



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**ANALÝZA ZABEZPEČENÍ FORMÁTU  
JPEG 2000 PROTI CHYBÁM**

ERROR RESILIENCE ANALYSIS FOR JPEG 2000

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**MAREK KOVALČÍK**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. DAVID BAŘINA, Ph.D.**

BRNO 2019

## Zadání bakalářské práce



21822

Student: **Kovalčík Marek**  
Program: Informační technologie  
Název: **Analýza zabezpečení formátu JPEG 2000 proti chybám**  
**Error Resilience Analysis for JPEG 2000**  
Kategorie: Zpracování obrazu

Zadání:

1. Seznamte se s kompresí obrazu. Zaměřte se na formát JPEG 2000.
2. Prostudujte jednotlivé parametry komprese obrazu pomocí JPEG 2000. Zaměřte se na parametry, které ovlivňují odolnost vůči poškození dat.
3. Srovnajte jednotlivé strategie zabezpečení dat ve formátu JPEG 2000. Tento bod zahrnuje tvorbu datové sady, její kompresi s různými parametry, cílené poškození dat, a nakonec srovnání s původními daty pomocí metriky PSNR.
4. Klady a zápory jednotlivých nastavení diskutujte.

Literatura:

- JPEG2000 Image Compression Fundamentals, Standards and Practice, by David Taubman and Michael Marcellin (ISBN-13: 978-0792375197)
- The JPEG 2000 Suite, by Peter Schelkens, Athanassios Skodras, Touradj Ebrahimi (ISBN-13: 978-0470721476)
- David Taubman, Michael Marcellin, JPEG2000: standard for interactive imaging, Proceedings of the IEEE, volume 90, issue 8, pp 1336 - 1357, IEEE, 2002

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Bařina David, Ing., Ph.D.**  
Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.  
Datum zadání: 1. listopadu 2018  
Datum odevzdání: 15. května 2019  
Datum schválení: 1. listopadu 2018

## Abstrakt

Tato práce se zabývá novodobým kompresním formátem JPEG 2000. Je zde analyzován vliv použití zabezpečovacích mechanismů při kompresi obrazu s různými nastaveními. Je zde zkoumáno, jaký vliv má použití vkládání značek pomáhajících při rekonstrukci poškozeného obrazu nebo použití kompresních módů zlepšujících odolnost proti chybám. Kvalita obrazů je vyhodnocována metrikou PSNR, která zjišťuje podobnost komprimovaného a referenčního souboru. Přidání určitých značek do datového toku nebo použití určitých kompresních módů by mělo pomoci se zabezpečením JPEG 2000 souboru proti poškození při rekonstrukci obrazu. Pro testování této hypotézy byl vytvořen model, který náhodně poškozoval komprimovaný soubor a hodnotil dekomprimované obrazy. Pro práci byla využívána knihovna Kakadu, která poskytuje efektivní práci s formátem JPEG 2000. Součástí práce je datová sada, na které probíhaly experimenty, a která se skládá z různorodých fotografií v nekomprimovaném formátu PPM v menších, ale také vyšších rozlišeních. Výsledkem této práce je zjištění, jaké nastavení komprese použít pro jakou skupinu obrazů, aby byla komprese efektivní a odolnost proti chybám co nejlepší. Závěr práce je věnován srovnání odolnosti proti poškození formátu JPEG 2000 a CCSDS 122.0.

## Abstract

The aim of this thesis is to analyze modern image compression format of JPEG 2000. It analyzes the effect of error resilience mechanisms on image compression with different settings. The impact of using tag embedding to help repair damaged images or using compression modes to improve error resilience is examined here. Quality is evaluated by the PSNR metric that detects the similarity of compressed and reference file. Adding certain tags to the data stream or using certain compression modes should help secure the JPEG 2000 file against image reconstruction damage. To test this hypothesis, there was created a model that accidentally damage the compressed file and evaluate decompressed images. The Kakadu library, which provides efficient work with the JPEG 2000 format, is used for the work. The experimental data set consists of various photographs in uncompressed PPM format in smaller but also in higher resolutions. The result of this work is to find out which compression settings to use for which group of images to make the compression efficient and secure the best. The end of this thesis is devoted to comparison of error resilience of JPEG 2000 and CCSDS 122.0.

## Klíčová slova

JPEG 2000, odolnost proti chybám, komprese obrazu, obrazový formát, ztrátová komprese, kakadu

## Keywords

JPEG 2000, error resilience, image compression, image format, lossy compression, kakadu

## Citace

KOVALČÍK, Marek. *Analýza zabezpečení formátu JPEG 2000 proti chybám*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. David Bařina, Ph.D.

# Analýza zabezpečení formátu JPEG 2000 proti chybám

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Davida Bařiny Ph.D. a Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Marek Kovalčík

13. května 2019

## Poděkování

Rád bych poděkoval vedoucímu práce doktoru Davidu Bařinovi za vstřícný přístup a užitečné rady při vypracování práce.

# Obsah

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Úvod</b>   | <b>2</b>  |
| <b>2</b> | <b>Komprese obrazu</b>                                      | <b>3</b>  |
| 2.1      | Ztrátová komprese . . . . .                                 | 3         |
| 2.2      | JPEG 2000 . . . . .   | 4         |
| 2.3      | Knihovna Kakadu . . . . .                                   | 12        |
| 2.4      | Knihovna OpenJPEG . . . . .                                 | 13        |
| <b>3</b> | <b>Analýza zabezpečení JPEG 2000 pro ztrátovou kompresi</b> | <b>15</b> |
| 3.1      | Model poškození . . . . .                                   | 15        |
| 3.2      | Metrika PSNR . . . . .                                      | 16        |
| 3.3      | Datová sada . . . . .                                       | 17        |
| 3.4      | Nevhodnost použití knihovny OpenJPEG . . . . .              | 18        |
| 3.5      | Postup provádění experimentů . . . . .                      | 19        |
| 3.6      | Analýza zabezpečení pomocí přidávaných značek . . . . .     | 20        |
| 3.7      | Analýza zabezpečení při použití módů . . . . .              | 22        |
| 3.8      | Analýza zabezpečení při použití značek i módů . . . . .     | 24        |
| 3.9      | Zabezpečení a režie v souvislosti . . . . .                 | 27        |
| 3.10     | Srovnání bezpečnosti JPEG 2000 a CCSDS 122.0 . . . . .      | 29        |
| <b>4</b> | <b>Závěr</b>  | <b>32</b> |
|          | <b>Literatura</b>   | <b>33</b> |
| <b>A</b> | <b>Obsah přiloženého paměťového média</b>                   | <b>34</b> |

# Kapitola 1

## Úvod

Tato práce se zabývá účinky ztrátové komprese formátu JPEG 2000 a jejím zabezpečení proti chybám. Chybami se v tomto kontextu rozumí například invertování několika bitů v souboru způsobené rušením při přenosu přes linku. Dalším příkladem, kdy mohou vzniknout takovéto chyby, může být dlouhodobé uložení souboru na mechanickém médiu. Cílem této práce není zjištění, jak předcházet a zabraňovat vzniku těchto chyb, ale jak nejlépe využít ochranné prostředky formátu JPEG 2000, aby při dekompresi byl soubor zpětně rekonstruovatelný a co nejvíce podobný originálním souboru.

Pro provádění analýzy byla vytvořena datová sada, která obsahuje 111 souborů pro několik různých kategorií obrazu. Jednotlivé kategorie v datové sadě se liší zejména rozlišením a velikostí souboru. Soubory v tomto datasetu jsou uloženy ve nekomprimovaném formátu PPM. Podobněji je datová sada popsána v kapitole 3.3.

V kapitole 2 je nastíněn princip formátu JPEG 2000. Dále jsou zde popsány způsoby zabezpečení, kterými se zabývá tato studie. Prvním způsobem zabezpečení je použití SOP a EPH značek, které se přidávají ke každému paketu v souboru a obsahují informace, které pomáhají při zpětné rekonstrukci při nalezení chyby v části datového toku. Druhým způsobem zabezpečení analyzovaným v této práci je použití určitých kompresních módů. Mezi prozkoumávané módy patří SEGMARK, BYPASS, RESET, RESTART, CAUSAL a ERTERM. Detailněji budou značky a módy používané při kompresi souboru popsány v kapitole 2.2. Následuje popis knihovny Kakadu a knihovny OpenJPEG, které byly použity pro kompresi a dekompresi obrazů při zpracovávání analýzy.

Následuje vlastní analýza, ve které bude nejprve představen použitý model poškození, metrika PSNR, která byla použita při srovnávání kvality zpracovaných obrazů vůči originálu. V této části je popsána také datová sada vytvořená a použita pro experimenty. Kapitola 3 se zabývá analyzováním vlivu různých druhů zabezpečení a jejich kombinacemi. Zkoumá se zde režie spojená s konkrétním druhem zabezpečení a jeho odolnost proti stupňujícímu se poškození. Výsledky jsou vizualizovány prostřednictvím grafů ukazujících srovnání různých nastavení. Před závěrem práce bude provedeno krátké srovnání zabezpečení formátu JPEG 2000 a formátu CCSDS verze 122.0, který se používá na palubě kosmických lodí pro kompresi nekonečně dlouhých pruhů v obraze. Tyto obrazy vznikají například při skenování povrchu Země družicemi z oběžné dráhy. Závěrem práce jsou popsána zjištění a doporučení pro kompresi dat do formátu JPEG 2000 při použití knihovny Kakadu pro zajištění co možná nejlepší odolnosti proti poškození.

## Kapitola 2

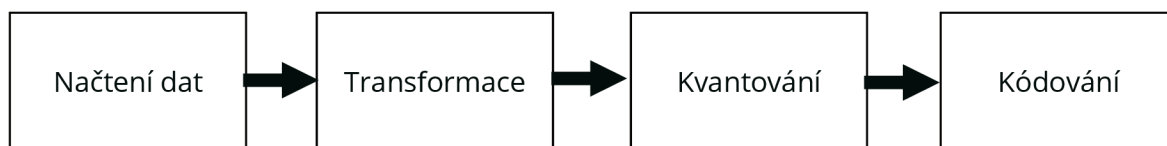
# Kompresie obrazu

V této kapitole bude popsána metoda ztrátové komprese a dále bude představen formát JPEG 2000 a jeho vlastnosti. Podrobněji zde budou popsány také markery a kompresní módy používané při práci. Dále pak budou představeny knihovny Kakadu a OpenJPEG, které byly používány pro experimenty a budou popsány parametry, které byly prozkoumávány. Na konci této kapitoly bude představeno také výchozí doporučení pro kompresi obrazů knihovnou Kakadu podle modelu Národní digitální knihovny České republiky.

### 2.1 Ztrátová komprese

Kompresí dat rozumíme zpracování počítačových dat (bytů a bitů) za účelem zmenšení velikosti souboru, jež tvoří a zároveň zachovat informaci v těchto datech obsaženou. Jedná o snahu zmenšit velikosti souborů pro archivaci či přenos po počítačových sítích. Rozlišujeme dva typy komprese - ztrátovou a bezztrátovou [7]. Bezztrátová komprese není tak účinná jako ztrátová, ovšem její výhoda je v možnosti rekonstruovat komprimovaný obraz do původní podoby.

Ztrátová komprese je způsob zakódování dat, při kterém nenávratně dochází ke ztrátě části dat. Obraz tedy není zpětně rekonstruovatelný narozdíl od bezztrátové komprese. Výhodou je však mnohem větší zmenšení výsledného souboru oproti referenčnímu. Ztrátová komprese má velké využití v multimédiích, kde lidské oko není schopno zaregistrovat určité zkreslení. Při vyšším kompresním poměru může docházet k vytvoření fragmentů uvnitř obrazu. V této studii se pracuje pouze se ztrátovou kompresí.



Obrázek 2.1: Průběh ztrátové komprese

Na obrázku 2.1 je znázorněn průběh procesu obecného zpracování obrazu pomocí ztrátové komprese. Načtená data zde mohou být zpracována například pomocí diskrétní kosinové transformace jako například u formátu JPEG nebo třeba diskrétní vlnkovou transformací, kterou používá formát JPEG 2000. Zpracované koeficienty dále postupují k procesu kvantování, kde dochází ke zkreslení a nenávratné ztrátě dat. Posledním krokem je zakódování kvantovaných dat.



Obrázek 2.2: Srovnání bezztrátové a ztrátové komprese

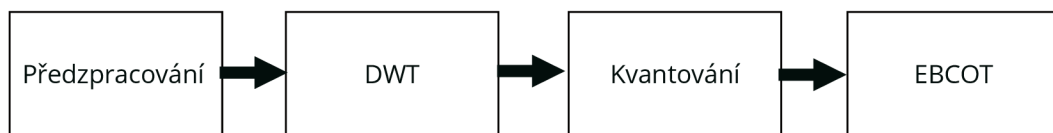
Na obrázku 2.2 je demonstrováno použití ztrátové komprese. V levé části se nachází originální obrázek ve formátu PNG využívající bezztrátovou kompresi. Velikost tohoto obrázku je 12 kB. Prostřední obrázek je komprimovaný do formátu JPEG využívajícího ztrátovou kompresi. Tento obrázek obsahuje o 85% méně původní informace a jeho velikost je 1,8 kB. Obrázek napravo je komprimován s ještě větším kompresním poměrem. Tento obraz obsahuje o 96% méně původní informace a jeho velikost je pouze 0,56 kB, ovšem zkreslení v tomto obraze je již velmi výrazné.

## 2.2 JPEG 2000

Jedná se o obrazový formát umožňující ztrátovou i bezztrátovou kompresi, který byl navržen jako nástupce formátu JPEG z roku 1992. První část standardu je popsána v dokumentu ISO/IEC 15444-1. Jádrem tohoto formátu je diskrétní vlnková transformace (DWT) [2]. Formát JPEG 2000 pracuje s barevným modelem YCbCr, kde jednotlivé složky mohou být přepočítány podle rovnic 2.1. V tomto modelu Y zastupuje jas obrázku a hodnoty Cb a Cr zastupují modrou a červenou složku.

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0,2999000 & 0,587000 & 0,114000 \\ -0,168735 & -0,331264 & 0,500000 \\ 0,500000 & -0,418688 & -0,081312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.1)$$

Pro dekompozici signálů je využita vícerozměrná analýza, která aproximuje originální signál z jemného na hrubší rozlišení. Podpásma jsou rozdělena na oblasti a bloky kódování. Každý blok je po bitových rovinách zkomprimován algoritmem EBCOT. Komprimovaný tok je přeorganizován do vrstev a následně zaobalen do paketů. Tento princip je ukázán na obrázku 2.4 a bude detailně popsán v této kapitole.

