

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

BRUNO MARQUES BASTOS

**Low-Complexity Transform-Quantization
Pair for 360° Image Compression**

Work presented in partial fulfillment of the
requirements for the degree of Bachelor in
Computer Science

Advisor: Prof. Dr. Thiago Lopes Trugillo da
Silveira

Porto Alegre
January 2025

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Prof^a. Márcia Cristina Bernardes Barbosa

Vice-Reitor: Prof. Pedro de Almeida Costa

Pró-Reitora de Graduação: Prof^a. Nádyá Pesce da Silveira

Diretor do Instituto de Informática: Prof. Luciano Paschoal Gaspary

Coordenador do Curso de Ciência de Computação: Prof. Marcelo Walter

Bibliotecário-chefe do Instituto de Informática: Alexsander Borges Ribeiro

*“Happiness can be found even in the darkest of times,
if one only remembers to turn on the light.”*

— PROF. ALBUS DUMBLEDORE

ACKNOWLEDGMENTS

Although this work is written in English, I have decided to write my acknowledgments in Portuguese to sincerely express my gratitude to those who have been essential throughout this journey.

Gostaria de expressar minha profunda gratidão à minha namorada, Patrícia de Mari Loss, que sempre me incentivou e me apoiou incansavelmente na busca pelos meus sonhos, tornando tudo isso possível. Agradeço também aos meus familiares – mãe, irmãos, tios(as) e primos(as) – que, de alguma forma, estiveram presentes nos momentos em que precisei de uma alegria ou um apoio. Em especial, faço uma homenagem à minha avó, Maria Helena Marques Bastos, cuja presença foi essencial para o meu crescimento pessoal. Também sou grato àqueles que, apesar de não serem parentes por laços sanguíneos, fazem parte da minha família do coração.

Minha sincera gratidão aos "professores da vida", que de alguma maneira me inspiraram, incentivaram e me motivaram a seguir o caminho da ciência. Agradeço especialmente aos professores que estiveram ao meu lado em um dos momentos mais difíceis deste percurso: Professor Dr. Luciano Gaspar, Professora Dra. Érika Fernandes, Professor Dr. Marcelo Walter e Luís Otávio Luz Soares, pessoas que admiro profundamente e considero grandes amigos. Um agradecimento mais que especial ao meu orientador, Dr. Thiago Lopes Trugillo da Silveira, que, com dedicação e sabedoria, esteve sempre ao meu lado, me inspirando, orientando e desafiando a crescer tanto pessoal quanto academicamente.

Agradeço também aos Professores Dr. Cláudio Machado Diniz, Dr. Mateus Grellert da Silva e Dra. Annabeth Petry Radunz, pela generosidade em aceitarem fazer parte da banca avaliadora deste trabalho, contribuindo de forma valiosa para o meu desenvolvimento.

Sou profundamente grato aos órgãos de fomento à pesquisa, como o CNPq, que, durante minha trajetória acadêmica, ofereceram o suporte financeiro necessário para o desenvolvimento de projetos e pesquisas. Agradeço também à Pró-Reitoria de Assuntos Estudantis (PRAE) pelo apoio e auxílios fundamentais para o meu percurso acadêmico.

Por fim, deixo meu agradecimento a todos aqueles que, de alguma forma, contribuíram para que eu pudesse concluir esta etapa com sucesso, mesmo que não mencionados nominalmente aqui.

ABSTRACT

This work presents a novel approach for approximate 360° image compression by combining latitude-adaptive quantization with low-complexity transformation and quantization matrices. The method is specifically designed to address the unique challenges posed by the equirectangular projection (ERP) format, which inherently suffers from redundancies, particularly near the poles. The proposed solution significantly reduces computational complexity by eliminating the need for multiplication operations and relying exclusively on addition and bit-shift operations, making it highly suitable for resource-constrained devices. A comprehensive analysis of several low-complexity transformations and quantization matrices is presented, showcasing the flexibility and plug-and-play nature of our method, which can be seamlessly integrated into existing compression pipelines. The performance of the proposed methods is evaluated on a widely used benchmark dataset—with 360° images in 4K, 6K, and 8K resolutions—and compared with their exact counterparts. The results reveal that the proposed approximate compression technique substantially reduces computational demand while maintaining competitive quality, with peak signal-to-noise ratio (PSNR) losses not exceeding 4 dB at lower bitrates. These findings highlight the potential of the method to enable efficient 360° media processing in applications where computational resources and energy efficiency are critical.

Keywords: 360° image compression. low-complexity transform and quantization. latitude-adaptive quantization.

Par Transformada-Quantização de Baixa Complexidade para Compressão de Imagens 360°

RESUMO

Este trabalho apresenta uma nova abordagem para compressão de imagem aproximada de 360° combinando quantização adaptativa à latitude com matrizes de transformação e quantização de baixa complexidade. O método é projetado especificamente para abordar os desafios exclusivos impostos pelo formato de projeção equirretangular (ERP), que sofre inerentemente de redundâncias, particularmente perto dos polos. A solução proposta reduz significativamente a complexidade computacional eliminando a necessidade de operações de multiplicação e confiando exclusivamente em operações de adição e deslocamento de bits, tornando-a altamente adequada para dispositivos com recursos limitados. Uma análise abrangente de várias transformações de baixa complexidade e matrizes de quantização é apresentada, mostrando a flexibilidade e a natureza plug-and-play da estrutura, que pode ser perfeitamente integrada aos pipelines de compressão existentes. O desempenho dos métodos propostos é avaliado em um conjunto de dados de referência amplamente utilizado — com imagens de 360° em resoluções 4K, 6K e 8K — e comparado com suas contrapartes exatas. Os resultados revelam que a técnica de compressão aproximada proposta reduz substancialmente a demanda computacional, mantendo a qualidade competitiva, com perdas de pico de relação sinal-ruído (PSNR) não excedendo 4 dB em taxas de bits mais baixas. Essas descobertas destacam o potencial do método para permitir o processamento eficiente de mídia de 360° em aplicações onde recursos computacionais e eficiência energética são críticos.

Palavras-chave: compressão de imagens 360°, transformada e quantização de baixa complexidade, quantização adaptativa à latitude.

LIST OF FIGURES

Figure 1.1 Omnidirectional capture (<code>Aerial City</code> from Boyce, Alshina and Abbas (2017)) represented (a) on the sphere surface (visualized using orthographic projection) and (b) on the plane (ERP). A (c) narrow-FoV view depicts one possible perspective image captured from the same scene and camera location.	13
Figure 2.1 Pipeline of JPEG standard.	15
Figure 2.2 Distortion curve of ERP sampling.	18
Figure 2.3 Samples of different representations of 360° images.	19
Figure 3.1 Latitude-adaptive quantization from Simone et al. (2016) for $\phi = \frac{\pi}{4}$	21
Figure 4.1 Pipeline of our proposed compression algorithm for 360° images.	29
Figure 5.1 Average RD curves for planar image compression. Comparison between the JPEG standard (WALLACE, 1992) (C, Q_J) and low-complexity JPEG-like approaches from Oliveira et al. (2017) ($T_1, np2^\circ(Q_J \square Z_1)$), Brahimi et al. (2021) ($T_2, np2^\uparrow(Q_B \square Z_2)$), Araar and Chabbi (2022) ($C, np2^\circ(Q_H)$), and Oliveira et al. (2018) (T_3, Q_J).	32
Figure 5.2 Average RD curves for spherical image compression. Comparison between Simone et al. (2016) (C, Q_ϕ) and the proposed methods (T_3, \tilde{Q}_ϕ and T_3, \tilde{Q}_ϕ^*).	34
Figure 5.3 Compressed <code>Aerial City</code> frame (BOYCE; ALSHINA; ABBAS, 2017) at roughly 0.4 bpp with (b) Simone et al. (2016) (C, Q_ϕ), (c) T_3, \tilde{Q}_ϕ and (d) T_3, \tilde{Q}_ϕ^* . Results are best viewed with digital zoom.	35
Figure 5.4 Average RD curves for spherical image compression. Comparison between Simone et al. (2016) (C, Q_ϕ) and low-complexity variants of the proposed method.	37
Figure 5.5 Compressed <code>Aerial City</code> (BOYCE; ALSHINA; ABBAS, 2017) frame at roughly 0.4 bpp with varying transforms — (a) T_1 (OLIVEIRA et al., 2017) and (b) T_2 (BRAHIMI et al., 2020) —, base quantization matrices — (c) Q_H (ARAAR; CHABBI, 2022) and (d) Q_B (BRAHIMI et al., 2021) —, and power-of-two rounding functions — (e) ($np2^\uparrow(\cdot)$) (BRAHIMI et al., 2021) and (f) ($np2^\downarrow(\cdot)$). Results are best viewed with digital zoom.	39

