty větší než hodnota prahu  $t$  jsou ponechány, ostatní jsou vynulovány.

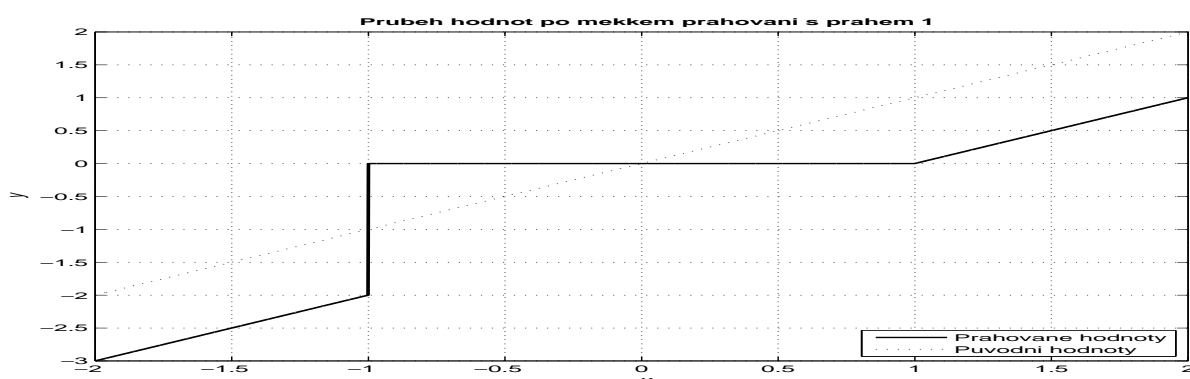
$$y(n) = \begin{cases} x(n) & \text{pro } |x(n)| > t \\ 0 & \text{pro } |x(n)| \leq t \end{cases} \quad (11)$$

- **Soft thresholding:** Tzv. měkké prahování, hodnoty větší než práh jsou nahrazeny hodnotami vypočtenými podle definice, ostatní hodnoty jsou vynulovány.

$$y(n) = \begin{cases} \text{sgn } x(n)(|x(n)| - t) & \text{pro } |x(n)| > t \\ 0 & \text{pro } |x(n)| \leq t \end{cases} \quad (12)$$



Obrázek 4: Původní hodnoty a hodnoty po tvrdém prahování s prahem 1



Obrázek 5: Původní hodnoty a hodnoty po měkkém prahování s prahem 1

### 2.3 Výpočet hodnoty prahu

Hodnotu prahu  $t$  můžeme odhadnout, avšak poté provedené prahování nemusí dávat dobré výsledky. U následné rekonstrukce nemusí dojít k potlačení šumu. Bylo navrženo několik postupů, jak co nejlépe hodnotu prahu vypočítat. Některé ale přesto využívají empiricky zjištěné konstanty. Pro účely této práce byl užity dva výpočty prahu, které pracují se statistickými charakteristikami prahovaných koeficientů.

- **Odhad dle Tribouleyho**

$$t = \sqrt{2\sigma_y^2 \ln M_y} \quad (13)$$

Číslo  $\sigma_{ij}^2$  představuje rozptyl koeficientů určených k prahování,  $M_{ij}$  je počet nenulových koeficientů [7]. Z této definice je vidět, že při lokálním prahování je vypočtená hodnota prahu jiná pro každé zvlášť prahované koeficienty.

- **Normal shrink**

$$t = \frac{\beta\sigma^2}{\sigma_y}, \quad \beta = \sqrt{\ln \frac{L_k}{J}}, \quad \sigma^2 = \left( \frac{\text{med } |y_{ij}|}{0.6745} \right)^2 \quad (14)$$

Číslo  $\sigma_y$  představuje směrodatnou odchylku koeficientů, které se budou prahovat.  $L_k$  je jejich počet a  $J$  značí počet dekompozicí, po kterých příslušné detailní wavelet koeficienty vznikly. Normal shrink je založen na *MAD odhadu rozptylu aditivního šumu*, v definici je označen  $\sigma^2$ . MAD odhad se počítá z mediánu absolutních hodnot diagonálních detailních koeficientů z 1. úrovně rozkladu (značí se  $y_{ij}$ ) [5]. Tyto koeficienty obsahují nejvyšší frekvence a nesou tedy nejvíce šumu a nejméně obrazové informace. Metoda normal shrink je navržena pro lokální prahování.