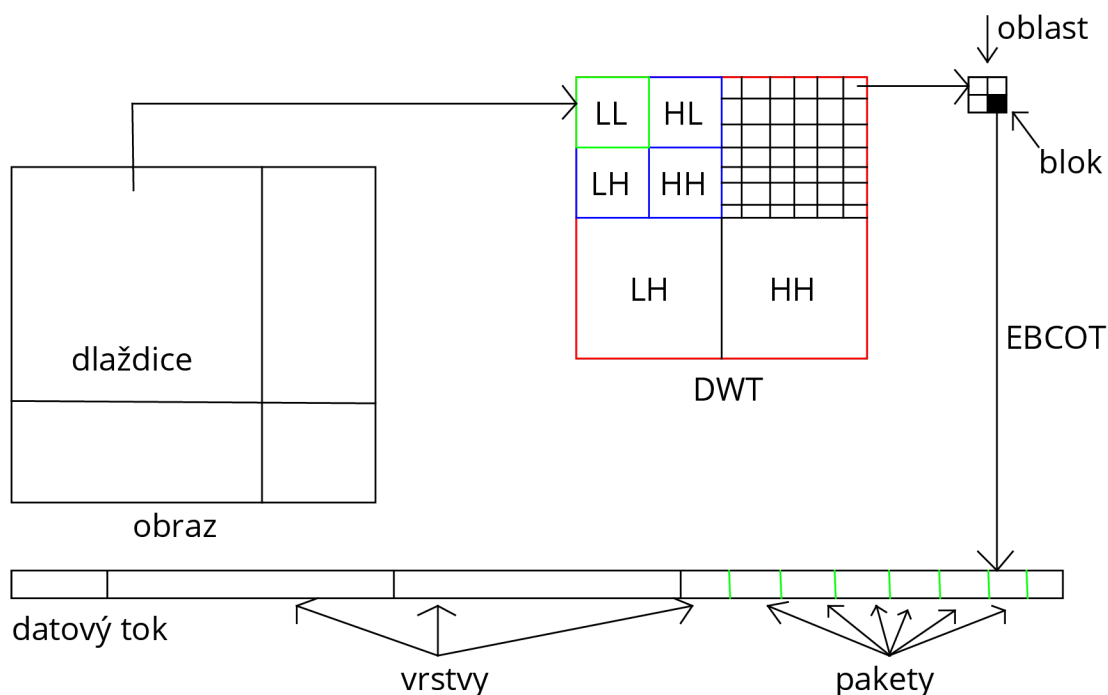


Obrázek 2.3: Schéma komprese metodou JPEG 2000. Převzato z [3]



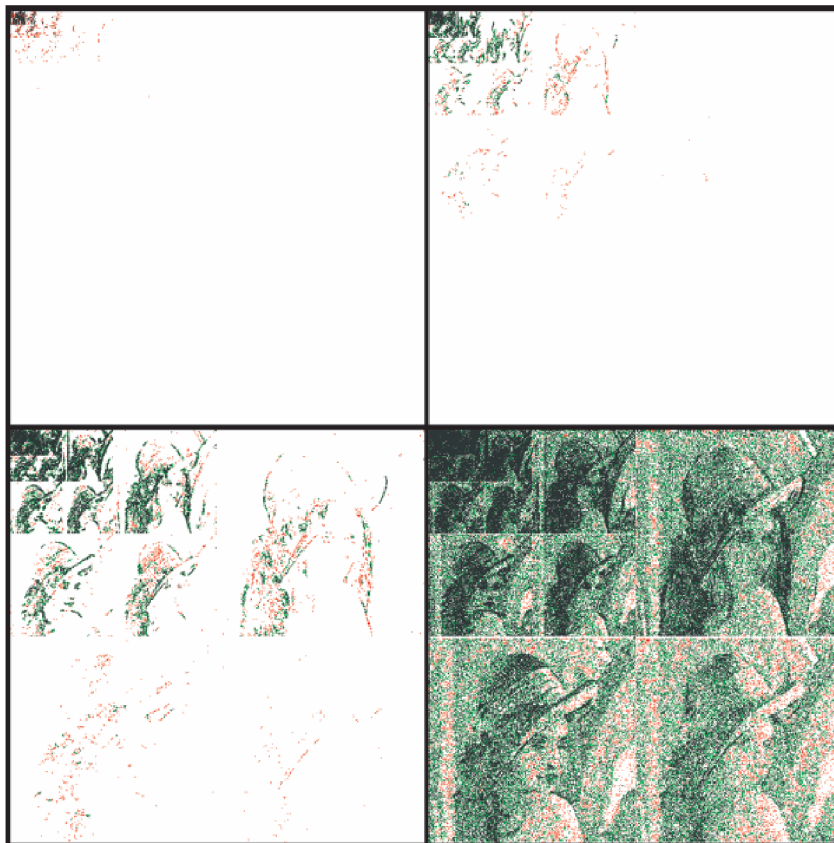
V procesu předzpracování probíhá podobně jako u formátu JPEG rozdělení obrazu do bloků. Zde se tyto bloky nazývají dlaždice. Dlaždice jsou zpracovávány na sobě nezávisle a pro každou je provedena diskretní vlnková transformace DWT, která bude popsána níže.

Každá barevná složka je rozdělena na dlaždice o stejné velikosti. Pouze okrajové dlaždice mohou mít menší velikost, jelikož se ořežou části přesahující hranice obrazu. Na každé dlaždici se spočítá diskretní vlnková transformace DWT. Jednotlivá rozlišení jsou u DWT na obrázku 2.4 znázorněna barevně. Na hranicích jednotlivých dlaždic mohou vznikat artefakty, které způsobují šum v obraze. Použití většího počtu menších dlaždic může zapříčinit vzniku většího šumu v obraze. Naopak použití malého počtu velkých dlaždic zvedá paměťové nároky při zpracování.



Obrázek 2.4: Princip elementů formátu JPEG 2000 [3]

Diskretní vlnková transformace umožňuje přenést obraz ve více rozlišeních najednou. Standard JPEG 2000 využívá dvourozměrnou vlnkovou transformaci. Základem dvourozměrné diskretní vlnkové transformace je jednorozměrná transformace aplikovaná zvláště na sloupce a zvláště na řádky obrazu, kde je na vstupní koeficienty aplikována dvojice filtrů horní a dolní propusti definované rovnicemi 2.2 a 2.3. Následně je signál podvzorkován - jsou z něj odstraněny liché koeficienty. Každá úroveň rozkladu obsahuje čtyři podpásma - HL, LH, HH, LL. Tato čtyři subpásma mají dohromady stejnou plochu jako originální obraz. S každou další úrovní rozkladu je rozlišení vždy poloviční. Princip této dekompozice je naznačen na obrázku 2.4 a 2.5. Subpásma v levé horní části dekompozice, označené jako LL, může být dále dekomponováno, což je demonstrováno na obrázku 2.6. První písmeno označení subpásma značí typ frekvenčního lineárního filtru ve vertikálním zpracování, druhé písmeno značí typ frekvenčního lineárního filtru pro horizontální zpracování signálu. Typem může být horní propust označená písmenem H nebo dolní propust označená písmenem L.



Obrázek 2.5: Dekompozice obrazu na nižší rozlišení [9]

Diskrétní vlnková transformace využívá pyramidovou strukturu filtrů s impulzní odezvou (FIR). Jednotlivé úrovně rozkladu odpovídají jednotlivým rozlišením, která jsou s každou další úrovní poloviční. Pro ztrátovou kompresi se využívá vlka CDF 9/7 a pro beztrátovou CDF 5/3.

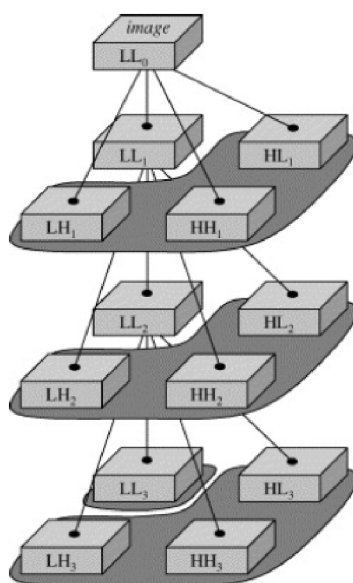
$$y_L[n] = \sum_{k=-\infty}^{\infty} x[k]h_0[2n - k] \quad (2.2)$$

$$y_H[n] = \sum_{k=-\infty}^{\infty} x[k]h_1[2n - k] \quad (2.3)$$

Diskrétní vlnková transformace je paměťově i výpočetně složitým algoritmem. Z hlediska implementace na grafické kartě je tradiční DWT také obtížně paralelizovatelná. Byl navržen algoritmus známý jako *schéma lifting*, který nabízí úsporu paměťové i výpočetní složitosti. Výpočet je realizován tak, že se nejprve sestaví matice filtrů horní propusti  $h_0$  2.2 a dolní propusti  $h_1$  2.3, která je znázorněná rovnicí 2.4.

$$P(z) = \begin{bmatrix} h_{0_{even}} & h_{1_{even}} \\ h_{0_{odd}} & h_{1_{odd}} \end{bmatrix} \quad (2.4)$$

$P$  značí krok predikce, který bude brán v úvahu samostatně při dalším zpracování. Jedná se o vysokoprůchodový filtr, který vypočítá vlnovou funkci k vlnkové transformaci. Krok aktualizace vypočítává změny měřítka, což má za následek hladší verzi dat.



Obrázek 2.6: Rozklad dlaždice do 4 podpásem LL, LH, HL, HH [6]

Polyfázová matice 2.4 je matice  $2 \times 2$  obsahující analýzy nízkoprůchodových a vysokoprůchodových filtrů, z nichž každá je rozdělena na jejich sudé a liché polynomiální koeficienty a je normalizována. Odtud je matice započítána do řady  $2 \times 2$  horních a dolních trojúhelníkových matic, z nichž každá má diagonální hodnoty rovné jedné. Horní trojúhelníkové matice obsahují koeficienty pro predikční kroky a dolní trojúhelníkové matice obsahují koeficienty pro kroky aktualizace. Polyfázová matice je zpracována do formy, kde  $a$  je koeficient pro předpovědní krok a  $b$  je koeficient pro krok aktualizace. Toto zobrazuje rovnice 2.5.

$$P(z) = \begin{bmatrix} 1 & a(1+z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b(1+z) & 1 \end{bmatrix} \quad (2.5)$$

Schéma lifting může používat i složitější extrakci dat využívající najednou více prediktivních a aktualizacních kroků. Stejně tak může být používáno více kroků škálování. Toto je ukázáno v rovnici 2.6, kde  $a$  je koeficient pro první prediktivní krok a  $b$  je koeficientem prvního aktualizacního kroku. Písmena  $c$  a  $d$  poté postupně označují koeficienty pro druhý prediktivní a druhý aktualizacní krok. V poslední matici této rovnice označuje  $k_1$  škálovací koeficient pro sudé vzorky a  $k_2$  je koeficientem pro škálování lichých vzorků.

$$P(z) = \begin{bmatrix} 1 & a(1+z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b(1+z) & 1 \end{bmatrix} \begin{bmatrix} 1 & c(1+z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ d(1+z) & 1 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \quad (2.6)$$

Takto získané koeficienty dále postupují do procesu kvantování. Kvantování je prováděno pro všechna podpásma a jeho úkolem je zmírnit přesnost dat, zanedbat určitou část a tím docílit komprese. Účinnost procesu závisí na velikosti kvantizačního kroku. Při použití bezztrátové komprese je kvantizační krok roven jedné, což znamená, že se kvantování neprovádí. Při použití ztrátové komprese je kvantizační krok počítán pro každé zpracovávané podpásma.

Unikátní vlastností formátu JPEG 2000 je funkce ROI (Region of interest), neboli oblast zájmu. Při zpracování je možné definovat část, která může mít libovolný tvar, a která

bude zpracovávána s větší přesností. Tato část bude rozložena na menší části. V obraze lze tímto způsobem odlišit důležité a nedůležité části. Využití tohoto mechanismu může být například při rozpoznávání obličejů v obraze. Příklad využití funkce ROI je znázorněna na obrázku 2.7. Při dekompresi je tedy oblast zájmu známa dříve než pozadí. To může být užitečné například při přenosu velkých obrazů po síti.



Obrázek 2.7: Příklad využití funkce ROI pro zvýraznění obličeje na fotografii [6]

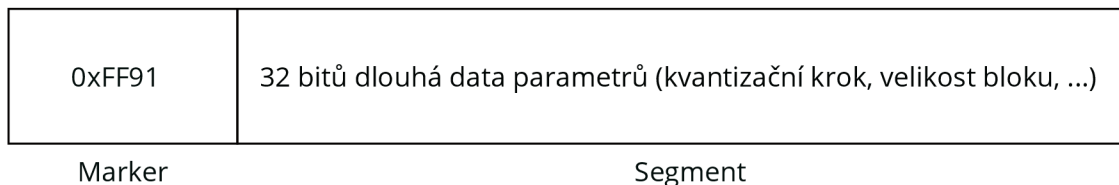
Posledním prvkem zpracování obrazu je kódování algoritmem EBCOT (Embedded Block Coding with Optimized Truncation), což je znázorněno na obrázku 2.3. Pro účely kódování koeficientů jsou sub pásma rozdělena na obdélníkové oblasti, které odpovídají prostorovým částem obrazu a jsou tak využitelné při sekvenčním pořadí progresivního přenosu. Velikostí oblasti je v důsledku omezena velikost paketů a informace o těchto oblastech jsou uloženy v segmentech COD a COC [3]. Vstupem pro zpracování algoritmem EBCOT jsou bloky o velikost  $32 \times 32$  nebo  $64 \times 64$  pixelů, které obsahují kvantizovaný výstup z diskretní vlnkové transformace v podobě vlnkových koeficientů. Bloky jsou zpracovávány na sobě nezávisle. Vstupní koeficienty se kódují od nejvýše významného bitu MSB po nejméně významný bit LSB a zpracují se tedy všechny koeficienty popředí ještě před koeficienty pozadí.

Datový tok je rozdělen na vrstvy kvality, jejichž účelem je postupné zvyšování kvality s každou další dekódovanou vrstvou. Všechna komprimovaná data obrazu jsou nakonec zformována do paketů. Paket je spojitý segment, který obsahuje data z jedné konkrétní dlaždice, vrstvy kvality, barevné složky, rozlišení a oblasti. Uvozuje se hlavičkou paketu a dále obsahuje informace o kvalitě, barevných složkách, rozlišení a pozici. Paket může být dále opatřen různými markery, o kterých bude řeč v následující části.