LIST OF TABLES

Table 3.1	Spherical image compression related work.....	22
Table 3.2	Low-complexity transforms.	23
Table 3.3	Alternative quantization matrices for JPEG.	25
Table 4.1	Permutations to obtain a low-complexity quantization matrix.....	27
Table 5.1	Arithmetic complexity comparison.	35

LIST OF ABBREVIATIONS AND ACRONYMS

bpp	Bits-per-pixels
CMP	Cube-map projection
DCT	Discrete cosine transform
ERP	Equirectangular projection
FoV	Field of view
FPGA	Field Programmable Gate Array
HEVC	High-Efficiency Video Coding
JPEG	Joint Photographic Experts Group
VVC	Versatile Video Coding
GFT	Graph Fourier transform
LUT	Look-up table
PSNR	Peak Signal-to-Noise Ratio
QF	Quantization factor
SSIM	Structural Similarity Index Measure
RD	Rate-distortion
WS-PSNR	Weighted-to-Spherically-Uniform PSNR
WS-SSIM	Weighted-to-Spherically-Uniform SSIM

LIST OF SYMBOLS

\mathbf{C}	DCT transformation matrix
\mathbf{Q}	Quantization matrix
\oslash	Element-wise matrix division
$\lfloor \cdot \rfloor$	Rounding function
$\mathcal{S}(\cdot, \cdot)$	Scaling function for quantization
\odot	Element-wise matrix multiplication
θ	Longitude angle
ϕ	Latitude angle
\mathbf{S}	Diagonal scaling matrix
\mathbf{T}	Low-complexity transformation matrix
$\text{np2}^\circ(\cdot)$	Rounding function to the nearest power-of-two value
$\lceil \cdot \rceil$	Ceiling function
$\text{np2}^\uparrow(\cdot)$	Rounding function to the nearest greater power-of-two value
$\lfloor \cdot \rfloor$	Flooring function
$\text{np2}^\downarrow(\cdot)$	Rounding function to the nearest smaller power-of-two value
$\text{diag}(\cdot)$	Function that extracts the diagonal matrix
$\mathcal{A}(\cdot, \cdot)$	Arrangement function for latitude-adaptive quantization
\mathbf{Q}_ϕ	Latitude-adapted quantization matrix
\square	Short for element-wise matrix multiplication/division

CONTENTS

1 INTRODUCTION.....	12
1.1 Motivation.....	12
1.2 Goals.....	13
1.3 Organization.....	13
2 THEORETICAL FOUNDATION.....	15
2.1 JPEG Image Compression	15
2.2 360° Image Representations.....	17
3 RELATED WORK	20
3.1 360° Image Compression.....	20
3.2 Low-Complexity Image Compression.....	21
4 METHODOLOGY	26
5 RESULTS AND DISCUSSION.....	30
5.1 Preliminary Experiments and Results	30
5.2 Main Experiments and Results.....	31
5.3 Complementary Experiments and Results	36
6 CONCLUSION	40
REFERENCES.....	41

1 INTRODUCTION

1.1 Motivation

Omnidirectional cameras—360°, spherical, or panoramic cameras—capture entire scenes in a single shot (AZEVEDO et al., 2020). These cameras are typically battery-powered devices equipped with lenses capable of covering the whole ($180^\circ \times 360^\circ$) field of view (FoV) (SILVEIRA; JUNG, 2023). Consequently, panoramas are visual signals defined on the surface of a sphere, providing immersive experiences to users in virtual, augmented, and mixed reality applications (SIMONE et al., 2016).

By capturing the surrounding environment, panoramas require significantly higher resolutions compared to conventional images (AZEVEDO et al., 2020). It makes the use of efficient and low-complexity data compression techniques indispensable. However, despite their spherical topology, 360° media are commonly transformed into planar representations through sphere-to-plane mappings (STANDARDISATION, 2019). This mapping process is fundamental, as it enables the application of well-established compression algorithms, such as the JPEG standard (WALLACE, 1992).

The most used sphere-to-plane mappings are the equirectangular projection (ERP), the cubemap projection, and those derived from three-dimensional geometric shapes, such as icosahedron and octahedron (SILVEIRA; JUNG, 2023; HE et al., 2018; EDER et al., 2019; LEE et al., 2022). Each of these mapping functions has its pros and cons, but none are entirely free from distortions. The ERP is considered the *de facto* planar format due to its simplicity but suffers from severe distortions in polar regions (SILVEIRA; JUNG, 2023). Figure 1.1 depicts a spherical image, its ERP-formatted planar version, and a perspective view capture covering part of the scene.

Although applicable, traditional image and video compression methods designed for planar images disregard the underlying geometry of spherical signals. In the case of ERP, pixels are uniformly distributed on the plane, but this distribution does not accurately represent the sphere. Few pixels represent large surface areas near the equator, whereas many pixels correspond to smaller surface areas in polar regions (AZEVEDO et al., 2020). This discrepancy introduces redundancies that can be exploited by adapted compression methods, preserving image quality with minimal degradation (SIMONE et al., 2016).

Among the existing approaches for 360° image compression, many struggle in processing high-resolution ERP images; some of them for using data-dependent data

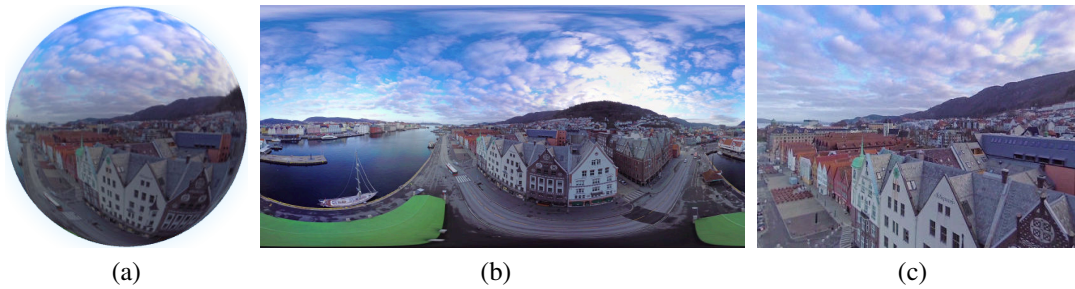


Figure 1.1 – Omnidirectional capture (Aerial City from Boyce, Alshina and Abbas (2017)) represented (a) on the sphere surface (visualized using orthographic projection) and (b) on the plane (ERP). A (c) narrow-FoV view depicts one possible perspective image captured from the same scene and camera location.

decorrelation tools (RIZKALLAH et al., 2018), and others (LI et al., 2022; BIDGOLI et al., 2022) for being extremely hardware intensive. A competing class of methods attains compelling results with tractable processing time and hardware by introducing slight modifications into traditional image and video codecs (SIMONE et al., 2016; JABALLAH; BHAVSAR; LARABI, 2020; XIU; HE; YE, 2018; ZHANG et al., 2019). Thus, balancing computational complexity and image quality remains a key challenge in advancing omnidirectional image compression technologies.