### 3 Hodnocení výsledků

#### 3.1 Kritérium hodnocení

Protože prahování koeficientů a následná rekonstrukce poskytuje různé výsledky, je důležité porovnat rekonstruované obrazy. K porovnání lze využít **peak signal to noise ratio** (zkracováno jako PSNR), což je poměr mezi maximální hodnotou pixelu nezašuměného obrazu  $I$  ( $MAX_I$ ) velikosti  $M \times N$  a hodnotou šumu v obrazu po rekonstrukci [1]. Šum v obrazu po rekonstrukci je také možno chápat jako kvalitu odstranění šumu. Číselně je tato kvalita vyjádřena pomocí středního součtu čtverců chyb (angl. **mean square error**) mezi nezašuměným obrazem  $I$  a obrazem po rekonstrukci  $K$ .

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \|I(i, j) - K(i, j)\|^2 \quad (15)$$

Hodnota  $MSE$  je využita pro výpočet  $PSNR$  (definuje se v logaritmické stupnici).

$$PSNR = 10 \log \frac{MAX_I^2}{MSE} = 20 \log \frac{MAX_I}{\sqrt{MSE}} \quad (16)$$

#### 3.2 Porovnání některých hodnot PSNR

Následující tabulky shrnují některá PSNR kritéria získaná při rekonstrukci různých obrazů za použití studovaných metod wavelet transformace a prahování. Pro lepší porovnání byla také u každého obrazu vypočtena hodnota PSNR mezi obrazem bez šumu a s nepotačeným šumem. Tato hodnota bude nejmenší, proto čím větší PSNR u daného obrazu bude, tím byl více potlačen šum.

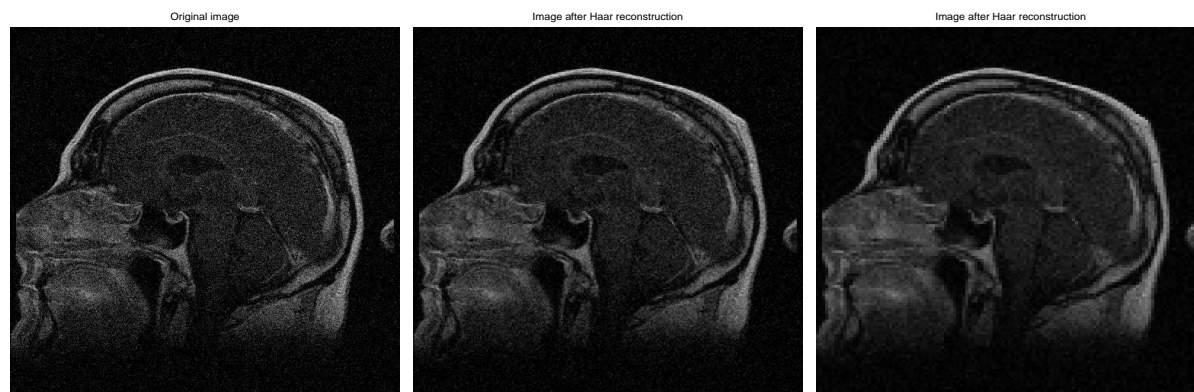
Málo zašuměný obrázek				
Výpočet prahu	Rozsah prahování	Prahovací funkce	Haarova transformace	
			Úroveň 1	Úroveň 2
Tribouley	Global	Soft	24.6675	24.6892
Tribouley	Local	Hard	24.6651	25.1792
Normal shrink	Local	Hard	23.0247	23.3493
Normal shrink	Local	Soft	23.0247	24.9948
Bez prahování			21.0406	