Algoritmus EBCOT je flexibilní při vytváření datového toku. Lze využívat škálovatelnosti a při použití vrstev kvality lze využívat zkrácení nebo postupné zvyšování kvality obrazu a zaměřovat se na určité části obrazu pomocí ROI [1].

## První způsob zabezpečení - použití SOP a EPH značek

Standard JPEG 2000 obsahuje širokou škálu markerů, které obsahují informace o struktuře datového toku. Markery mohou být umístěny v hlavičce datového toku i hlavičkách paketů. Za markerem může následovat segment s dalšími informacemi. Tento segment je variabilní délky a může být dlouhý nula až čtyři byty. Struktura markeru a jeho segmentu je zobrazena na obrázku 2.8.



Obrázek 2.8: Struktura markeru a segmentu[11]

Segment může nést další informace, jako například velikost kroku při kvantování, velikost oblasti rozdělení subpásem, velikost code bloku v oblasti a další. Informace obsažená v segmentu markeru závisí na typu markeru.

Jsou-li během komprese použity SOP a EPH značky, přidají se určité informace před a za každý paket v souboru. SOP (Start Of Packet) se přidává před každý paket a EPH (End Of Packet) se přidává za každý paket. Režie použití značky SOP je 6 bytů pro každý výskyt. Dva byty označují začátek značky a další 4 byty nesou další informace viz. obrázek 2.8. Režie značky EPH je pouze 2 byty. Jejich účelem by mělo být pomoci při rekonstrukci poškozených paketů v souboru. Zvyšují odolnost souboru proti přenosovým chybám a jejich užití se doporučuje v rušném prostředí, kdy může dojít k narušení přenosu celého komprimovaného toku a kdy neexistují jiné mechanismy, které by si poradily s přenosovými chybami. Přijímající systém (protokol) tak může snadno detekovat, že přenesená data nejsou kompletní a podle umístění značek SOP a EPH dokáže lokalizovat poškozený paket a poškozenou oblast v obrazu. Použití těchto značek tak zvyšuje odolnost vůči přenosovým chybám. Oba tyto markery jsou součástí datového toku. Hodnota v poli marker dle obrázku 2.8 pro značku SOP je 0xFF91 a pro EPH 0xFF92.

Datový tok obrazu ve formátu JPEG 2000 může obsahovat další markery, které je možno vidět v níže přiložené tabulce.

## Druhý způsob zabezpečení - využití módů komprese

Jiným způsobem zabezpečení než vkládání značek do datového toku, je využití kompresních módů a jejich kombinací. Kompresní módy analyzované v této práci jsou BYPASS, SEGMARK, RESET, RESTART, CAUSAL a ERTERM, které budou lépe popsány níže.

- BYPASS - Tento režim umožňuje redukovat složitost kódování při vysokém bitrate s malou nebo žádnou ztrátou účinnosti komprese. Je-li tento mód zapnut, kodér při kompresi vynechá zpracování některých méně významných dat, čímž se urychlí proces komprese i následné dekomprese. Při použití tohoto kompresního módu je použito takzvané líné vyhodnocování, které snižuje složitost kódování bitové roviny za cenu účinnosti kódování. .
- SEGMARK - V případě použití tohoto módu během komprese se přidává posloupnost čtyř bitů '1010' za každou bitovou rovinu. Přidané bity pomáhají při zjišťování poškození míst. Přidávání zabezpečení probíhá při kódování algoritmem EBCOT.
- RESET - Při použití tohoto módu se stavy kontextu kódování znovu inicializují při kódování každé části. Kontextovými stavy se rozumí pravděpodobnosti použité v aritmetice kódování. V opačném případě jsou stavy kontextu inicializovány pouze jednou před prvním kódováním. Vynucený reset snižuje účinnost kódování, ale pomáhá oddělit kódované části. Společně s použitím módů RESTART a CAUSAL umožňují paralelní zpracování kódování.
- CAUSAL - Přepínač kauzálního režimu zavádí jemné úpravy v procesu použitém k vytvoření kontextových značek pro kódování bitové roviny. Cílem je zajistit, aby vzorky v daném proužku mohly být kódovány bez závislosti na vzorcích z budoucích pruhů. Ve výchozím stavu jsou kontexty kódování tvořeny pomocí  $3 \times 3$  velkého kontextového okna. Dopad přepínače causal je, že budoucí vzorky jsou pro účely vytvoření kontextu považovány jako nevýznamné.
- RESTART - Tento mód způsobí restartování aritmetického kodéru na začátku každého kódovacího průchodu. Když je tedy tento režim využit, každý kódovací průchod má svůj vlastní segment kódového slova MQ. Je třeba poznamenat, že délka každého z těchto segmentů je signalizována v hlavičce paketu. Pokud není tento mód použit, aritmetický kodér se restartuje pouze na začátku každého zpracování každého bloku.
- ERTERM - Použití tohoto přepínače říká dekodéru, že při nalezení poškozeného segmentu během zpracování jej nemá hned zahazovat celý. Pokud kodér používá předvídatelnou politiku ukončení a tento mód je zapnutý, napomáhá tento mód při detekci použitelných částí poškozených segmentů v datovém toku i hlavičkách paketů. .

## Zajímavé kombinace kompresních módů

- Paralelní zpracování - Kódové bloky jsou při této variantě kódovány nezávisle, což zajišťuje lepší propustnost při zpracování. Tento mód není k dispozici při dekódování souboru. Paralelního zpracování je možno dosáhnout současným použitím přepínačů RESET, RESTART a CAUSAL. Výhodou je, že se nemusí v paměti udržovat kompletní reprezentace stavových polí.
- Líné paralelní zpracování - Při použití paralelního zpracování se každý vzorek zpracovává jedním z tří kódovacích průchodů. V určitou chvíli mohou být zbylé procesory nečinné. Pro vyhnutí se těmto důsledkům může být současně s paralelním zpracováním použit mód BYPASS a vzniká takzvané líné paralelní zpracování, jehož použití snižuje počet použitých paralelních kódérů.
- Kombinace ERTERM a SEGMARK - Tato kombinace vytvoří samostatný segment předvídatelně ukončeného kódového slova pro každý kódovací průchod. Jakákoliv chyba v bitovém toku tak pravděpodobně opustí kódér ve stavu, který je v rozporu s předvídatelnou ukončovací politikou. Dekodér při nalezení takové části v datovém toku tato data zahodí. Souběžné použití kompresních módů ERTERM A SEGMARK nabízí dobrou odolnost proti chybám, ale za vyšší režii.

## Srovnání JPEG a JPEG 2000

Na rozdíl od formátu JPEG obsahuje formát JPEG 2000 možnost využití bezztrátové komprese. Nevýhodou JPEG 2000 je větší režie při zpracovávání. Podle případových studií zvládá JPEG 2000 zkomprimovat obraz od 20% do 200% lépe než JPEG[8].

Formát JPEG využívá diskrétní cosinovou transformaci, zatímco JPEG 2000 využívá diskrétní vlnkovou transformaci, která je na výpočet náročnější, a proto JPEG 2000 vyžaduje komplexnější kompresní software. Čtení i zápis proto trvá o něco déle.

Navzdory tomu, že je JPEG 2000 technicky vspělejší, je stále více používán starší JPEG, jelikož JPEG 2000 není se svým starším modelem zpětně kompatibilní a většina fotografií v dnešní době je právě ve formátu JPEG.

## 2.3 Knihovna Kakadu

Knihovna kakadu je produkt Kakadu Software<sup>1</sup>, který obsahuje implementaci první části standardu JPEG 2000 podle ISO/IEC 15444-1 a značnou část druhé a třetí části standardu. Tato knihovna je napsána v jazyce C++ a autem je David Taubman. K tomuto softwaru existuje velmi rozsáhlá a dobře zpracovaná dokumentace, diskuzní fórum a také názorné příklady použití.

Při experimentování s knihovnou Kakadu byly využívány následující nastavení. Jejich parametry se měnily v závislosti na zkoumané vlastnosti. V nastavení bylo vycházeno z profilu Národní knihovny a jejího doporučeného nastavení pro kategorii *Knihy a periodika* [12], které je detailně popsáno v kapitole 2.4.

### Parametry používané při kompresi obrazu knihovnou Kakadu

| Parametr    | Hodnota     | Výnam  |
|-------------|-------------|--|
| -i          | INAME       | Parametr INAME definuje jméno vstupního souboru              |
| -o          | ONAME       | Parametr ONAME definuje jméno výstupního souboru             |
| Creversible | no          | Zapnutí ztrátové komprese                                    |
| Cblk        | {64,64}     | Velikost jednoho bloku                                       |
| Corder      | RPCL        | Průběh zobrazení, určuje pořadí paketů v code streamu        |
| Cstiles     | {1024,1024} | Velikost jedné dlaždice                                      |
| Clevels     | 5           | Počet úrovní rozlišení                                       |
| Cmode       | X           | BYPASS SEGMARK RESET RESTART                                 |
| -rate       | X           | X označuje Počet bitů na pixel požadovaný při kompresi       |
| Clayers     | 12          | Počet vrstev kvality v obraze                                |
| Cprecincts  | {256, 256}  | Velikost oblastí podpásem ({128, 128}) pro poslední rozklady |
| Cuse_sop    | no   yes    | Použití značek pro označení začátků paketů                   |
| Cuse_eph    | no   yes    | Použití značek pro označení konce paketů                     |

Tabulka 2.1: Parametry používané při kompresi - Kakadu

Při kompresi obrazu byly vždy explicitně zadávány všechny parametry zobrazené v tabulce 2.1. Měnily se hodnoty parametrů -rate, Cmode, Cuse\_sop a Cuse\_eph v závislosti na aktuálním typu komprese.

### Parametry použité při dekompresi obrazu knihovnou Kakadu

| Parametr   | Hodnota | Výnam  |
|------------|---------|--|
| -i         | INAME   | Parametr INAME definuje jméno vstupního souboru  |
| -o         | ONAME   | Parametr ONAME definuje jméno výstupního souboru |
| -resilient |         | Snaha opravit chyby v poškozených datech         |

Tabulka 2.2: Parametry používané při dekompresi - Kakadu

Při dekompresi obrazů byl kromě definování vstupního a výstupního souboru použit ještě přepínač -resilient, který se snaží napomáhat dekompresi při odhalení poškozeného místa.

<sup>1</sup>Kakadu Software - <http://kakadusoftware.com>



## 2.4 Knihovna OpenJPEG

Knihovna OpenJPEG<sup>2</sup> je otevřená knihovna implementující kodek JPEG 2000 a je napsána v jazyce C. Narozdíl od Kakadu je výpočetně pomalejší a nenabízí tolik nástrojů jako Kakadu. Má velice stručnou dokumentaci a v mnoha případech je její použití nedostačující.

Při experimentování s knihovnou OpenJPEG byly využívány následující nastavení. Jejich parametry se měnily v závislosti na zkoumané vlastnosti. V nastavení bylo vycházeno z profilu Národní knihovny a jejího doporučeného nastavení pro kategorii *Knihy a periodika* [12], které je detailně popsáno v kapitole 2.4.

### Parametry používané při kompresi obrazu knihovnou OpenJPEG

| Parametr | Hodnota   | Výnam  |
|----------|-----------|--|
| -i       | INAME     | Parametr INAME definuje jméno vstupního souboru            |
| -o       | ONAME     | Parametr ONAME definuje jméno výstupního souboru           |
| -I       | no        | Zapnutí ztrátové komprese                                  |
| -b       | 64,64     | Velikost jednoho bloku                                     |
| -p       | RPCL      | Průběh zobrazení, určuje pořadí paketů v code streamu      |
| -t       | 1024,1024 | Velikost jedné dlaždice                                    |
| -n       | 5         | Počet úrovní rozlišení                                     |
| -M       | X         | Součet označující použitou kombinaci módů viz. 2.2         |
| -r       | X         | Kompresní poměr udávaný pro každou vrstvu kvality          |
| -c       | 256, 256  | Velikost oblastí podpásem (128, 128) pro poslední rozklady |
| -SOP     |           | Použití značek pro označení začátků paketů                 |
| -EPH     |           | Použití značek pro označení konce paketů                   |

Tabulka 2.3: Parametry používané při kompresi - OpenJPEG

Při kompresi obrazu byly vždy explicitně zadávány všechny parametry zobrazené v tabulce 2.3. Měnily se hodnoty parametrů -r, -M, použití -SOP a -EPH v závislosti na aktuálním typu komprese.

### Parametry použité při dekompresi obrazu knihovnou OpenJPEG

| Parametr | Hodnota | Výnam  |
|----------|---------|--|
| -i       | INAME   | Parametr INAME definuje jméno vstupního souboru  |
| -o       | ONAME   | Parametr ONAME definuje jméno výstupního souboru |

Tabulka 2.4: Parametry dekomprese - OpenJPEG

Narozdíl o knihovny Kakadu nemá dekompresní program knihovny OpenJPEG mód -resilient, který pomáhá dekompresi při nalezení poškozených míst.

<sup>2</sup>OpenJPEG - <https://www.openjpeg.org/>

## Výchozí nastavení komprese

Při kompresi obrazu se v této práci vycházelo z doporučeného nastavení dle profilu Národní digitální knihovny [12]. Specifikace vychází z potřeb Národní knihovny České republiky a z porovnání doporučení pro kódování souborů ve formátu JPEG 2000 v rámci zahraničních projektů zabývajících se digitalizací a dlouhodobým uchováním obrazových dat. Národní digitální knihovna zveřejňuje tři kategorie nastavení pro obrazová data. V této práci se vychází z doporučeného nastavení pro zpřístupňující kopie pro knihy a periodika využívající ztrátovou kompresi obrazu.

| Popis parametru                  | Hodnota                       |
|----------------------------------|-------------------------------|
| Využití                          | Knihy, periodika              |
| Grafický formát                  | JPEG 2000                     |
| Přípona formátu                  | *.jp2                         |
| Využitá část specifikace         | 1. část JPEG 2000             |
| Migrační software                | Kakadu                        |
| Druh komprese                    | ztrátová                      |
| Transformace                     | 9/7 nevratný filtr            |
| Kompresní poměr                  | 1:20 až 1:30                  |
| Dlaždice                         | 1024 × 1024                   |
| Průběh zobrazení                 | RPCL                          |
| Počet dekompozičních úrovní      | 5                             |
| Počet vrstev kvality             | 12                            |
| Velikost regionů (precincts)     | {256,256},{256,256},{128,128} |
| Použití zájmových oblastí        | Ne                            |
| Velikost bloků                   | 64 × 64                       |
| Značka lokalizující dlaždice TLM | Ano ("R")                     |
| Přemostění BYPASS                | Ano                           |
| ICC profily                      | Ne                            |
| Použití značek SOP               | Volitelné                     |
| Použití značek EPH               | Volitelné                     |
| Vložená metadata                 | Ne                            |

Tabulka 2.5: Výchozí nastavení komprese dle NDK [12]

V tabulce 2.5 jsou shrnuta doporučení pro kompresi obrazů kategorie zpřístupňujících kopií pro využití na knihy a periodika. Hodnota v poli parametru velikosti regionů (precincts) znamená, že se použije velikosti regionu  $256 \times 256$  pro první dvě dekompoziční úrovně a pro nižší úrovně se použije velikost regionů  $128 \times 128$ .