In this work, we explore the spherical compression approach proposed by Simone et al. (2016), adapting it to operate at very low complexity. To our knowledge, this study represents the first effort to apply low-complexity compression strategies to 360° images.

1.2 Goals

The main goal of this work is to propose a low-complexity JPEG-like compression method composed of a transform-quantization pair that considers the distortions intrinsically introduced by the ERP mapping. To this end, the proposed approach combines low-complexity transforms and latitude-adaptive low-complexity quantization, eliminating complex operations throughout the compression flow.

1.3 Organization

This rest of this work is structured as follows. Chapter 2 addresses the fundamental concepts explored in this work, such as the JPEG standard and sphere-to-plane mapping functions. Chapter 3 presents the main works related to our study. More pre-

cisely, we review compression methods tailored to 360° images/videos and JPEG-like low-complexity compression methods for planar imagery. Chapter 4 provides a detailed description of the proposed method, highlighting its implementation and innovative aspects. Chapter 5 comprehensively analyzes the experimental results, demonstrating the method's efficiency and plug-and-play nature. Finally, Chapter 6 summarizes the key points of this work and offers recommendations for future research.

2 THEORETICAL FOUNDATION

This chapter provides the foundational concepts necessary to understand the methods and techniques presented throughout this document. Section 2.1 reviews the basics of the JPEG standard, which serves as a critical background for the compression techniques discussed later. Section 2.2 explores common (multi-)planar representations of 360° imagery, highlighting the challenges associated with these representations and their impact on image processing and compression. By establishing this theoretical groundwork, this chapter sets the stage for the more advanced topics covered in the following chapters.

2.1 JPEG Image Compression

The JPEG standard allows lossy *regular* image compression and controls the perceptual quality of the resulting image (WALLACE, 1992). Hereafter, we briefly discuss the main parts of the JPEG compression pipeline considering grayscale images.

The standard JPEG pipeline is depicted in Figure 2.1, where the processes in blue and green blocks correspond to the encoding and decoding stages of the regular image compression workflow, respectively. Lighter-toned blocks denote the lossless stage of the process, while darker-toned blocks are the lossy stage, which is the central focus of this study. We discuss each of these steps next.

A given image I is firstly divided into 8×8 disjoint blocks $I_{i,j}$, where i and j vary according to the image dimensions. Each block is processed individually. The pixel values from the spatial domain are converted to the spectral domain by applying the direct

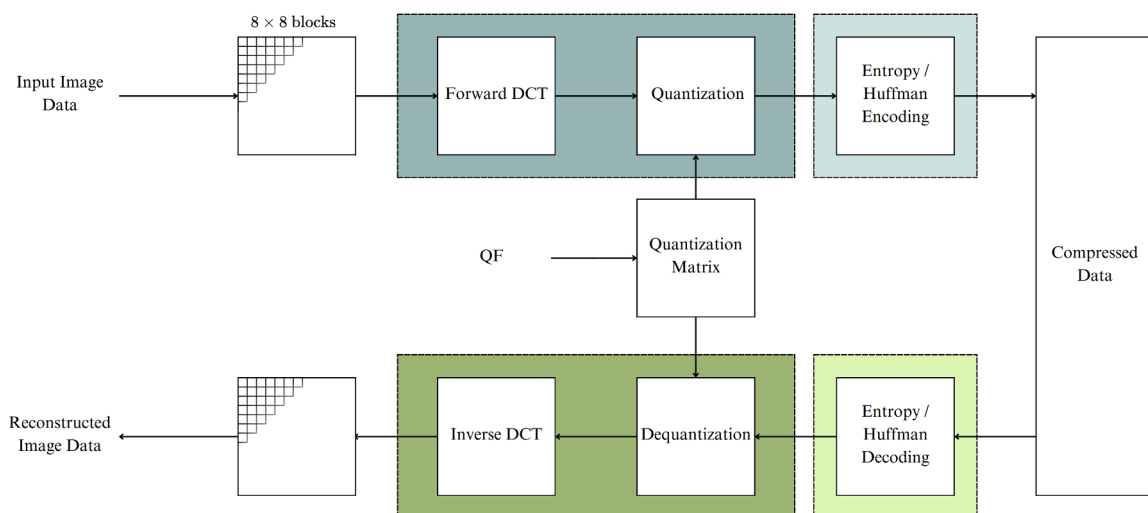


Figure 2.1 – Pipeline of JPEG standard.

2D discrete cosine transform (DCT) (RAO; YIP, 1990) using

$$\mathbf{H}_{i,j} = \mathbf{C} \cdot \mathbf{I}_{i,j} \cdot \mathbf{C}^\top, \quad (2.1)$$

where \mathbf{C} is the 8-point DCT matrix with entries given by Rao and Yip (1990)

$$c_{m,n} = \alpha_m \cos\left(\frac{\pi m(2n+1)}{16}\right), \quad m, n = 0, 1, \dots, 7, \quad (2.2)$$

and $\alpha_0 = 1/\sqrt{8}$ and $\alpha_l = 1/2$ for $l \neq 0$.

After applying the forward DCT to the block data, the transformed blocks $\mathbf{H}_{i,j}$ undergo a quantization process given by

$$\hat{\mathbf{H}}_{i,j} = \lfloor \mathbf{H}_{i,j} \oslash \mathbf{Q} \rfloor, \quad (2.3)$$

where \oslash and $\lfloor \cdot \rfloor$ are the division and rounding to the nearest integer operations applied element-wisely.

The quantization process causes the least relevant coefficients to be discarded. The quantization matrix \mathbf{Q} is parameterized by an user-supplied quality factor term, $QF \in \{0, 1, 2, \dots, 100\}$ (WALLACE, 1992). This QF parameter scales the values of the coefficients in the quantization matrix, where a low QF tends to eliminate more data, resulting smaller file sizes and loss of detail. The standard quantization matrix used in JPEG is given by Wallace (1992)

$$\mathbf{Q} = \mathcal{S}(\mathbf{Q}_J, QF) = \begin{cases} \left\lfloor \frac{(\frac{5000}{QF}) \cdot \mathbf{Q}_J + 50}{100} \right\rfloor, & \text{if } QF < 50 \\ \left\lfloor \frac{(200 - 2QF) \cdot \mathbf{Q}_J + 50}{100} \right\rfloor, & \text{otherwise} \end{cases}, \quad (2.4)$$

where \mathbf{Q}_J is a base quantization matrix given by

$$\mathbf{Q}_J = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}. \quad (2.5)$$

After quantization, the coefficients $\hat{\mathbf{H}}_{i,j}$ of all blocks are losslessly encoded using differential encoding and Huffman encoding (WALLACE, 1992).

The decoding process performs the reverse path, which includes lossless and lossy decoding. The (lossy) dequantization of a block $\hat{\mathbf{H}}_{i,j}$ is obtained by

$$\tilde{\mathbf{H}}_{i,j} = \hat{\mathbf{H}}_{i,j} \odot \mathbf{Q}, \quad (2.6)$$

where \odot denotes the element-wise matrix multiplication operation. The inverse transformation of a dequantized block $\tilde{\mathbf{H}}_{i,j}$ is given by (WALLACE, 1992)

$$\tilde{\mathbf{I}}_{i,j} = \mathbf{C}^\top \cdot \tilde{\mathbf{H}}_{i,j} \cdot \mathbf{C}. \quad (2.7)$$

After rearranging all the blocks $\tilde{\mathbf{I}}_{i,j}$, we obtain an approximation $\tilde{\mathbf{I}}$ to the original image \mathbf{I} .