Tabulka 1: Hodnoty  $PSNR$  vypočtené pro různé kombinace wavelet transformace a prahování, obraz měl přidáný gaussovský bílý šum s  $m = -0.05$  a  $v = 0.01$

Více zašuměný obrázek				
Výpočet prahu	Rozsah prahování	Prahovací funkce	Daubechies transformace	
			Úroveň 1	Úroveň 2
Tribouley	Global	Soft	19.7768	17.4956
Tribouley	Local	Hard	19.7731	17.4764
Normal shrink	Local	Hard	19.2067	16.9502
Normal shrink	Local	Soft	19.2067	17.4527
Bez prahování			15.0667	

Tabulka 2: Hodnoty  $PSNR$  vypočtené pro různé kombinace wavelet transformace a prahování, obraz měl přidáný gaussovský bílý šum s  $m = 0$  a  $v = 0.02$

### 3.3 Ukázky výsledků



(a) Původní obrázek

(b) Obrázek po Haarovo rekonstrukci z 1.úrovně, Tribouleyho ze 2.úrovně, výpočet prahu Normal výpočet prahu, globální prahování shrink, lokální prahování funkcí hard thresholding

(c) Obrázek po Haarovo rekonstrukci z 2.úrovně, výpočet prahu Normal výpočet prahu, globální prahování shrink, lokální prahování funkcí soft thresholding

Obrázek 6: Obrázek s přidaným gaussovským bílým šumem ( $m = -0.05$  a  $v = 0.01$ ) a obrázky po Haarovo rekonstrukci s využitím různých metod prahování

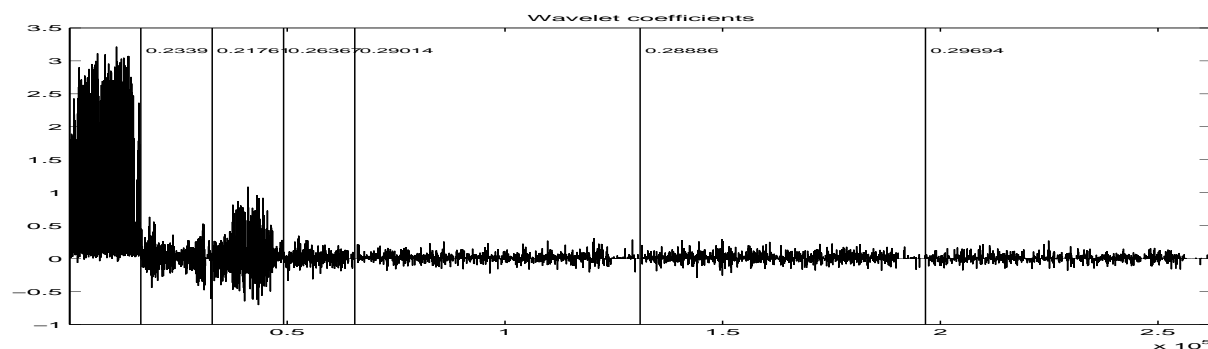


(a) Původní obrázek

(b) Obrázek po Daubechies rekonstrukci z 1.úrovně, výpočet prahu 2.úrovně, výpočet prahu Normal Normal shrink, lokální prahování funkcí hard thresholding

(c) Obrázek po rekonstrukci ze 2.úrovně, výpočet prahu Normal Normal shrink, lokální prahování funkcí soft thresholding

Obrázek 7: Obrázek s přidaným gaussovským bílým šumem ( $m = 0$  a  $v = 0.02$ ) a obrázky po Daubechies rekonstrukci s využitím různých metod prahování