## Kapitola 3

# Analýza zabezpečení JPEG 2000 pro ztrátovou kompresi

Na začátku této kapitoly bude popsán model poškození používaný při experimentech a vzá-  
pětí popis metriky PSNR používané k měření kvality dekomprimovaných obrazů. Následovat  
bude popis datové sady na níž byly prováděny experimenty a dále pak budou v této kapitole  
porovnávány výsledky analýzy pro zpracovávání obrazů s různými kompresními parametry.  
Knihovna OpenJPEG se při experimentech ukázala být nevhodná, což je popsáno podrob-  
něji v kapitole 3.4. Analýza je cílena na dva druhy zabezpečení. Prvním z nich je použití  
přidávání značek *SOP* (start of packet) a *EPH* (end of packet) do datového toku a analýza  
vlivu na změnu velikosti a zabezpečení souboru. Druhou částí bude analýza odolnosti proti  
poškození pomocí použití různých kombinací kompresních módů *BYPASS*, *SEGMARK*,  
*RESET*, *RESTART*, *CAUSAL* a *ERTERM*, které jsou detailněji popsány v kapitole 2.2.

### 3.1 Model poškození

Poškození bylo realizováno tak, že v každém vytvořeném souboru ve formátu JPEG 2000  
byla invertována určitá procentuální část bitů. Toto poškození bylo prováděno až za hla-  
vičkou JPEG 2000 souboru. Model byl použit tak, že invertované bity v souboru byly rov-  
noměrně rozloženy v celém datovém toku. Experimentováním bylo zjištěno, že poškození  
více bitů v jednom bytu vede ke stejným výsledkům jako invertování jednoho bitu v bytu.  
Vždy je poškozen celý byte. Z tohoto důvodu byl model poškození navržen na invertování  
samotných bitů náhodně v datovém toku.

V první iteraci poškození bylo invertováno 0,00005 % bitů v souboru. S každou další  
iterací poškození bylo poškození dvojnásobné. Poškození 0,02 % bitů v souboru se experi-  
mentováním ukázalo jako hraniční. Míra poškození vyjádřena procentuálně může zavádět  
tím, že se jedná o příliš malé poškození. Počet bitů pro pár konkrétních souborů je demon-  
strován v tabulce 3.1.

| Velikost | Bitrate | Procentuální poškození | Počet poškozených míst |
|----------|---------|------------------------|------------------------|
| 2,7 MB   | 0,2     | 0,000005 %             | 2                      |
| 2,7 MB   | 0,2     | 0,000640 %             | 141                    |
| 2,7 MB   | 0,2     | 0,002560 %             | 562                    |
| 2,7 MB   | 0,2     | 0,005120 %             | 1 124                  |
| 2,7 MB   | 0,2     | 0,010240 %             | 2 247                  |

|         |     |            |         |
|---------|-----|------------|---------|
| 2,7 MB  | 0,2 | 0,020480 % | 4 493   |
| 13,6 MB | 1,0 | 0,000005 % | 6       |
| 13,6 MB | 1,0 | 0,000640 % | 699     |
| 13,6 MB | 1,0 | 0,002560 % | 2 792   |
| 13,6 MB | 1,0 | 0,005120 % | 5 583   |
| 13,6 MB | 1,0 | 0,010240 % | 11 164  |
| 13,6 MB | 1,0 | 0,020480 % | 22 3328 |
| 24,5 MB | 1,8 | 0,000005 % | 11      |
| 24,5 MB | 1,8 | 0,000640 % | 1 256   |
| 24,5 MB | 1,8 | 0,002560 % | 5 019   |
| 24,5 MB | 1,8 | 0,005120 % | 10 037  |
| 24,5 MB | 1,8 | 0,010240 % | 20 073  |
| 24,5 MB | 1,8 | 0,020480 % | 40 145  |

Tabulka 3.1: Počet invertovaných bitů v souboru

V prvním sloupci tabulky 3.1 je zobrazena velikost souboru komprimovaného do formátu JPEG 2000 a ve druhém sloupci je zobrazen bitrate tohoto souboru. Třetí sloupec obsahuje použité poškození vyjádřené procentuálně a ve čtvrtém sloupci této tabulky je přepočten počet poškozených míst v konkrétním souboru s určitým poškozením. Hodnoty v této tabulce jsou primárně řazeny podle hodnoty bitrate a sekundárně podle procentuální velikosti poškození.

## 3.2 Metrika PSNR

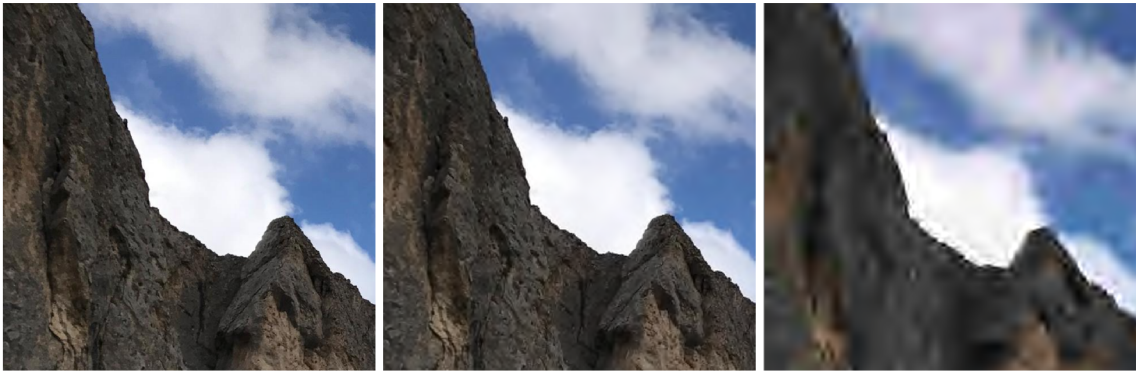
Pro kvalitativní srovnávání zpracovaných obrazů vůči originálu byla využita metrika PSNR (Peak signal-to-noise ratio), která udává špičkový poměr signálu k šumu. Vyhodnocuje podobnost referenčního obrazu vůči zkraslenému. Jedná se o nejpoužívanější a nejjednodušší objektivní metodu hodnocení kvality obrazu. Metoda PSNR využívá střední kvadratickou chybu MSE, která je definována rovnicí 3.1.

$$\text{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left[ I(i, j) - K(i, j) \right]^2 \quad (3.1)$$

Koeficienty  $m$  a  $n$  udávají rozměry obrázku a výraz  $I(i, j) - K(i, j)$  udává odchylku mezi originálním a zkrasleným signálem.

$$\text{PSNR} = 20 \cdot \log_{10} \left( \frac{\text{MAX}^2}{\sqrt{\text{MSE}}} \right) \quad (3.2)$$

Proměnná MSE je výsledkem výrazu z rovnice 3.2 a proměnná MAX je rovna 255 pro bitovou hloubku obrazu rovnou 8 bitům ( $2n - 1$ ) pro každý kanál. Pro RGB obrázky je MSE suma přes všechny složky (R, G, B) dělena třemi. Výsledky metody PSNR se udávají v decibelech [dB]. Typické hodnoty pro komprimované obrázky se pohybují mezi 30-40 dB. Čím vyšší číslo metodou PSNR dostaneme, tím je obraz podobnější originálu. Korelace s výsledky subjektivních testů jsou okolo 80%, což je porovnatelné s komplexnějšími metodami měření kvality obrazu, a proto je metoda PSNR brána jako důvěryhodná. [5]



Obrázek 3.1: Srovnání kvality obrazu

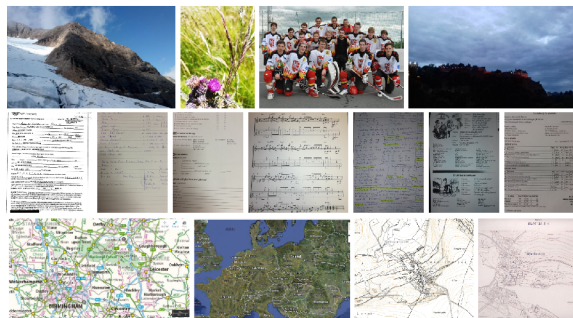
Na obrázku 3.1 je demonstrováno srovnání kvality obrazů za použití ztrátové komprese. V levé části lze vidět originální obraz bez zkreslení ve formátu PPM využívající bezztrátovou kompresi. Prostřední obrázek je výsledkem zpracování, které zahrnovalo ztrátovou kompresi a dekompresi. Tímto způsobem bylo dosaženo určitého zkreslení a kvalita tohoto souboru oproti originálnímu je 36 decibelů. V pravé části lze vidět obraz s výraznějším zkreslením. Jeho kvalita vůči originálu byla změřena na 29 decibelů. Při analýze byla stanovena hranice 35 decibelů, pod níž nemohla kvalita dekomprimovaného obrazu klesnout, aby mohl být prohlášen za úspěšně dekomprimovaný.

### 3.3 Datová sada

V použité datové sadě se nachází 111 fotografií v nekomprimovaném formátu PPM, které jsou rozděleny do tří kategorií a se liší rozlišením obrazů a velikostmi souborů. Ve všech datasetech jsou použity fotografie ve formátu RGB24 (8 bitů na pixel). Dataset obsahuje portréty, fotografie krajin, mapové podklady i skenované dokumenty. Celková velikost vytvořeného datasetu je 2,56 GB.

| Dataset             | Počet souborů | Rozlišení    | Velikost souborů |
|---------------------|---------------|--------------|------------------|
| běžné fotografie    | 100           | 5 až 15 Mpx  | 2 až 35 MB       |
| skenované dokumenty | 10            | 16 až 80 Mpx | 50 až 230 MB     |
| mapové podklady     | 1             | 110 Mpx      | 320 MB           |

Tabulka 3.2: Použité datasety



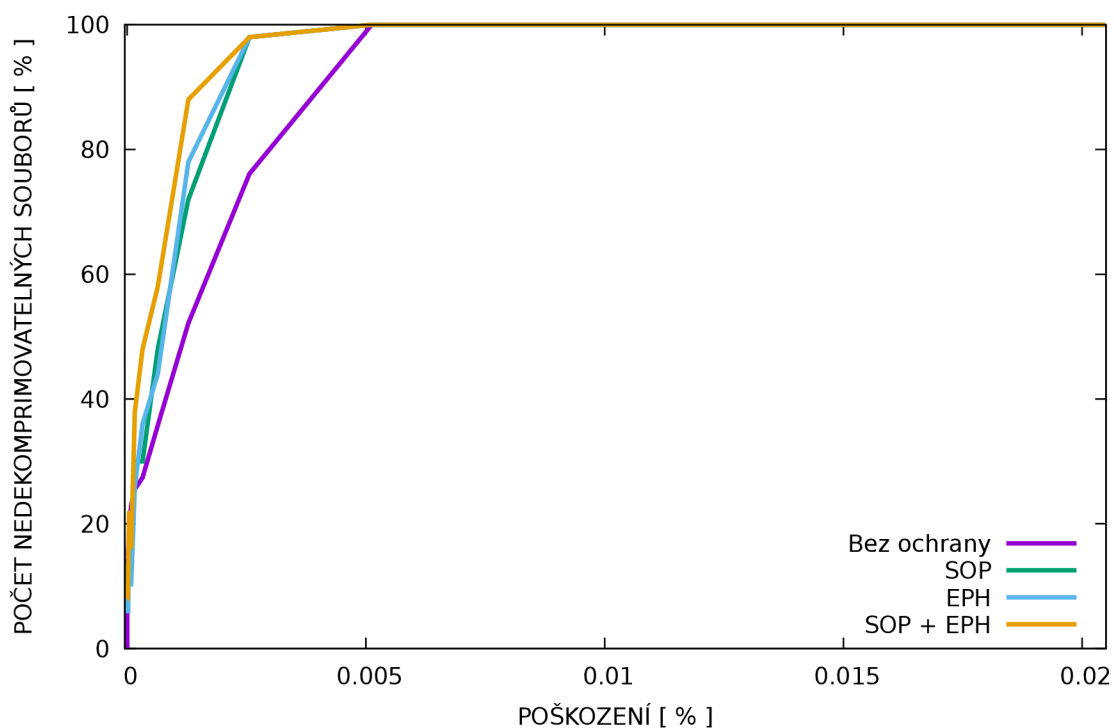
Obrázek 3.2: Matice náhledů z obrázků datové sady

### 3.4 Nevhodnost použití knihovny OpenJPEG

Použití zabezpečovacích mechanismů ve formátu JPEG 2000 může mít za následek zvýšení velikosti komprimovaného souboru. Při nízkých bitratech může být toto zvětšení velmi výrazné, zejména při vkládání značek SOP a EPH do datového toku, jak je možno vidět dále v kapitole 3.6.

Knihovna OpenJPEG nenabízí narozdíl od knihovny Kakadu pružnou dekompresi obrazu jak je popsáno v kapitole 2.2 a 2.4. Zapnutí tohoto módu při dekompresi poškozeného obrazu v případě použití knihovny Kakadu způsobí, že se expanzní program bude snažit opravit nebo potlačit poškozené místo.

Jelikož tyto zabezpečovací mechanismy zvýší velikost souboru a knihovna OpenJPEG neumí pracovat při dekompresi v pružném módu podobně jako knihovna Kakadu, při poškození dosahuje knihovna OpenJPEG horších výsledků pro všechny typy zabezpečení. Toto je možno vidět v grafu na obrázku 3.3, kde křivky pro počet neopravitelných souborů s použitými zabezpečovacími mechanismy nabývají větších hodnot než křivka neopravitelných souborů pro soubory bez přidaných zabezpečovacích mechanismů.