2.2 360° Image Representations

To enable efficient processing and display of 360° images, it might be convenient to project them onto one or more planes (SHAFI; SHUAI; YOUNUS, 2020; STANDARDISATION, 2019). This approach also allows using compression techniques designed for regular images (BOYCE; ALSHINA; ABBAS, 2017). Despite the variety of sphere-to-plane mapping functions available to obtain planar representations from 360° images, none is distortion- and artifact-free (EDER et al., 2019; AZEVEDO et al., 2020). Among these techniques, the equirectangular projection (ERP) stands out as the *de facto* sphere-to-plane mapping function due to its simplicity (ULLAH; KWON; CHOI, 2021).

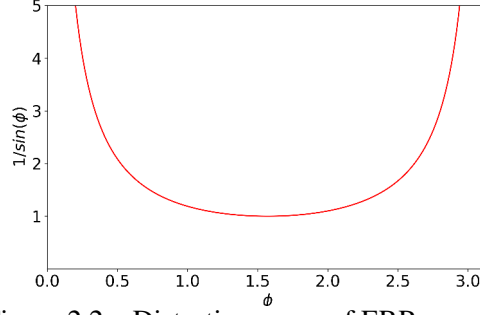


Figure 2.2 – Distortion curve of ERP sampling.

In the ERP mapping, a point on the surface of the unit sphere $\mathbf{x} \in S^2 \subset \mathbb{R}^3$ is determined by two angular parameters, θ and ϕ , which correspond, respectively, to the longitude and latitude of the sphere (TORII; IMIYA; OHNISHI, 2005):

$$\mathbf{x} = [x \ y \ z]^\top = [\cos \theta \sin \phi \ \sin \theta \sin \phi \ \cos \phi]^\top, \quad (2.8)$$

where $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi)$. The point \mathbf{x} is then mapped to a pixel \mathbf{p} in a $w \times h$ discrete planar grid (ERP image) (SILVEIRA; JUNG, 2023):

$$\mathbf{p} = [u \ v]^\top = \left[\left\lfloor \frac{\theta w}{2\pi} \right\rfloor \ \left\lfloor \frac{\phi h}{\pi} \right\rfloor \right]^\top, \quad (2.9)$$

where w and h represent the width and height of the image, respectively (SILVEIRA; JUNG, 2023). Because of the inherent ERP’s non-uniform sampling, objects become increasingly distorted as they approach the polar regions (FERREIRA; SACHT; VELHO, 2017) (recall Figure 1.1). The ERP sampling rate is $\frac{1}{\sin(\phi)}$, which quantifies the deformation as a function of the latitude (SIMONE et al., 2016). This behavior is illustrated in 2.2, where the poles are located at $\phi = 0$ and $\phi = \pi$, with the sampling rate tending to infinity in these regions.

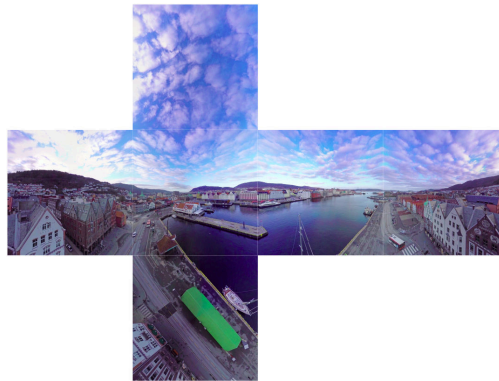
Besides the ERP, alternative sphere-to-plane mapping approaches emerge, each with specific advantages and disadvantages. In the cube-map projection (CMP), the spherical image is projected into six faces, forming a cube circumscribed to the sphere (HE et al., 2018). This method reduces distortions within each face (WANG et al., 2018) but hards the stitching process along the edges where the faces meet (SILVEIRA; JUNG, 2023). Other multi-planar representations based on 3D geometric shapes, such as icosahedrons and octahedrons, can also be explored to mitigate distortion artifacts further, although the faces connectivity issues are aggravated (EDER et al., 2019; LEE et al., 2022). Figure 2.3 illustrates various representations of spherical images applied to a sam-

ple image. The visual distortions introduced by these sphere-to-plane mappings and the connectivity issues between the faces resulting from multi-planar projections – such as those observed in CMP and Icosahedral Projection – are clearly noticeable.

Nevertheless, despite the potential of these alternative methods in general spherical image processing (HE et al., 2018; EDER et al., 2019; LEE et al., 2022), this work relies exclusively on equirectangular projection (ERP) images, owing to their simplicity, widespread adoption, and compatibility with established processing and visualization techniques.



(a) Equirectangular Projection.



(b) Cube-map Projection.



(c) Icosahedral Projection.

Figure 2.3 – Samples of different representations of 360° images.

3 RELATED WORK

This chapter overviews the key works related to the research topic. Section 3.1 briefly reviews existing approaches for 360° media compression. Section 3.2 discusses low-complexity JPEG-like compression based on low-complexity transform and quantization. This chapter provides essential context for the proposed approach, highlighting techniques and methodologies that have influenced the development of this study.

3.1 360° Image Compression

Existing 360° media compression methods usually operate over existing block-based image and video codecs, such as JPEG (WALLACE, 1992), the High-Efficiency Video Coding (HEVC) (POURAZAD et al., 2012), and the Versatile Video Coding (VVC) (ZHAO et al., 2021).

Youvalari, Aminlou and Hannuksela (2017) propose a latitude-adaptive downsampling approach that produces a non-regular shaped image, precluding further image processing. Lee et al. (2017) also resample the image in the spatial domain but reduce the image to a rhombic shape, further rearranged to a square. Both methods serve as pre-processors, and a compression algorithm must be further applied.

In contrast to these pre-processing-based strategies, Simone et al. (2016) proposed an innovative approach that directly adapts the compression process to the ERP structure by leveraging the JPEG compression pipeline. Quantization is performed by selecting, based on a given latitude ϕ , the column indices of the base table that will constitute the ϕ -adjusted quantization matrix. This procedure is illustrated in Figure 3.1 for a latitude of $\phi = \frac{\pi}{4}$. This method adjusts the quantization matrix for each latitude block, allowing for latitude-dependent compression. Blocks closer to the poles, where oversampling is most pronounced, are subjected to more aggressive quantization. This selective quantization reduces the overall bitrate while maintaining the visual fidelity of the reconstructed image in perceptually critical areas. The simplicity and elegance of this method make it highly compatible with existing JPEG infrastructure and readily extendable to video codecs. For instance, similar latitude-adaptive principles can be applied in video encoding by offsetting the quantization parameter (QP) based on block latitude, as explored in studies such as (JABALLAH; BHAVSAR; LARABI, 2020; XIU; HE; YE, 2018; ZHANG et al., 2019).

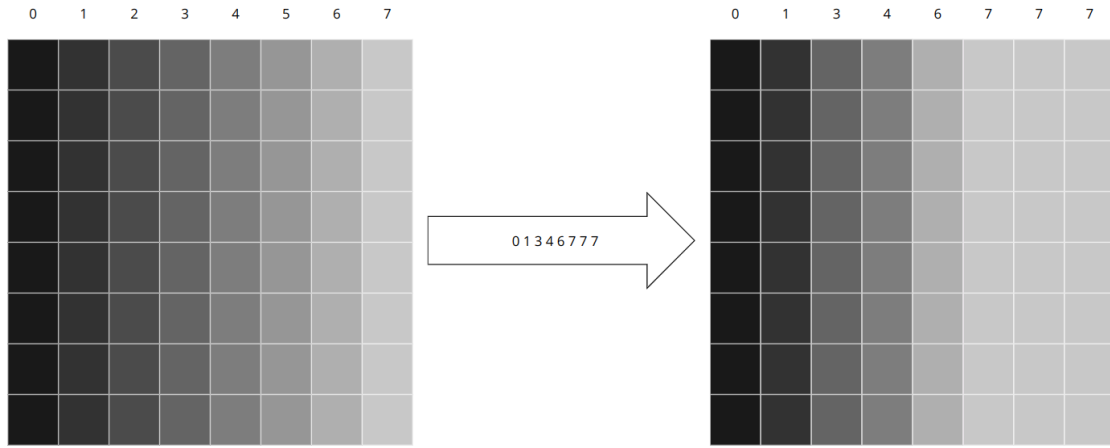


Figure 3.1 – Latitude-adaptive quantization from Simone et al. (2016) for $\phi = \frac{\pi}{4}$.