Obrázek 8: Prahované wavelet koeficienty vzniklé po Daubechies dekompozici obrázku 7 (a) do 2. úrovně, bylo využito lokální prahování funkcí soft thresholding a práh byl vypočítán pomocí Normal shrink. U jednotlivých skupin koeficientů je uvedena hodnota vypočteného prahu

## 4 Závěr

Cílem naší práce bylo potlačit šum v obrazu pomocí diskrétní wavelet transformace a prahování. Ze studovaných transformací, metod prahování a způsobů výpočtu jsme získali různé výsledky. Aby bylo možné lépe posoudit, která kombinace je nejlepší, bylo použito kritérium *PSNR*. Pro porovnání bylo také spočteno *PSNR* obrazu bez šumu. Čím je toto kritérium větší, tím více byl v obrazu potlačen šum.

Pro hodnocení kvality ale nestačí pouze porovnat hodnoty *PSNR*, je třeba také zjistit, jak vypadá výsledný obrázek po rekonstrukci. Obecně větší potlačení šumu a vyšší *PSNR* zaručuje Haarova transformace, globální prahování a rekonstrukce ze 2. úrovně. Rekonstrukce ze 2. úrovně ale přináší nevýhody v podobě poničených hran výsledného obrazu, toto poničení lze zmírnit použitím prahovací funkce soft thresholding. Při rekonstrukcích z 1. úrovně nemá na *PSNR* vliv hard či soft thresholding.

Lépsi výsledky poskytly obrázky s menším množstvím přidaného šumu. Se zašuměním obrázku souvisí také použití studovaných metod výpočtu prahu. Tribouleyho metoda využívá počet nenulových koeficientů v prahovaných koeficientech. Čím je více nenulových koeficientů, tím větší je hodnota prahu a více koeficientů po prahování je rovno nule. Obrázek s menším množstvím přidaného šumu nemá tolik nenulových koeficientů jako obrázek s větším množstvím přidaného šumu. Při našem experimentu se ale v těchto případech zdála hodnota prahu příliš vysoká a potlačeny tak byly téměř všechny detailní koeficienty včetně těch nesoucích obrazovou informaci. Použití Tribouleyho metody je tedy vhodné hlavně pro málo zašuměné obrázky. Naproti tomu metoda Normal shrink poskytla dobré výsledky i u více zašuměných obrázků. Využívá totiž rozptyl wavelet koeficientů nesoucích nejvíce šumu. Pokud obsahuje obrázek málo šumu, metoda Normal shrink odhadne příliš malý práh, detailní koeficienty nejsou výrazně narahovány. Výsledný obraz se tedy podobá spíše původnímu zašuměnému obrazu.



## Reference

- [1] Wikipedia - the free encyclopedia,  
<http://en.wikipedia.org>. Internet.
- [2] B. S. Morse. Lecture 4: Thresholding,  
[http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/MORSE/threshold.pdf](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/threshold.pdf).  
Provo, UT 2002.
- [3] C. Valens. A Really Friendly Guide to Wavelets,  
<http://pagesperso-orange.fr/polyvalens/clemens/wavelets/wavelets.html>. Internet, 2004.
- [4] I. Kaplan. A Linear Algebra View of the Wavelet Transform,  
[http://www.bearcave.com/misl/misl\\_tech/wavelets/matrix/index.html](http://www.bearcave.com/misl/misl_tech/wavelets/matrix/index.html). Internet, 2002.
- [5] L. Kaur, S. Gupta, R. C. Chauhan. Image Denoising using Wavelet Thresholding. Punjab,  
India 2002.
- [6] R. Polikar. The Wavelet Tutorial,  
<http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html>. Internet, 2001.
- [7] W. Härdle, G. Kerkycharian, D. Picard, A. Tsybakov. Wavelets, Approximation and Sta-  
tistical Applications,  
<http://www.quantlet.com/mdstat/scripts/wav/html/index.html>. Internet, 2005.