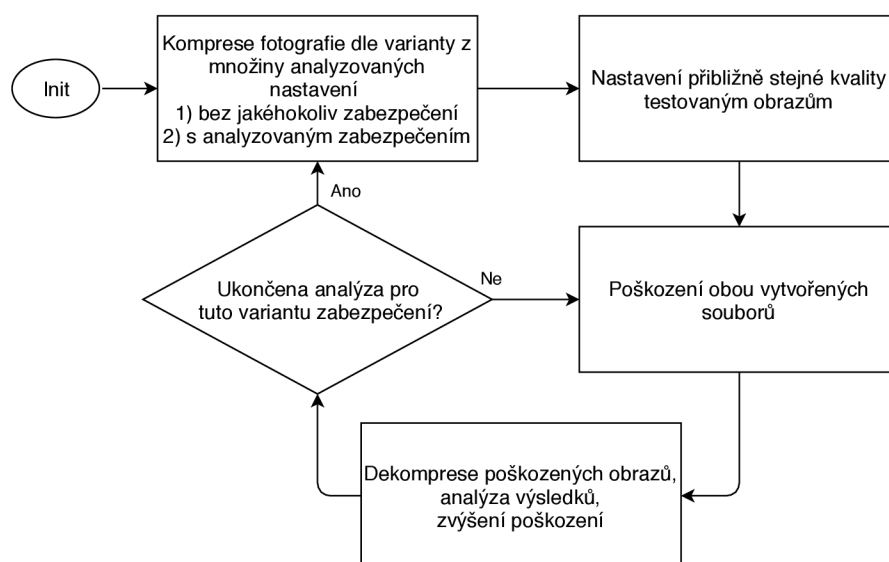
Obrázek 3.3: Nedekomprimovatelné soubory - OpenJPEG

Experimentálně bylo zjištěno, že má knihovna OpenJPEG při kompresi obrazu mnohonásobně pomalejší zpracování. Při měření rychlosti zpracování souboru o velikosti 320 000 bytů a 110 megapixelů trvalo knihovně Kakadu zpracování obrazu přibližně 3 vteřiny. Při použití knihovny OpenJPEG se stejnými kompresními parametry trvalo zpracování téměř dvě minuty. Pro tyto důvody je zbytek práce věnován analýze odolnosti proti poškození formátu JPEG 2000 za použití výhradně knihovny Kakadu.

### 3.5 Postup provádění experimentů

Obrazy byly zpracovávány podle níže popsaného algoritmu. Tento postup je vizualizován diagramem na obrázku 3.4.

1. Komprese referenčního obrazu do dvou variant. První s použitím zabezpečovacích mechanismů a druhá bez nich. Obou variantám byl zadán stejný požadovaný bitrate.
2. Nastavení přibližně stejné kvality komprimovaných souborů vůči referenčnímu za cenu zvyšování bitrate a tudíž i velikosti souboru.
3. V tomto momentě bylo možno změřit zvětšení souboru se zabezpečením oproti souboru bez použití zabezpečení při přibližně stejné kvalitě obrazu. Z toho vyplývá režie použitého zabezpečení.
4. Dále probíhalo poškozování obrazu. Poškození bylo realizováno invertováním  $x$  % bitů v tomto souboru. Použitý model poškození invertoval procentuálně danou část náhodně vybraných bitů v datovém toku. Blíže je tento model poškození popsán v kapitole 3.1.
5. Další částí byla dekomprese poškozených obrazů zpět do formátu PPM.
6. Při úspěšné dekompresi probíhalo měření kvality souboru vůči referenčnímu. Pokud kvalita poškozeného dekomprimovaného obrazu klesla pod definovaný práh, soubor byl prohlášen za neúspěšně zpracovaný. Experimentálním zjišťováním byl tento práh stanoven na hodnotu 35 dB.
7. Poté bylo zvýšeno poškození a analýza probíhala znovu od bodu bodu 4. Poškozování se zvyšovalo až do dosažení hraničního poškození obrazu viz. kapitola 3.1.
8. S každou další iterací se stupňoval explicitně zadávaný bitrate a měnily se parametry komprese.



Obrázek 3.4: Vývojový diagram průběhu analýzy

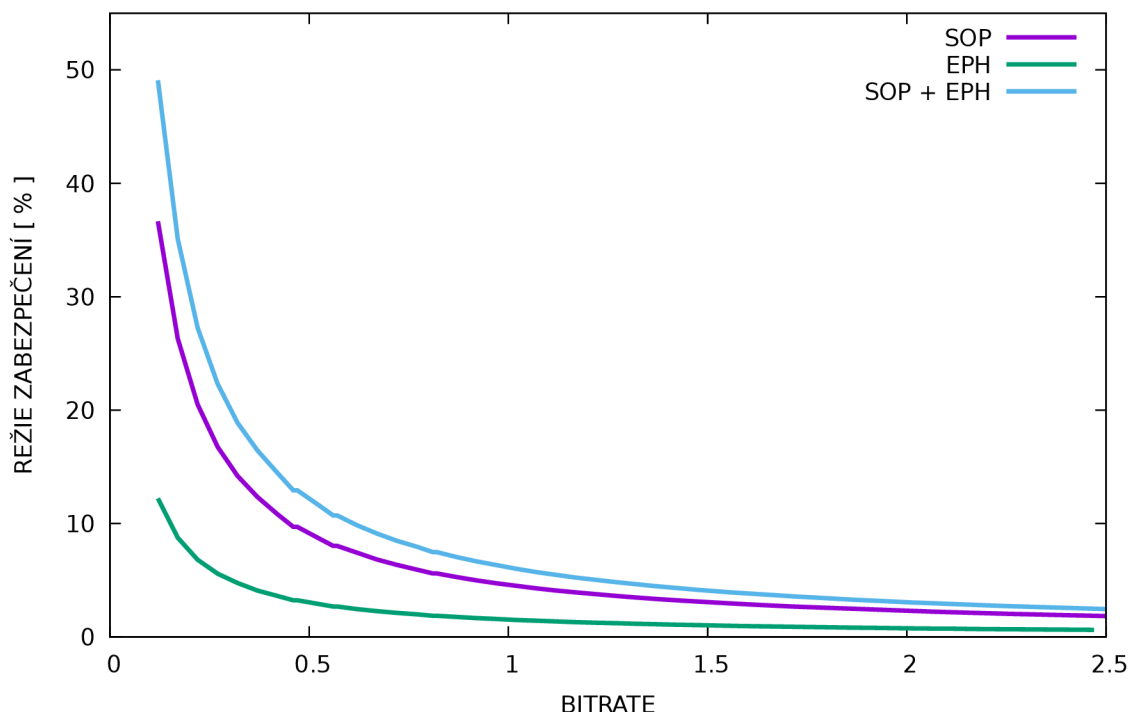
### 3.6 Analýza zabezpečení pomocí přidávaných značek

V této části je ukázán vliv značek SOP a EPH na odolnost proti chybám zpracovávaného souboru při použití knihovny Kakadu. Tyto značky budou v experimentech analyzovány samostatně i kombinovaně. Podrobně jsou popsány v kapitole 2.2. Je zde ukázáno jak se mění velikost režie při použití tohoto druhu zabezpečení v závislosti na měnícím se počtu bitů na pixel. Dále pak bude znázorněno jak se při stupňujícím se poškození souboru zvyšuje počet nedekomprimovatelných souborů.

#### Režie zabezpečení přidávaných značek

Při použití zabezpečovacích mechanismů vkládání značek před a za každý paket v datovém toku souboru se zvětšuje velikost tohoto souboru. V závislosti na počtu bitů na jeden pixel v obraze se také mění jeho velikost. Režii zabezpečení je v tomto kontextu myšleno, jak velkou procentuální část velikosti výsledného souboru zabírají právě tyto bezpečnostní prvky. Značky SOP i EPH mají konstantní velikost a jsou podrobně popsány v kapitole 2.2.

Na obrázku 3.5 je ukázáno jak se mění režie tohoto druhu zabezpečení v závislosti na zvyšujícím se počtu bitů na pixel. Režie zabezpečení je vynesena na vertikální osu grafu a na horizontální ose je vyznačen naměřený počet bitů na pixel. V grafu lze vidět tři křivky. Zelená křivka znázorňuje změnu režie při použití pouze značky EPH, která se přidává za konec každého paketu v datovém toku souboru. Z grafu je patrné, že režie tohoto zabezpečení je menší než použití ostatních variant. Průměrná nejvyšší režie se u značky EPH pohybovala okolo 12 %. Tato hodnota byla naměřena při nízkých bitratech okolo 0,12 bitů na pixel. Při bitratech okolo jednoho bitu na pixel byla průměrná režie tohoto markeru pod 1,5 % celkové velikosti souboru a při ještě vyšších bitratech byla téměř zanedbatelná.



Obrázek 3.5: Režie zabezpečení přidávaných značek



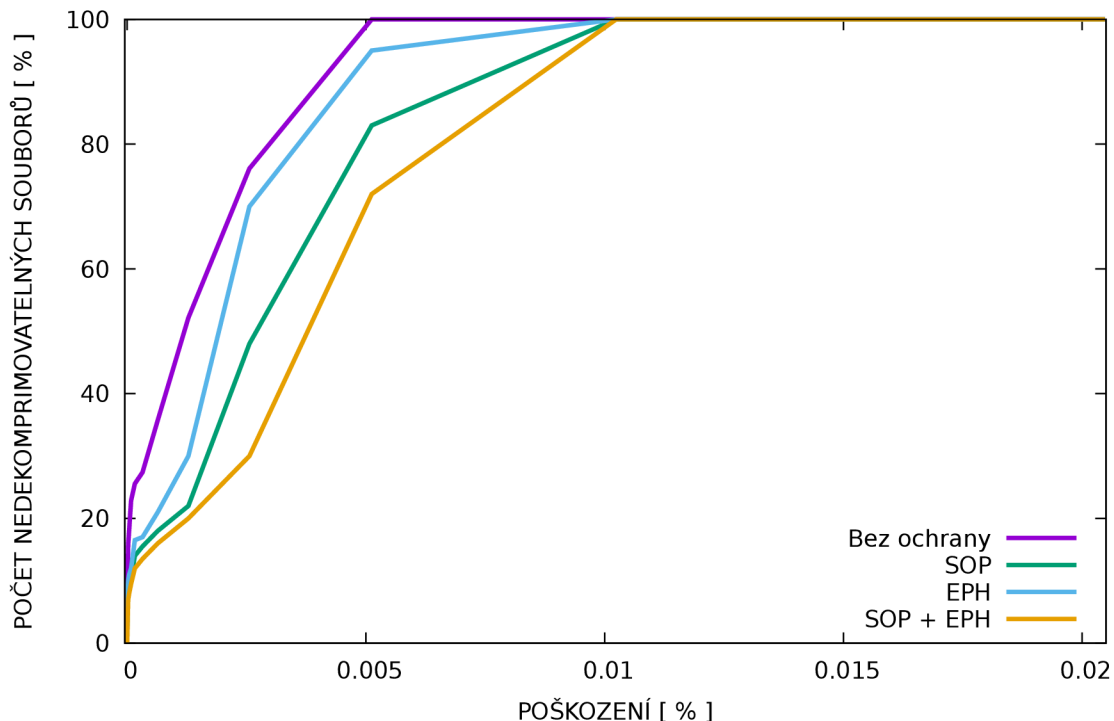
Fialová křivka v grafu znázorňuje režii použití samotného markeru SOP v závislosti na stupňujícím se počtu bitů na pixel. Charakteristika této křivky je velmi podobná té předchozí znázorňující režii značky EPH. V tomto případě je však režie několikrát vyšší. Je to způsobeno také tím, že značka SOP zabírá šest bytů v datovém toku, kdežto značka EPH pouze dva. Pro nižší bitraty okolo 0,12 bitů na pixel se průměrná režie pohybovala okolo 36,6 % z celkové velikosti výsledného souboru. Pro bitraty okolo jednoho bitu na pixel se tato režie pohybovala okolo 4,5 % a při ještě vyšších bitrate byla nižší než dvě procenta velikosti souboru.

Poslední křivka v grafu znázorňuje měnící se režii při použití varianty s SOP i EPH značkami přidanými do datového toku. Tato varianta je znázorněna modrou křivkou. Charakteristika je opět velmi podobná dvěma předchozím, ale její hodnoty jsou opět o něco vyšší. Lze pozorovat že pro nízké bitraty mohou tyto značky zabírat skoro až polovinu z celkové velikosti komprimovaného souboru. Při bitarech okolo jednoho bitu na pixel je režie této varianty okolo 6 % a pro vyšší bitraty klesne pod 3 % z celkové velikosti souboru.

Při experimentování s přidáváním značek SOP před každý paket a značek EPH za každý paket v obrazovém souboru bylo potvrzeno, že narůstá jeho velikost a pro menší soubory můžou tyto značky zabírat velkou část z velikosti souboru.

### Neúspěšně zpracované soubory

V této sekci je ukázáno kolik procent souborů se při jaké variantě zabezpečení s využitím přidaných značek nedokázalo zpracovat. Do této statistiky se počítaly nejen soubory, které se nepovedly po poškození dekomprimovat, ale také soubory, jejichž kvalita hodnocená metodikou PSNR klesla pod 35 dB v porovnání vůči originálnímu obrazu.



Obrázek 3.6: Nedekomprimovatelné soubory - zabezpečení SOP a EPH

Na obrázku 3.6 je zobrazen graf, který na vertikální ose má procentuální počet neúspěšně zpracovaných souborů a na horizontální ose je zobrazeno poškození. Poškození každého souboru začínalo na 0,000005 % bitů a s každou další iterací bylo dvojnásobné. Model poškození je podrobněji popsán v kapitole 3.1.

Fialová křivka v grafu 3.6 zobrazuje průměrný počet nedekomprimovatelných poškozených souborů bez jakéhokoliv zabezpečení. Velmi podobný průběh má modrá křivka, která ukazuje průběh neúspěšnosti pro zabezpečení pouze pomocí značek EPH. Lze vidět, že toto zabezpečení není příliš efektivní, protože výsledky analýzy této varianty jsou velmi blízké variantě bez zabezpečení.

Lépe je na tom varianta zabezpečení při použití samotného markeru SOP, což je v grafu vykresleno zelenou křivkou. Oranžová křivka zobrazuje počet neopravitelných souborů při použití varianty se zapnutím SOP i EPH markerů. Je patrné, že při analýze bylo v tomto případě dosaženo nejlepších výsledků.

Při bližším pohledu na tento graf lze také vidět, že hraniční poškození pro všechny použité typy zabezpečení je přibližně 0,01 % bitů.

### 3.7 Analýza zabezpečení při použití módů

V této části je ukázán vliv použití analyzovaných kompresních módů a jejich kombinací během komprese na zabezpečení souboru proti poškození. Jedná se o použití módů BYPASS, SEGMARK, RESET, RESTART, CAUSAL a ERTERM. Podrobně jsou popsány v kapitole 2.2. Je zde popsáno jak se mění velikost souboru při použití těchto módů. Dále pak bude znázorněno jak se na stupňujícím se poškození zvyšuje počet nedekomprimovatelných souborů při práci s knihovnou Kakadu.