It is worth mentioning that other approaches exist. Rizkallah et al. (2018) use the data-dependent Graph Fourier Transform (GFT) to data decorrelation and redefine the traditional block-based coding units to irregular regions. Recently, neural image compression methods suitable to panoramas were also introduced (LI et al., 2022; BIDGOLI et al., 2022). Learning-based solutions might, however, still suffer from generalization issues (LIEBERMAN et al., 2023), and both GFT- and learning-based methods are known to be computationally demanding. Table 3.1 summarizes the main works related to the area of media compression 360° .

Given the practical constraints of computational complexity and the need for real-time processing in many applications, more straightforward approaches such as the latitude-adaptive quantization proposed by Simone et al. (2016) become particularly attractive for this study. We will delve into the details of (SIMONE et al., 2016) when coupling it with our low-complexity transform and quantization strategies in Chapter 4.

3.2 Low-Complexity Image Compression

The JPEG standard compresses images by transforming data and then quantizing (WALLACE, 1992). However, the floating-point DCT and integer quantization matrices demand operations that might be difficult to apply in low-power devices (BAYER; CINTRA, 2012; CINTRA; BAYER, 2011; BRAHIMI et al., 2020; OLIVEIRA et al., 2018; OLIVEIRA et al., 2017; BRAHIMI et al., 2021). In contrast, low-complexity compression methods focus on approximate transforms (CINTRA; BAYER, 2011; BRAHIMI et al., 2020; OLIVEIRA et al., 2018) or quantization techniques (OLIVEIRA et al., 2017; BRAHIMI et al., 2021; ARAAR; CHABBI, 2022). These approaches balance compres-

Table 3.1 – Spherical image compression related work

Reference	Methodology
(SIMONE et al., 2016)	Spherical image encoder based on latitude-adaptive quantization matrix.
(YOUVALARI; AMINLOU; HANNUKSELA, 2017)	Spherical image preprocessing for encoding based on latitude-adaptive downsampling.
(LEE et al., 2017)	Spherical image preprocessing for encoding based on latitude-adaptive downsampling.
(RIZKALLAH et al., 2018)	Graph-based 360° image encoder adapted to the spherical surface.
(XIU; HE; YE, 2018)	Spherical video encoder based on latitude-adaptive quantization parameters.
(ZHANG et al., 2019)	Spherical video encoder based on latitude-adaptive quantization parameters.
(JABALLAH; BHAVSAR; LARABI, 2020)	Spherical video encoder based on latitude-adaptive quantization parameters.
(LI et al., 2022)	Spherical image encoder based on Convolutional Neural Networks.
(BIDGOLI; MAUGEY; ROUMY, 2019)	Spherical image encoder based on Convolutional Neural Networks.

sion efficiency and (arithmetic) complexity at acceptable performance drops.

Unlike the DCT, which relies on floating-point multiplications, approximate transforms use simpler operations, such as additions and bit-shifts, improving computational efficiency with minimal loss in the decorrelation performance (SILVEIRA et al., 2022). Most DCT approximations, hereafter represented by $\tilde{\mathbf{C}}$, can be split into two components: an adjustment real-valued diagonal matrix \mathbf{S} and a low-complexity matrix \mathbf{T} :

$$\tilde{\mathbf{C}} = \mathbf{S} \cdot \mathbf{T}, \quad (3.1)$$

where

$$\mathbf{S} = \sqrt{(\mathbf{T} \cdot \mathbf{T}^T)^{-1}}. \quad (3.2)$$

The low-complexity matrix \mathbf{T} is composed of coefficients that reduce computational power requirements when implemented in hardware (SILVEIRA et al., 2022). These low-complexity matrices typically have coefficients in $\{0, \pm 1, \pm \frac{1}{2}, \pm 2\}$ (SILVEIRA et al., 2022). Null entries lead to the absence of operations, entries in $\{\pm 1\}$ are implemented as signed additions, and entries in $\{\pm \frac{1}{2}, \pm 2\}$ are simple bit-shift operations. In contrast, the adjustment matrix \mathbf{S} compensates for the simplifications introduced by the low-complexity matrix, bringing the resulting matrix $\tilde{\mathbf{C}}$ closer to orthonormality (CINTRA; BAYER, 2011). The matrix \mathbf{S} can be embedded in the quantization matrix to reduce the multiplication cost of the transform-quantization pair (BAYER; CINTRA, 2012;

Table 3.2 – Low-complexity transforms.

Reference	Transformation Matrix
(CINTRA; BAYER, 2011)	$\mathbf{T}_1 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & -1 & -1 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & -1 & 1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & -1 & 1 & -1 & 1 & 0 \end{pmatrix}$
(BRAHIMI et al., 2020)	$\mathbf{T}_2 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{pmatrix}$
(OLIVEIRA et al., 2018)	$\mathbf{T}_3 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 1 & 0 & 0 & -1 & -2 & -2 \\ 2 & 1 & -1 & -2 & -2 & -1 & 1 & 2 \\ 1 & 0 & -2 & -2 & 2 & 2 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 2 & -2 & 0 & 1 & -1 & 0 & 2 & -2 \\ 1 & -2 & 2 & -1 & -1 & 2 & -2 & 1 \\ 0 & -1 & 2 & -2 & 2 & -2 & 1 & 0 \end{pmatrix}$

CINTRA; BAYER, 2011; BRAHIMI et al., 2020; OLIVEIRA et al., 2018; OLIVEIRA et al., 2017; BRAHIMI et al., 2021).

Three DCT approximations — $\tilde{\mathbf{C}}_1$ (CINTRA; BAYER, 2011), $\tilde{\mathbf{C}}_2$ (BRAHIMI et al., 2020), and $\tilde{\mathbf{C}}_3$ (OLIVEIRA et al., 2018) — that follow the formalism above are considered in this work. The respective low-complexity matrices are shown in Table 3.2. Their fast algorithms allow 1D data transformation with reduced cost. Precisely, \mathbf{T}_1 can be implemented at the expense of 22 additions (CINTRA; BAYER, 2011), whereas \mathbf{T}_2 demands 16 additions (BRAHIMI et al., 2020), and \mathbf{T}_3 requires 24 additions and 6 bit-shifts only (OLIVEIRA et al., 2018).

The quantization/dequantization step involves divisions/multiplications and rounding, which might be expensive for the hardware. An efficient simplification can be achieved by adjusting the coefficients of the quantization matrix to powers of two and

replacing divisions/multiplications with bit shifts (OLIVEIRA et al., 2017; ARAAR; CHABBI, 2022; BRAHIMI et al., 2021).

Oliveira et al. (2017) propose to approximate the entries of the standard JPEG quantization matrix \mathbf{Q}_J to their closest powers of two using

$$\text{np}2^\circ(x) = 2^{\lfloor \log_2(x) \rfloor}. \quad (3.3)$$

Araar and Chabbi (2022) also use Equation (3.3) to obtain a low-complexity quantization matrix changing the base matrix to another that better captures the characteristics of the human visual system \mathbf{Q}_H , illustrated in Table 3.3.

Similarly, Brahimi et al. (2021) presented a rounding function that raises the coefficients of a matrix to the next power of two using

$$\text{np}2^\uparrow(x) = 2^{\lceil \log_2(x) \rceil}. \quad (3.4)$$

Their method is applied to another base quantization matrix \mathbf{Q}_B , also illustrated in Table 3.3.

Equations (3.3) and (3.4) are the main tools for devising low-complexity quantization matrices from any base quantization matrix. Although not explicitly mentioned in the literature, the function

$$\text{np}2^\downarrow(x) = 2^{\lfloor \log_2(x) \rfloor} \quad (3.5)$$

could also be considered as an alternative. The adoption of Equations (3.3), (3.4), and (3.5) will be further discussed in Section 5.3.

As said before, the adjustment matrix \mathbf{S} associated with a low-complexity matrix \mathbf{T} can be incorporated into the (de)quantization step of a JPEG-like codec (OLIVEIRA et al., 2017). From a given a base quantization matrix \mathbf{Q} (e.g., \mathbf{Q}_J , \mathbf{Q}_B , or \mathbf{Q}_H), we obtain two distinct quantization matrices (forward and backward quantization matrices)

$$\mathbf{Q}^f = \mathbf{Q} \oslash \mathbf{Z} \quad (3.6)$$

and

$$\mathbf{Q}^b = \mathbf{Q} \odot \mathbf{Z}, \quad (3.7)$$

Table 3.3 – Alternative quantization matrices for JPEG.

Reference	Quantization Matrix
(ARAAR; CHABBI, 2022)	$\mathbf{Q}_H = \begin{pmatrix} 16 & 16 & 16 & 16 & 17 & 18 & 21 & 24 \\ 16 & 16 & 16 & 16 & 17 & 19 & 22 & 25 \\ 16 & 16 & 17 & 18 & 20 & 22 & 25 & 29 \\ 16 & 16 & 18 & 21 & 24 & 27 & 31 & 36 \\ 17 & 17 & 20 & 24 & 30 & 35 & 41 & 47 \\ 18 & 19 & 22 & 27 & 35 & 44 & 54 & 65 \\ 21 & 22 & 25 & 31 & 41 & 54 & 70 & 88 \\ 24 & 25 & 29 & 36 & 47 & 65 & 88 & 115 \end{pmatrix}$
(BRAHIMI et al., 2021)	$\mathbf{Q}_B = \begin{pmatrix} 20 & 17 & 18 & 19 & 22 & 36 & 36 & 31 \\ 19 & 17 & 20 & 22 & 24 & 40 & 23 & 40 \\ 20 & 22 & 24 & 28 & 37 & 53 & 50 & 54 \\ 22 & 20 & 25 & 35 & 45 & 73 & 73 & 58 \\ 22 & 21 & 37 & 74 & 70 & 92 & 101 & 103 \\ 24 & 43 & 50 & 64 & 100 & 104 & 120 & 92 \\ 45 & 100 & 62 & 79 & 100 & 70 & 70 & 101 \\ 41 & 41 & 74 & 59 & 70 & 90 & 100 & 99 \end{pmatrix}$

where $\mathbf{Z} = \text{diag}(\mathbf{S}) \cdot \text{diag}(\mathbf{S})^\top$. Equations (3.6) and (3.7) replace \mathbf{Q} from Equations (2.3) and (2.6), respectively. To perform the transform and (de)quantization with no multiplications, Equations (3.3), (3.4), or (3.5) are applied to the entries of the forward and backward quantization matrices from Equations (3.6) and (3.7).

Oliveira et al. (2017) show that by combining a low-complexity transform, such as \mathbf{T}_1 (CINTRA; BAYER, 2011), with low-complexity approximations for the forward and backward quantization matrices (Equations (3.6) and (3.7) and) using Equation (3.3) allows a JPEG-like low-complexity encoder/decoder. We explore this same intuition but in the context of 360° image compression by coupling the latitude-adaptive quantization formulation from Simone et al. (2016).

4 METHODOLOGY

Our literature review did not reveal any records of previous studies focused on low-complexity 360° image compression. To our knowledge, this work pioneers the area. The central concept is to explore the intrinsic characteristics of ERP mapping, incorporating adaptive quantization, and progressively increasing the compression level in image regions with higher redundancy (poles). In addition, our work combines low-complexity transformation and quantization strategies, aiming at efficiency. The objective is to minimize computational costs of the transformation and quantization steps in a JPEG-like codec while maintaining the efficiency and visual quality of the resulting image.

To address the distortions induced by the ERP mapping, adaptive latitude-based quantization techniques can be employed. Here, we base on the methodology from Simone et al. (2016), which, unlike others (JABALLAH; BHAVSAR; LARABI, 2020; XIU; HE; YE, 2018; ZHANG et al., 2019), applies to still images. Specifically, a latitude-adapted quantization matrix \mathbf{Q}_ϕ is constructed by rearranging the columns of a base quantization matrix \mathbf{Q} depending on the block latitude $\phi \in [0, \pi)$. Note that the ϕ interval discretization depends on the ERP image size (SIMONE et al., 2016). The latitude-adapted quantization matrix can be obtained through¹

$$\mathbf{Q}_\phi = \mathcal{A}(\mathbf{Q}, \phi) = [\mathbf{q}_k], \quad k = \min(7, \max(\lfloor k' / \sin(\phi) \rfloor, 0)), \quad (4.1)$$

where $k' = 0, 1, 2, \dots, 7$, $\mathcal{A}(\cdot, \cdot)$ is a function that returns a ϕ -adapted quantization matrix \mathbf{Q}_ϕ , and \mathbf{q}_k represents the k -th column vector from \mathbf{Q} (recall the intuition from Figure 3.1).

This study proposes to adapt the method outlined by Simone et al. (2016). This adaptation introduces specialized matrices to minimize the computational cost of the encoding and decoding processes. By employing this approach, distinct transformation and quantization matrices can be utilized. We describe the plug-and-play nature of our approach in Chapter 5.

To construct a low-complexity quantization matrix from a base quantization matrix and the scaling component of a low-complexity transform, three straightforward operations must be applied. Following the complexity reduction strategy detailed in Sec-

¹The formulation adopted here slightly differs from (SIMONE et al., 2016). Simone et al. (2016) claims using the ceiling operator and no clipping, unlike Equation (4.1) although their results seem to corroborate our formulation and not theirs. Also, because the ϕ range from their paper is $[-\pi/2, \pi/2)$, they adopt $\cos(\cdot)$ instead of $\sin(\cdot)$.

Table 4.1 – Permutations to obtain a low-complexity quantization matrix.

Permutation	First Op.	Second Op.	Third Op.	Formulation
P1	(1)	(2)	(3)	$\mathcal{A}(\text{np2}(\mathbf{Q} \square \mathbf{Z}), \phi)$
P2	(1)	(3)	(2)	$\text{np2}(\mathcal{A}((\mathbf{Q} \square \mathbf{Z}), \phi))$
P3	(2)	(1)	(3)	$\mathcal{A}(\text{np2}(\mathbf{Q}) \square \mathbf{Z}, \phi)$
P4	(2)	(3)	(1)	$\mathcal{A}(\text{np2}(\mathbf{Q}), \phi) \square \mathbf{Z}$
P5	(3)	(1)	(2)	$\text{np2}(\mathcal{A}(\mathbf{Q}, \phi) \square \mathbf{Z})$
P6	(3)	(2)	(1)	$\text{np2}(\mathcal{A}(\mathbf{Q}, \phi)) \square \mathbf{Z}$

tion 3.2 and considering the geometric properties introduced by the ERP mapping, these operations include quantization matrix coefficients scaling (Equation (3.6) and Equation (3.7)), power-of-two coefficient approximation (Equation (3.3), Equation (3.4) or Equation (3.5)), and matrix column rearranging based on ϕ (Equation (4.1)). The order of these operations is vital, as it can produce varying results. From the six possible operation permutations, only one leverages advantages from a hardware implementation perspective.