#### Režie zabezpečení použitých módů

Samostatně použité módy se při analýze na změně velikosti příliš nepodílely. Jejich režie byla velice malá a u některých případů dokonce neměřitelná. Nebyly zde analyzovány všechny kombinace kompresních módů popsanych v kapitole 2.2. V některých případech užití samostatně zadaných módů měl komprimovaný soubor až neměřitelnou režii, odolnost proti poškození se také nezvětšila, a proto zde nebudou rozebírány všechny možné kombinace. Některé kompresní módy však bylo zajímavé vyšetřovat i zvlášť. Patří mezi ně například samostatné použití módu BYPASS nebo SEGMARK. V těchto experimentech se pracovalo s módy umožňující líné zpracování, paralelní zpracování či líné paralelní zpracování a jejich kombinace s markery zvyšující odolnost proti poškození. Tyto varianty jsou podrobněji popsány v kapitole 2.2. Č

Při analýze samostatně užitého módu BYPASS, který umožňuje redukovat složitost kódování se průměrná velikost souboru oproti souboru bez použití kompresních módů nepatrně zmenšila a to přibližně o 0,18 %. Jednalo se o jediný případ, kdy došlo ke zmenšení souboru. Ostatní analyzované kombinace nepatrně zvětšovaly komprimovaný soubor. Při užití samotného módu SEGMARK se průměrné zvětšení souboru pohybovalo okolo hodnoty 0,26 %. Průměrná hodnota zvětšení souboru při použití líného paralelního zpracování se vyšplhala na 0,33 %. Při variantě se současným použitím módů SEGMARK a ERTERM hodnota režie stoupla na 0,38 %. Následovala analýza kombinací paralelního zpracování, líného paralelního zpracování s použitím módu SEGMARK a paralelního zpracování ve variantě se zapnutým módem SEGMARK. Režie všech těchto tří variant se pohybovala zhruba od 0,41

do 0,45 %. Posledním experimentem v této části bylo současné zapnutí všech analyzovaných kompresních módů a průměrná reže se v tomto případě vyšphala lehce nad půl procenta.

| Použitá kombinace kompresních módů              | Velikost |
|---|----------|
| BYPASS  | -0,18 %  |
| SEGMARK   | +0,26 %  |
| BYPASS, RESET, RESTART, CAUSAL                  | +0,33 %  |
| SEGMARK, ERTERM                                 | +0,38 %  |
| RESET, RESTART, CAUSAL                          | +0,41 %  |
| BYPASS, RESET, RESTART, CAUSAL, SEGMARK         | +0,43 %  |
| RESET, RESTART, CAUSAL, SEGMARK                 | +0,45 %  |
| BYPASS, RESET, RESTART, CAUSAL, SEGMARK, ERTERM | +0,53 %  |

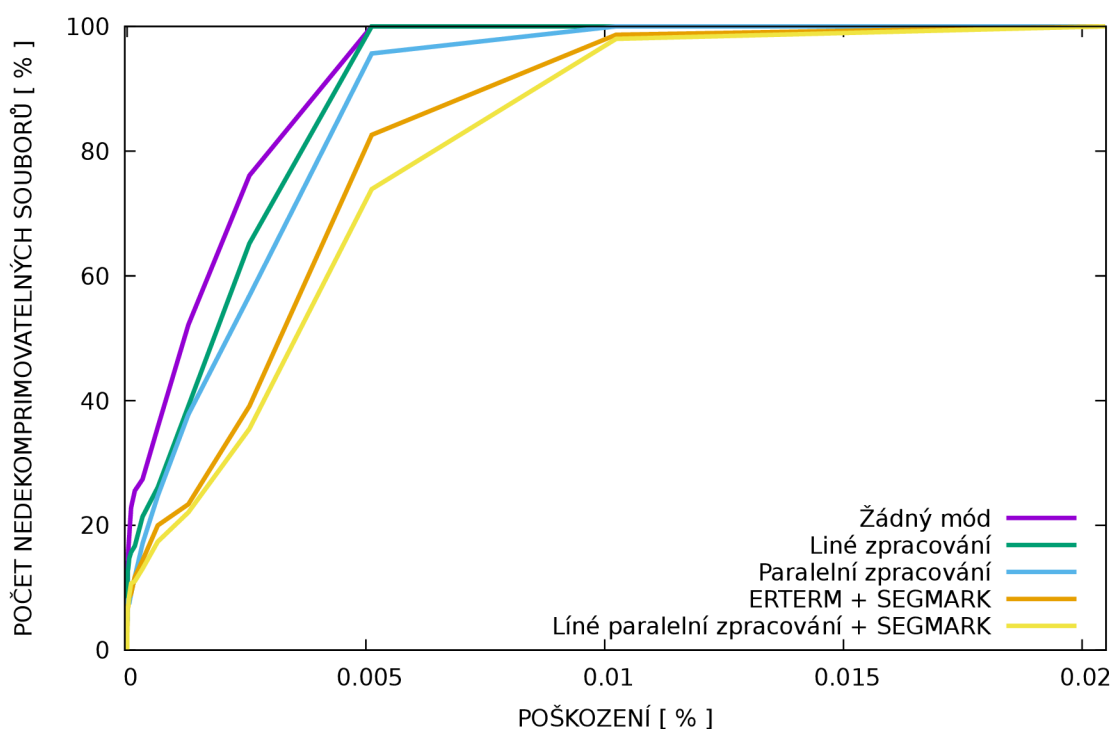
Tabulka 3.3: Shrnutí reže kompresních módů

Výsledky experimentování komprese obrazu s použitím kompresních módů jsou shrnuty v tabulce 3.3. V prvním sloupci je uvedena kombinace použitých módů a v druhém sloupci lze vidět průměrnou změnu velikosti souboru po kompresi oproti souboru komprimovaného bez použití jakýchkoliv módů. Hodnoty v této tabulce jsou seřazeny od nejnižší naměřené reže po nejvyšší.

### Neúspěšně zpracované soubory

Tato sekce je věnována analýze odolnosti proti poškození souborů zpracovávanými s kompresními módy. Stejně jako v předchozí analýze zabezpečení pomocí přidávaných značek budou za neúspěšně zpracované případy brány takové soubory, které se nepovedly po poškození dekomprimovat, ale také soubory, jejichž kvalita hodnocená metodikou PSNR klesla pod 35 dB v porovnání vůči originálnímu obrazu. Na obrázku 3.7 je na vertikální osu grafu vynesena procentuální počet nedekomprimovatelných souborů a na horizontální ose stupňující se poškození obrazu. Pro přehlednost nejsou do grafu vynášeny všechny analyzované kombinace kompresních módů, ale budou popsány v tomto textu.

Fialová křivka v tomto grafu ukazuje počet neopravitelných souborů pro žádné použité kompresní módy a tato křivka odpovídá křivce žádného zabezpečení v grafu 3.6. Samostatné použití módu BYPASS, RESET, RESTART, CAUSAL se ukázalo mít velmi malý nebo žádný vliv na zvýšení odolnosti souboru proti poškození. V těchto variantách dosahovaly naměřené hodnoty velmi podobných výsledků jako použití samotného módu BYPASS, jehož odolnost vůči stupňujícímu se poškození zobrazuje zelená křivka v grafu 3.7. Použití samotného ERTERM dosáhlo podobných výsledků jako použití kombinace módů pro paralelní zpracování, což jsou RESET, RESTART a CAUSAL. Výsledky z naměřených hodnot pro paralelní režim jsou v grafu zobrazeny modrou křivkou. Lepších výsledků bylo dosaženo za použití samotného módu SEGMARK nebo kombinace BYPASS a SEGMARK. Ještě lepších výsledků v odolnosti proti poškození dosáhla kombinace módů ERTERM a SEGMARK bez použití módu BYPASS. Pro přehlednost je do grafu vynesena pouze nejlepší z těchto kombinací a je znázorněna oranžovou křivkou. Při zapnutí všech analyzovaných kompresních módů křivka odolnosti proti poškození nepatrně klesla oproti kombinaci s nejlepšími výsledky v analýze odolnosti proti poškození. Touto kombinací se ukázalo být použití líněného paralelního zpracování se současně zapnutým módem SEGMARK. Výsledky analýzy této varianty zabezpečení jsou v grafu naznačeny žlutou křivkou.



Obrázek 3.7: Počet nedekomprimovatelných souborů - kompresní módy

V porovnání s grafem 3.6, kde se ukázalo být poškození 0,01 % bitů v souboru jako hraniční pro všechny analyzované kombinace komprese, lze v grafu 3.7 vidět, že u některých kombinací použití kompresních módů se podařilo opravit soubory i při tomto poškození, což ilustrují žlutá a červená křivka.

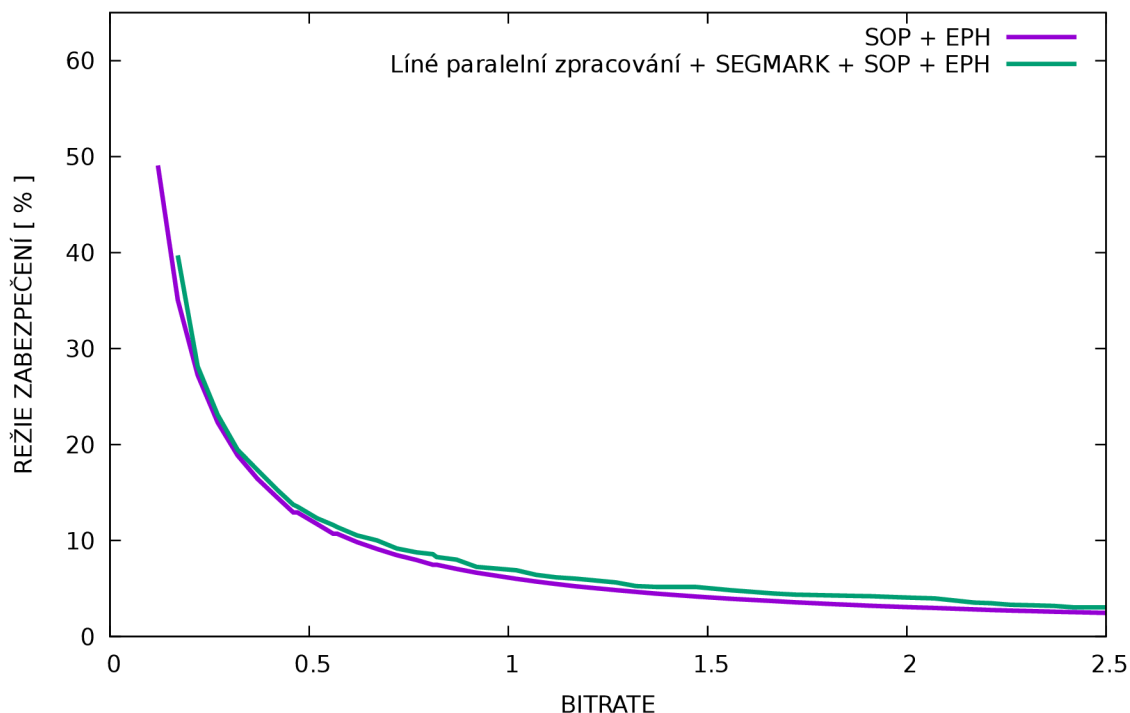
### 3.8 Analýza zabezpečení při použití značek i módů

V této části je popsán vliv použití nejlepší varianty kompresních módů a nejlepší varianty přidaných značek zvyšujících odolnost proti poškození. Jedná se o variantu s použitím módů BYPASS, RESET, RESTART, CAUSAL a SEGMARK najednou současně s vložením SOP a EPH značek do datového toku. Že právě tyto dvě varianty vykazují nejlepší odolnost proti poškození je popsáno v kapitolách 3.6 a 3.7. Podrobně jsou tyto zabezpečovací prvky popsány v kapitole 2.2. Je zde ukázáno jak se mění velikost režie při použití tohoto druhu zabezpečení v závislosti na měnícím se počtu bitů na pixel. Dále pak bude znázorněno jak se na stupňujícím se poškození zvyšuje počet nedekomprimovatelných souborů při použití knihovny Kakadu.

#### Režie zabezpečení při použití módů i markerů

V této části analýzy bude vizualizována naměřená režie zabezpečení, při použití nejlepších možností komprese popisovaných v předchozích částech této práce. Jedná se o současné použití SOP i EPH značek a použití všech kompresních módů popsanych v kapitole 2.2, konkrétně BYPASS, RESET, RESTART, CAUSAL a SEGMARK.

Naměřená průměrná režie této varianty komprese je zobrazena zelenou křivkou v grafu na obrázku 3.8. Pro srovnání je v grafu zobrazena také křivka režie zabezpečení při SOP i EPH markerů. Ta je v tomto grafu zobrazena fialovou barvou a odpovídá křivce z grafu 3.5. Křivka znázorňující průměrnou režii při použití nejlepší kombinace kompresních módů není v grafu zahrnuta, jelikož jsou její hodnoty velmi nízké. Analýza režie použití kompresních módů je popsána v kapitole 3.7 a tabulce 3.3.