Table 4.1 demonstrates the possible operation permutations, where \square is a short for either Equation (3.6) or Equation (3.7) and $\text{np2}(\cdot)$ stands for either Equation (3.3), Equation (3.4), or Equation (3.5). Note that selecting the QF parameter, embedded into \mathbf{Q} , is the first operation taken.

Considering the operations involved — (1) scaling of the quantization matrix, (2) rounding coefficients to powers of two, and (3) column rearranging based on ϕ — it becomes evident that certain permutations should be excluded from the analysis process. Operation (2) must be applied only after the adjustment; if it is applied beforehand, then the transformed coefficients revert to the Integral/Real domain, which results in the elimination of Permutations P3, P4, and P6. From a mathematical perspective, Operation (1) should be performed after Operation (3), as the resulting quantization matrix should be adjusted based on the coefficients of the selected transform. This requirement makes Permutation P5 important for the analysis.

However, Permutations P1 and P2, which are expected to produce identical results due to the commutative property between Operations (2) and (3), are of special interest. Transforming a matrix into powers of two and then replicating its columns is simpler in terms of hardware implementation than replicating the columns first and subsequently transforming all the resulting matrices into powers of two, making Permutation P1 preferable than P2. Column arrangement can be performed via multiplexation. We note that Permutations P1/P2 lead to compelling results compared to Permutation P5. We call here $\tilde{\mathbf{Q}}_\phi = \mathcal{A}(\text{np2}(\mathbf{Q} \square \mathbf{Z}), \phi)$ as a short for $\tilde{\mathbf{Q}}_\phi^f = \mathcal{A}(\text{np2}(\mathbf{Q} \odot \mathbf{Z}), \phi)$ and

$$\tilde{\mathbf{Q}}_\phi^b = \mathcal{A}(\text{np2}(\mathbf{Q} \odot \mathbf{Z}), \phi).$$

Following the introduction of the new operations and the analysis of their application sequence, a novel pipeline model has been developed for compressing 360° images with low computational cost. This pipeline is illustrated in Figure 4.1, where the red, yellow, and gray blocks represent Operations (1), (2), and (3), respectively. It is crucial to note that both the forward and backward transformations are DCT approximations, as they employ low-complexity matrices, similar to the quantization and dequantization steps. Furthermore, a new input parameter, ϕ , has been defined to differentiate the quantization process across varying latitudinal levels of the image. The look-up table (LUT) output can be efficiently implemented in hardware using multiplexers. These multiplexers select columns from the power-of-two forward and backward quantization matrices.

Equations (4.2) and (4.3) exemplify the forward and backward quantization matrices obtained using $\mathcal{A}(\text{np2}^\circ(\mathcal{S}(\mathbf{Q}_J, 75) \oslash \mathbf{Z}_3), \frac{\pi}{8})$ and $\mathcal{A}(\text{np2}^\circ(\mathcal{S}(\mathbf{Q}_J, 75) \odot \mathbf{Z}_3), \frac{\pi}{8})$ for a 4K (3840×1920) ERP image:

$$\tilde{\mathbf{Q}}_{f, \frac{\pi}{8}} = \begin{pmatrix} 64 & 64 & 64 & 128 & 128 & 256 & 256 & 512 \\ 64 & 128 & 128 & 128 & 128 & 512 & 512 & 512 \\ 64 & 128 & 128 & 256 & 256 & 512 & 512 & 512 \\ 64 & 128 & 256 & 256 & 256 & 1024 & 1024 & 512 \\ 64 & 128 & 256 & 256 & 256 & 512 & 512 & 512 \\ 128 & 256 & 512 & 512 & 512 & 1024 & 1024 & 1024 \\ 256 & 512 & 1024 & 1024 & 512 & 1024 & 1024 & 1024 \\ 512 & 1024 & 1024 & 1024 & 512 & 1024 & 1024 & 1024 \end{pmatrix}, \quad (4.2)$$

$$\tilde{\mathbf{Q}}_{b, \frac{\pi}{8}} = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 2 & 2 & 2 & 2 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{2} & \frac{1}{2} & 1 & 2 & 2 & 2 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 2 & 2 & 2 & 2 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 1 & 2 & 2 & 2 & 2 \\ 1 & 1 & 2 & 2 & 4 & 4 & 4 & 4 \\ 1 & 1 & 2 & 2 & 4 & 4 & 4 & 2 \\ 2 & 2 & 2 & 2 & 4 & 4 & 4 & 2 \\ 4 & 2 & 2 & 2 & 4 & 2 & 2 & 2 \end{pmatrix}, \quad (4.3)$$

where \mathbf{Z}_3 denotes the adjustment matrix derived from \mathbf{T}_3 (OLIVEIRA et al., 2018).

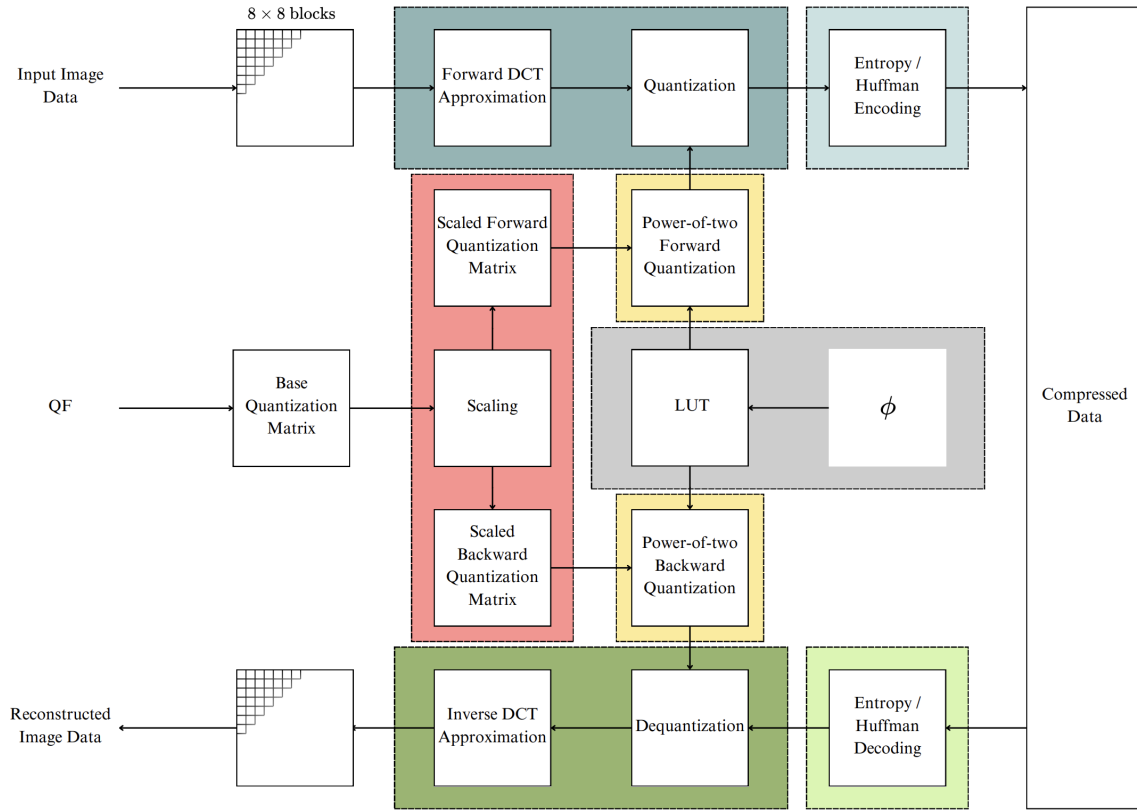


Figure 4.1 – Pipeline of our proposed compression algorithm for 360° images.