Obrázek 3.8: Režie zabezpečení při použití kompresních módů i markerů

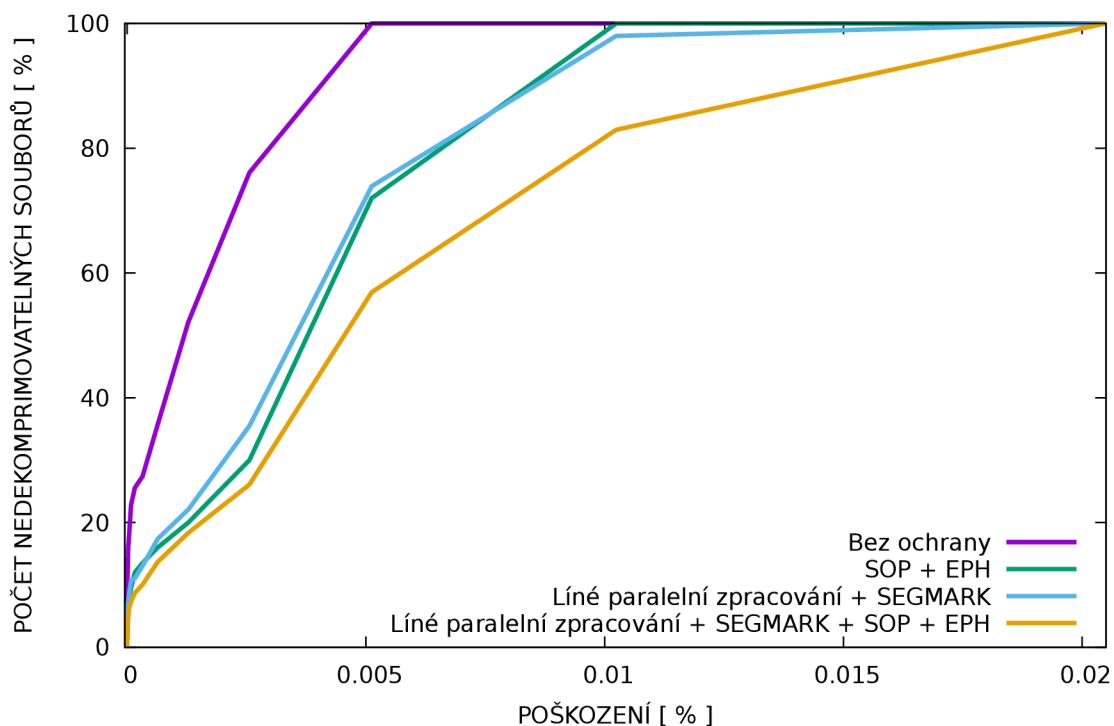
V grafu lze vidět, že obě křivky nezačínají ve stejném bodě. Křivka zobrazující klesající průměrnou režii zabezpečení pro obrazy komprimované se SOP a EPH značkami má počáteční bod posunutý o desetinu bitrate k vertikální ose. To, že obě křivky nezačínají ve stejném bitrate je způsobeno tím, že použitím určitého zabezpečení se vytvoří určitý spodní limit bitrate. Při použití SOP i EPH značek a nejlepší kombinace analyzovaných kompresních módů je tato spodní hranice vyšší než při použití pouze SOP a EPH. Například při explicitním použití obou druhů zabezpečení a požadovaném například bitrate 0,02 bitů na pixel, nebude mít nikdy výsledný zkomprimovaný obraz tuto hodnotu bitrate, ale vyšší. Zelená křivka vizualizující použití nejlepší variantu použití kompresních módů a současně SOP a EPH značek má podobnou klesavou charakteristiku jako křivka zobrazující klesající režii pro SOP a EPH. Z grafu je zřetelné, že průměrná hodnota naměřené režie je pro tuto variantu větší pro celý analyzovaný interval počtu bitů na pixel. Zvětšení souboru při použití tohoto druhu zabezpečení není ovšem tak výrazné a pohybuje se okolo 0,8 % než při zabezpečení pouze SOP a EPH značkami.

## Neúspěšně zpracované soubory při použití módů i markerů

Tato sekce je věnována analýze odolnosti proti stupňujícímu se poškození pro soubory komprimované současně s nejlepší kombinací analyzovaných kompresních módů a se SOP a EPH značkami. Blíže jsou všechny použité varianty komprese popsány v kapitole 2.2. Výsledky tohoto experimentu jsou vizualizovány v grafu na obrázku 3.9. Stejně jako v přechozích analýzách pro použití pouze SOP a EPH značek nebo pouze kompresních módů, je i v tomto případě na vertikální osu grafu vyneseno procentuální počet neopravitelných souborů a na horizontální ose je procentuálně vyjádřeno poškození souboru. Poškození opět začíná na hodnotě 0,000005 % a s každou další iterací je dvojnásobné.

Počet neopravitelných souborů pro kompresní variantu, kde nebylo použito žádné zvláštní zabezpečení, je v tomto grafu znázorněn stoupající fialovou křivkou. Lze vidět, že pro soubory bez jakýchkoli zabezpečovacích mechanismů se hraniční poškození pohybuje přibližně okolo hodnoty 0,005 % poškozených bitů.

Pro srovnání jsou zde přidány také křivky získané z předchozí analýzy. Zelená křivka v grafu 3.9 vizualizuje počet neopravitelných souborů v závislosti na zvyšujícím se poškození pro soubory komprimované s přidávanými SOP a EPH značkami. Je to křivka s nejlepšími výsledky z analýzy zabezpečení SOP a EPH markerů z kapitoly 3.6. Modrá křivka v tomto grafu vizualizuje variantu s nejlepší odolností proti poškození souboru pro kombinace analyzovaných kompresních módů. Touto kombinací se ukázalo být zapnutí líného paralelního režimu zpracování a současné použití módu SEGMARK. Tato křivka odpovídá experimentu popsanému v kapitole 3.7.



Obrázek 3.9: Počet nedekomprimovatelných souborů - kompresní módy a markery

Nejdůležitější částí tohoto grafu je křivka zobrazující počet neopravitelných poškozených obrazů komprimovaných za použití SOP a EPH značek, líného paralelního režimu

zpracování a módu SEGMARK. Tato křivka je v grafu vykreslena oranžovou barvou. Je vidět, že hodnoty této varianty zabezpečení zobrazené v grafu 3.9 dosahují lepších výsledků než ostatní varianty. Nepodařilo se úspěšně rekonstruovat menší procento poškozených obrazů pro všechny stupně poškození. Při poškození 0,01024 % souboru vykazovaly předchozí dvě varianty zabezpečení průměrně 98 až 100 % selhání při opravě poškozeného souboru. Při variantě ilustrované žlutou křivkou v tomto grafu se procento neúspěšně zpracovaných souborů zmenšilo na přibližně 82 % při stejném poškození.

### 3.9 Zabezpečení a režie v souvislosti

Ve výše uvedených kapitolách je ukázáno jak se mění režie použitého zabezpečení v závislosti na zvyšujícím se počtu bitů na pixel a jak moc jsou tyto varianty zabezpečení účinné při ochraně dat při stupňujícím se poškození. V této kapitole budou různá zabezpečení provnávána v souvislosti s jejich režii. Cenou je v tomto smyslu míněna změna velikosti souboru s přidanými značkami či použitými módy během komprese. Aby se úroveň ochrany proti poškození dala vyjádřit jedním číslem, je tato vlastnost určitého druhu komprese vy počítávána jako průměrný počet nedekomprimovatelných souborů. Počet neopravitelných souborů byl průměrován na intervalu od 0 do 0,02 % poškozených bitů v souboru. Podobným způsobem byla vyhodnocována také režie určitého druhu zabezpečení. Režie byla agregována na intervalu od 0,1 do 2,5 bitů na pixel.

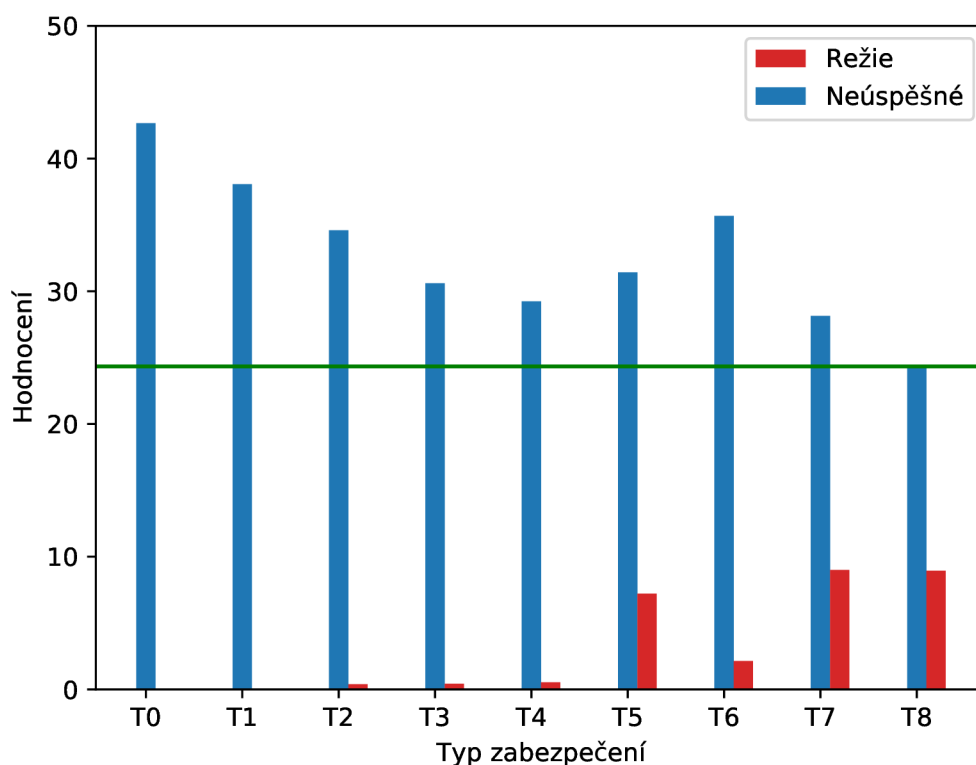
V prvním sloupci tabulky 3.4 označeném jako T0 je označeno hodnocení varianty s použitím žádného zvláštního zabezpečení. Jelikož nebyly použity žádné zvláštní mechanismy ovlivňující odolnost proti poškození, režie není v tomto případě uvažována a u ostatních variant T1 až T8 je změna velikosti souboru brána vzhledem k variantě T0. V této variantě byla spočítána hodnota pro průměrný počet neúspěšně zpracovaných souborů 42,66 %.

| S  | Varianta komprese                               | Neúspěšné | Režie   |
|----|---|-----------|---------|
| T0 | Bez zabezpečení                                 | 42,66 %   | -       |
| T1 | Líné zpracování                                 | 38,07 %   | -0,18 % |
| T2 | Paralelní zpracování                            | 34,60 %   | +0,39 % |
| T3 | ERTERM + SEGMARK                                | 30,61 %   | +0,43 % |
| T4 | Líné paralelní zpracování + SEGMARK             | 29,24 %   | +0,53 % |
| T5 | SOP   | 31,43 %   | +7,22 % |
| T6 | EPH   | 35,68 %   | +2,14 % |
| T7 | SOP + EPH                                       | 28,14 %   | +9,00 % |
| T8 | Líné paralelní zpracování + SEGMARK + SOP + EPH | 24,34 %   | +9,94 % |

Tabulka 3.4: Režie a efektivita zabezpečení

Soupec označený jako T1 ukazuje hodnocení varianty při použití samotného módu BY-PASS, tedy líného zpracování. V tomto jediném případě docházelo ke zmenšení souboru a to v průměru o 0,18 %. Průměrný počet neopravitelných souborů byl 38,07 %. Hodnoty získané experimentováním s variantou použití paralelního zpracování ukazuje sloupec T2. V tomto případě se počet neúspěšných souborů dostal na 34,60 % s celkovou průměrnou naměřenou hodnotou zvětšení souboru o 0,39 %. O trochu větší průměrná režie byla zaznamenána při použití varianty současného zapnutí módů ERTERM a SEGMARK. Její hodnota se dostala na +0,43 % a počet neopravitelných souborů na 30,61 %. Tato varianta je v tabulce 3.4 znázorněna sloupcem T3. Sloupec T4 obsahuje průměrované naměřené hodnoty pro

variantu s použitím líného paralelního zpracování a současného použití kompresního módu SEGMARK. Tato varianta se ukázala být nejúčinnější v odolnosti proti poškození v experimentování pouze s kombinacemi kompresních módů. Průměrný počet neopravitelných souborů se dostal na 29,24 % s průměrným zvětšením souboru o 0,53 %. Dalšími analyzovanými variantami bylo samostatné použití SOP a EPH značek, jejichž hodnoty jsou v tabulce uvedeny pod značením T5 a T6. Průměrný počet neopravitelných souborů u použití EPH značky dosáhl hodnoty 35,68 % a režie tohoto zabezpečení se pohybovala okolo 2,14 %. V odolnosti proti poškození lepších výsledků dosahovala použití samotného SOP markeru, ale za cenu vyšší režie, která se pohybovala okolo 9 %. O necelé procento horší průměrné režie bylo dosaženo ve variantě s použitím líného paralelního zpracování, módu SEGMARK a značek SOP i EPH. Hodnota průměrného počtu neopravitelných souborů se však vyšplhala na 24,34 %, což je nejmenší hodnota ze všech analyzovaných variant. Tento výsledek je v tabulce 3.4 vidět pod označením T8 a je zvýrazněn modře.



Obrázek 3.10: Režie a efektivita zabezpečení

Počet neopravitelných souborů vizualizují modré části v grafu na obrázku 3.10. Průměrná režie je v tomto grafu znázorněna červenou částí pro každou variantu zabezpečení. Sloupce s označením T0 až T8 korespondují s hodnotami v tabulce 3.4 a zelená čára ukazuje nejlepší dosažený výsledek v počtu neopravitelných souborů, který byl dosažen při použití kompresní varianty T8.



## 3.10 Srovnání bezpečnosti JPEG 2000 a CCSDS 122.0

Formát JPEG 2000 byl detailně popsán v kapitole 2.2. V této kapitole bude popsán formát CCSD 122.0, který bývá používán při kompresi obrazů naskenovaných vesmírnými družicemi. Budou zde porovnávány formáty JPEG 2000 a CCSDS 122.0 v odolnosti proti poškození obrazových souborů. Pro analýzu byl vytvořen jiný model poškození než u analýzy bezpečnosti pouze formátu JPEG 2000. Tento model je popsán v kapitole 3.10. Při analýze byla použita knihovna BPE<sup>1</sup> pro formát CCSDS 122.0 a knihovna Kakadu pro JPEG 2000.

### CCSDS 122.0

CCSDS (Consultative Committee for Space Data Systems) je mezinárodní organizace založená v roce 1982, která se stará o vývoj standardů pro data pro použití na vesmírných zařízeních [4]. Standard CCSDS 122.0 je standard pro obrazová data pořízená z kosmických lodí a družic. Stejně jako standard JPEG 2000 je založen na diskretní vlnkové transformaci a stejně tak podporuje ztrátovou i bezztrátovou kompresi obrazu. Dokáže zpracovávat nekonečně dlouhé snímky, které lze získat například skenování povrchu Země družicí z oběžné dráhy.

Zatímco formát JPEG 2000 dokáže zpracovávat vícekanálové obrazy, CCSDS 122.0 pokrývá kompresi dat pouze ve stupních šedi, a proto budou v této kapitole analyzována data pouze na datech ve stupních šedi. JPEG 2000 je velmi komplexní na zpracování což zapříčiňují několikanásobné diskretní vlnkové transformace a ukládání datového toku do vrstev pro postupné zvyšování kvality obrazu. Tento formát není příliš vhodný pro použití na vesmírných systémech. Tyto systémy potřebují efektivnější algoritmy, které dokáží zpracovávat obraz při vyšších kompresních poměrech a dokáží generovat obrazy vhodné pro použití například pro mapové aplikace. U formátu CCSDS 122.0 byl proveden kompromis mezi výkonem komprese a složitostí se zvláštním důrazem na aplikace kosmických lodí.

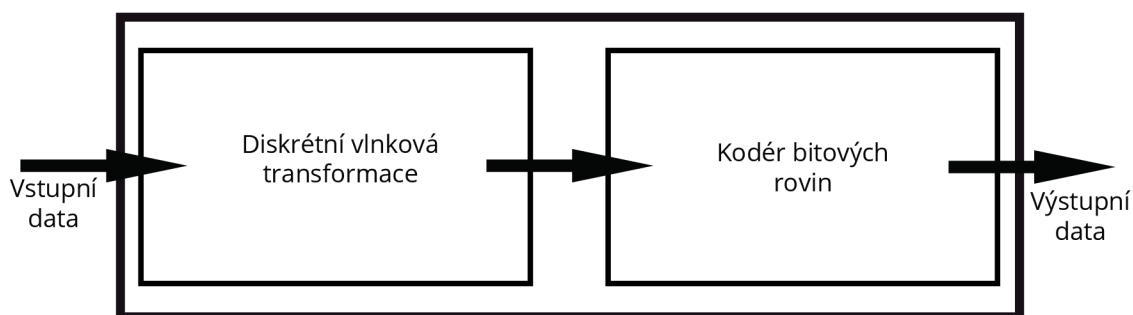
Kompresi se skládá ze dvou základních funkčních bloků, které lze vidět na obrázku 3.11. Vstupní data postupují do prvního bloku, ve kterém se provádí diskretní vlnková transformace DWT, která dekoreluje vstupní data. Takto zpracovaná data postupují do druhé funkčního bloku, kterým je kodér bitových rovin. Výstupem jsou data zpracována do segmentů. Každý segment je uvozen hlavičkou a následován pole obrazových dat.

Podle případových studií [10] zaměřených na kompresi dat pro vysoké rozlišení dosahuje CCSDS 122.0 lepších výsledků než formát JPEG. Komplexnost implementace je nižší ve srovnání se standardem JPEG 2000 a nedosahuje výrazně horších výsledků, a proto je vhodný pro použití na vesmírných stanicích.

### Model poškození

Pro analýzu odolnosti proti poškození byl v tomto případě použit jiný model poškození než model popsáný v kapitole 3.1. Byl vytvořen model, který simuluje poškození souvislého bloku dat v souboru. Při použití se od určité pozice v souboru přepisovalo několik bytů nulami. Poškození bylo testováno v rozmezí 0,3 až 60 % z celkové velikosti souboru, což mohlo vytvořit i několik tisíc přeepsaných bytů v souboru.

<sup>1</sup>CCSDS 122.0-B-2 Implementace - <http://hyperspectral.unl.edu/index.html>



Obrázek 3.11: Funkční bloky komprese s CCSDS 122.0 [10]

### Výsledky analýzy odolnosti proti poškození

Soubory zpracovávané podle standardu CCSDS 122.0 by měly disponovat vyšší odolností proti poškození než u formátu JPEG 2000. U přenosu obrazových souborů z vesmírných družic se počítá, že soubor potenciálně nekonečně široký může být poškozen souvislým blokem nečitelných dat způsobeným například rušením při přenosu. Aby se tento soubor podařilo úspěšně zpracovat i při vyšším poškození, pracují dekompresní programy tak, že poškozenou část do určité míry ignorují. Pro potvrzení této hypotézy byl použit model poškození popsany v kapitole 3.10. Analyzované soubory ve formátu JPEG 2000 použité při tomto experimentu byly komprimovány s líným paralelním režimem, módem SEGMARK a přidávanými SOP a EPH značkami, což zajišťuje nejvyšší odolnost proti poškození, jak je popsáno výše v kapitolách o zabezpečení formátu JPEG 2000. U souborů zpracovávaných knihovnou BPE ve formátu CCSDS 122.0 se měnil počet segmentů v obraze definovaný explicitně při kompresi. Obrazy byly komprimovány s bitratem rovným jedné pro obě knihovny.

Při poškození náhodně vybraných bitů v souboru se pro maximální zabezpečení souboru JPEG 2000 ukázalo hraniční poškození přibližně 0,02 % z celkové velikosti souboru. Při experimentování s jedním poškozeným souvislým blokem dat byl limit pro úspěšnou dekompresi souboru posunut na 0,7 %. Pro soubor o velikosti 24,5 MB a rozlišení 110 Mpx se jednalo o přibližně 35 krát větší poškození, ze kterého se soubor ve formátu JPEG 2000 při použití knihovny Kakadu dokázal rekonstruovat.

Ve druhé části tohoto experimentu se testovala odolnost proti poškození formátu CCSDS 122.0 za použití knihovny BPE. Ve výchozím nastavení je počet bloků v každém segmentu nastaven na hodnotu 256. Při experimentech s rozsahem poškození 0,7 %, které se ukázalo pro formát JPEG 2000 jako hraniční, se všechny poškozené soubory ve formátu CCSDS 122.0 podařilo dekomprimovat a jejich podobnost vůči originálnímu obrazu hodnocena metrikou PSNR dosahovala průměrných hodnot okolo 39 dB. Stupňováním poškození klesala kvalita úspěšně dekomprimovaných obrazů. Hraničním poškozením se ukázalo poškození 23,20 %, kdy se obraz pomocí knihovny BPE povedlo rekonstruovat. Průměrné PSNR se okolo 30 dB. Zvýšením počtu bloků v každém segmentu se zvýšila schopnost úspěšně dekomprimovat poškozený obraz, ovšem za cenu kvality obrazu. Kvalita úspěšně dekomprimovaných poškozených obrazů byla nižší než u použití menšího počtu bloků v každém segmentu.

| Poškození | JP2   | B256  | B1024 | PSNR JP2 | PSNR B256 | PSNR B1024 |
|-----------|-------|-------|-------|----------|-----------|------------|
| 0,70 %    | OK    | OK    | OK    | 41,23    | 39,59     | 27,84      |
| 1,22 %    | Chyba | OK    | OK    | -        | 37,79     | 27,82      |
| 6,10 %    | Chyba | OK    | OK    | -        | 33,50     | 27,72      |
| 12,20 %   | Chyba | OK    | OK    | -        | 33,17     | 26,65      |
| 18,31 %   | Chyba | OK    | OK    | -        | 32,75     | 24,37      |
| 21,36 %   | Chyba | OK    | OK    | -        | 31,65     | 23,64      |
| 23,19 %   | Chyba | OK    | OK    | -        | 31,22     | 23,13      |
| 23,80 %   | Chyba | Chyba | OK    | -        | -         | 23,02      |
| 24,41 %   | Chyba | Chyba | OK    | -        | -         | 22,27      |
| 36,62 %   | Chyba | Chyba | OK    | -        | -         | 19,56      |
| 48,82 %   | Chyba | Chyba | OK    | -        | -         | 16,84      |
| 61,03 %   | Chyba | Chyba | OK    | -        | -         | 13,66      |

Tabulka 3.5: Odolnost CCSDS 122.0 proti poškození

Výsledky těchto experimentů s formáty JPEG 2000 a CCSDS 122.0 shrnuje tabulka 3.5. V prvním sloupci je vzhledem k velikosti souboru procentuálně vyjádřena velikost poškozeného bloku. Sloupec označený zkratkou JP2 ukazuje úspěšnost dekomprese pro určitou velikost poškození souboru komprimovaného do formátu JPEG 2000 s nejvyšším možným dosaženým zabezpečením a doporučené komprese dle Národní digitální knihovny. Totéž ukazují sloupce B256 a B1024 pro formát CCSDS 122.0 s 256 či 1024 bloky v každém segmentu při určitém poškození. Sloupce PSNR JP2, PSNR B256 a PSNR B1024 ukazují PSNR úspěšně dekomprimovaného poškozeného souboru.

## Shrnutí

V této kapitole byla srovnávána odolnost proti poškození formátu JPEG 2000 a CCSDS 122.0 za pomoci knihoven Kakadu a BPE. CCSDS 122.0 má oproti JPEG 2000 značná omezení, mezi které patří například zpracovávání pouze jednokanálových dat a nižší kvalita zpracovaných obrazů. Formát CCSDS 122.0 však disponuje větší odolností proti chybám i v případě velmi poškozeného souboru. Kvalita rekonstruovaného obrazu je však velmi nízká. Hraniční velikost poškozeného místa pro úspěšnou dekompresi se u formátu JPEG 2000 a knihovny Kakadu ukázala na 0,7 % z celkové velikosti souboru. Pro formát CCSDS 122.0 je tato hranice několikanásobně vyšší v závislosti na nastavení počtu bloků v každém segmentu.

## Kapitola 4

# Závěr

V této práci je analyzováno zabezpečení formátu JPEG 2000 proti poškození. Analýza byla nejprve prováděna za pomoci dvou knihoven - Kakadu a OpenJPEG. Jelikož se knihovna OpenJPEG ukázala jako nevhodná pro další analýzu, práce je věnována zpracování obrazu pouze knihovnou Kakadu pro ztrátovou kompresi. Součástí této studie je framework, který byl vytvořen za účelem zpracování obrazů z připravené datové sady a vizualizaci výsledků.

Odolnost proti poškození byla testována ve třech variantách zabezpečení. První varianta zkoumala zvýšení odolnosti proti poškození při vložení určitých značek okolo paketů v datovém toku. Druhá se zaměřovala na analýzu zvýšení bezpečnosti za použití kompresních módů a poslední kombinovala předchozí dvě varianty zabezpečení.

Přidávání značek před a za pakety v souboru zvýšilo odolnost proti chybám a pomáhalo při rekonstrukci obrazu, ale cenou bylo zvětšení souboru. Jelikož mají tyto značky konstantní velikost, mohly zabírat i polovinu velikosti souboru komprimovaného s malým počtem bitů na pixel. Při menších kompresních poměrech nebyla tato režie tolik významná. Použití určitých kombinací kompresních módů při zpracování obrazu také zvyšovalo odolnost souboru proti poškození. Oproti zabezpečení s pouze přidanými značkami však měly mnohem menší režii.

Nejlepších výsledků v odolnosti proti chybám a kvalitě zpracovaných obrazů bylo dosaženo za současného použití líného paralelního režimu zpracování, módu SEGMARK a SOP a EPH značek. Při použití tohoto druhu ochrany obrazového souboru však narostla hodnota průměrného zvětšení komprimovaného souboru.

Pro dosažení nejlepší odolnosti proti chybám bych doporučil použití vkládání SOP a EPH značek při kompresi a současné použití líného paralelního režimu spolu se zapnutým módem SEGMARK. V případě, že je hlavním cílem dosáhnout co nejlepších kompresních poměrů a zároveň dosažení dobrého zabezpečení, tak bych doporučil použití kompresní varianty podle nastavení Národní digitální knihovny s vypnutím přidaných markerů, ale zapnutím líného paralelního zpracování a módu SEGMARK.

Závěrem práce byla srovnávána odolnost proti zabezpečení formátu JPEG 2000 a formátu CCSDS 122.0. Druhý ze jmenovaných dosahoval v experimentech mnohem lepších výsledků v odolnosti proti poškození dat. Oproti JPEG 2000 byly výsledky v bezpečnosti až několikanásobně lepší. Na druhou stranu mezi výhody JPEG 2000 oproti CCSDS 122.0 patří možnost zpracovávání vícekanálových dat a vyšší kvalita výsledných obrazů.

# Literatura

- [1] D. S. Taubman and M. W. Marcellin: *JPEG2000 Image Compression Fundamentals, Standards and Practice*, Springer Science+Business Media, LLC., 2002, ISBN 978-0792375197.
- [2] Anisimova, J. P. P., E.; Bednář: *Zpracování obrazu pomocí vlnkové transformace. Electro revue*. 2013, ročník 15, [navštíveno 21.12.2018].
- [3] Bařina, P., D.; Zemčik: *Multimédia. 2013, (studijní opora, FIT VUT v Brně)*. 2017.
- [4] The consultative committee for space sata systems *Image data compression - CCSDS 122.0-B-2*. 2017, washington, DC, USA.  
URL <https://public.ccsds.org/Pubs/122x0b2.pdf>
- [5] Šimek, J.: *Využití pokročilých objektivních kritérií hodnocení při kompresi obrazu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010, vedoucí diplomové práce Ing. Jan Malý.
- [6] D. S. Taubman and M. W. Marcellin: *JPEG2000: standard for interactive imaging*. Vol. 90, no. 8, pp. 1336–1357, Aug. 2002. doi:10.1109/JPROC.2002.800725.
- [7] Milan Sonka, V. H.; Boyle, R.: *Image Processing, Analysis, and Machine Vision. (2nd Edition)*. The University of Iowa, PWS, 1998, ISBN ISBN 0-534-95393-X.
- [8] Otemuyiwa, P.: *The Great JPEG 2000 Debate: Analyzing the Pros and Cons to Widespread Adoption Applications*. 2017, [Online; navštíveno 6.1.2019].  
URL [https://cloudinary.com/the\\_great\\_jpeg\\_2000\\_debate\\_analyzing\\_the\\_pros\\_and\\_cons\\_to\\_widespread\\_adoption](https://cloudinary.com/the_great_jpeg_2000_debate_analyzing_the_pros_and_cons_to_widespread_adoption)
- [9] Peter Schelkens, T. E., Athanassios Skodras: *The JPEG 2000 Suite*. 2002, ISBN 978-0470721476.
- [10] Preeti Panjwani, B. G. K. P. K. S., Arvind K. Singh: *Evaluation of CCSDS compression for high resolution imagery*. Indie, 2011.
- [11] V. Sanchez, A. B.: *The JPEG2000 Image Compression Standard*. University of Alberta, [Online; navštíveno 6.1.2019].  
URL [https://webdocs.cs.ualberta.ca/~anup/Courses/604/NOTES/slide\\_jpeg2000.pdf](https://webdocs.cs.ualberta.ca/~anup/Courses/604/NOTES/slide_jpeg2000.pdf)
- [12] Vychodil, B.: *Standardy pro obrazová data*. [Online; navštíveno 21.12.2018].  
URL <https://www.ndk.cz/standardy-digitalizace/standardy-pro-obrazova-data>

## Příloha A

# Obsah přiloženého paměťového média

- Vytvořený framework pro analýzu zabezpečení formátu JPEG 2000
- Používané knihovny Kakadu, OpenJpeg a implementace standardu CCSDS 122.0-B-2 (zdrojové soubory knihovny BPE)
- Datová sada obsahující 111 fotografií rozdělené do 3 kategorií, na kterých byla analýza prováděna
  - Běžné fotografi - 100 souborů v rozlišení 5 až 15 Mpx
  - Skenované dokumenty - 10 souborů v rozlišení 16 až 80 Mpx
  - Mapové podklady - 1 soubor s rozlišením 110 Mpx
- Data, na kterých bylo prováděno srovnání formátu JPEG 2000 a CCSDS 122.0
- Výsledky z provedené analýzy, datové soubory a vytvořené grafy.