It is worth mentioning that once the QF parameter is set, the proposed approach can be implemented using only additions and bit-shifts as the methods for *planar* image compression discussed in Section 3.2. However, this approach adjusts the quantization matrix for each block latitude as Simone et al. (2016). Equation (4.1) can be precomputed and the associated circuitry implemented using a preset block-latitude LUT and simple multiplexers for a fixed ERP image size, avoiding transcendental computation.

5 RESULTS AND DISCUSSION

Before assessing the proposed approach, we performed a preliminary exploration of relevant low-complexity compression methods devised for planar imagery (OLIVEIRA et al., 2017; OLIVEIRA et al., 2018; BRAHIMI et al., 2021; ARAAR; CHABBI, 2022). This analysis is discussed in Section 5.1. Then, in Section 5.2, we test our methodology, approximating the transformation and quantization steps from Simone et al. (2016). Then, in Section 5.3, we vary the key components of the proposed method such as the transform, base quantization matrix and power-of-two rounding function. All approaches were implemented by the author.

5.1 Preliminary Experiments and Results

A preliminary analysis of established methods for low-complexity planar image compression is a fundamental step before implementing our proposal. This review enables a thorough evaluation of the key aspects involved in each stage of the compression process, providing a deeper understanding of the strengths and limitations of existing approaches. Also, this analysis is crucial for identifying gaps that can be addressed and outlining improvements and innovations that contribute to advancing the state of the art.

All experiments conducted in this work considered uncompressed grayscale images¹. The USC-SIPI (USC-SIPI, 1997) dataset was used to experiment with planar imagery. The `ruler` image was excluded from the tests because it is a dual-tone image, and we considered the remaining 44 512×512 images for evaluation. Here, we adopt the Peak Signal-to-Noise Ratio (PSNR) (HUYNH-THU; GHANBARI, 2008) and the Structural Similarity (SSIM) (WANG et al., 2004) for assessing the quality of the decompressed planar images. The average number of bits per pixel (bpp) was used to measure the data rate associated with different compression levels. We set the quality factor parameter as $QF \in \{5, 10, 15, \dots, 95\}$, where lower QF values correspond to higher compression. The results are averaged across images for different resolutions and QF values and presented in rate-distortion (RD) curves. For simplicity, we omit the notation related to the QF parameter selection (Equation (2.4)) in the graphs.

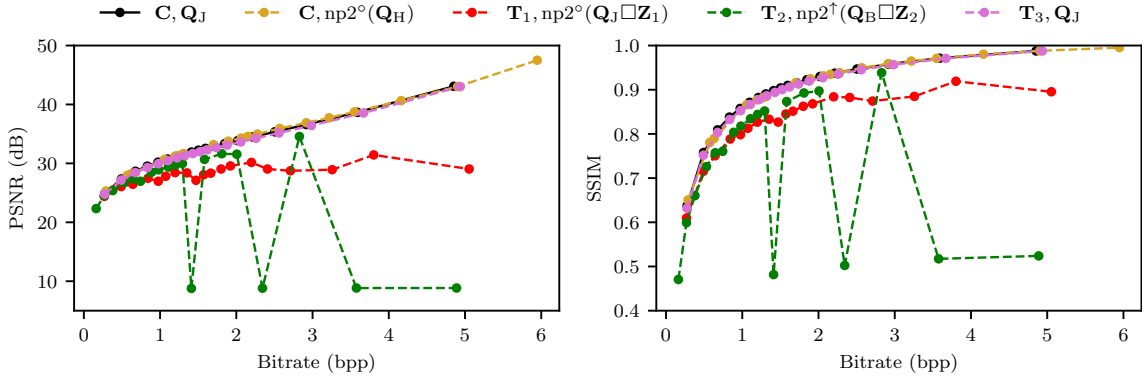
¹We selected grayscale images for tests because the quantization matrices explored in this work were devised for the luma component rather than the chroma component. Though, extending our method to work with color images is straightforward.

Figure 5.1 presents the RD curves corresponding to the JPEG standard (WALLACE, 1992) and the low-complexity JPEG-like compression methods (OLIVEIRA et al., 2017; OLIVEIRA et al., 2018; BRAHIMI et al., 2021; ARAAR; CHABBI, 2022) described in Section 3.2. The RD curves illustrate the performance of each method, offering an objective assessment of compression efficiency and image quality preservation. This comparative analysis allows us to identify the inherent trade-offs between computational complexity and reconstructive quality; insights that guided strategic decisions in the design and implementation of our solution.

The RD curves from Figure 5.1 show that the method proposed by Oliveira et al. (2018) exhibits a smooth curve, closely resembling the behavior of the traditional JPEG standard (WALLACE, 1992). Similarly, the approach proposed by Araar and Chabbi (2022) also exhibits a smooth curve, albeit with a slight shift towards higher bitrates. Although Oliveira et al. (2018) attained outstanding results, the significant performance gains in contrast to some other methods compared are partially attributed to the fact that they only approximate the JPEG’s transform stage. In contrast, Araar and Chabbi (2022) approximate the JPEG’s quantization phase only. The methods proposed by Oliveira et al. (2017) and Brahimi et al. (2021) employ both low-complexity transforms and quantization matrices. Thus, their RD curves display some peaks and valleys. Notably, the method proposed by Brahimi et al. (2021) demonstrates pronounced valleys when $QF \in \{60, 80, 90, 95\}$. We found that the rounding strategy applied to the quantization matrix from Brahimi et al. (2021) leads to overflow, resulting in coefficients out of range of representation values. It is worth mentioning that Brahimi et al. (2021) do not make explicit how to scale \mathbf{Q}_B to different QF values before power-of-two rounding. In this work, we considered applying $\text{np}2^\uparrow(\mathcal{S}(\mathbf{Q}_B, QF) \square \mathbf{Z}_2)$. The same applies to the method from Araar and Chabbi (2022), where we adopted $\text{np}2^\circ(\mathcal{S}(\mathbf{Q}_H, QF))$.

5.2 Main Experiments and Results

Here, the results obtained by the method from Simone et al. (2016) are presented and discussed, along with comparisons of its approximation proposed here. The approximated method follows the order of operations established as ideal from a hardware implementation perspective (P1 in Table 3.2) as claimed in Chapter 4. Simone et al. (2016) consistently outperforms JPEG in compression tests. For this reason, the low-complexity approaches are exclusively compared to this reference (exact) method.



(a) Average PSNR and SSIM for planar images

Figure 5.1 – Average RD curves for planar image compression. Comparison between the JPEG standard (WALLACE, 1992) (C, Q_J) and low-complexity JPEG-like approaches from Oliveira et al. (2017) ($T_1, np2^\circ(Q_J \square Z_1)$), Brahimi et al. (2021) ($T_2, np2^\circ(Q_B \square Z_2)$), Araar and Chabbi (2022) ($C, np2^\circ(Q_H)$), and Oliveira et al. (2018) (T_3, Q_J).

For assessing 360° methods, we selected video frames from the dataset associated with the Common Test Conditions for 360° Video Compression (CTC360) (BOYCE; ALSHINA; ABBAS, 2017). To our knowledge, no other datasets that contain real ERP images free of compression artifacts are available. This limitation was a key factor in our choice of dataset, as ensuring accuracy and reliability is essential for our analysis. We extracted 30 frames (the first, the middle, and the last frame) from the ten 8-bit, 30-frames-per-second ERP video sequences of different resolutions. Precisely, we used twelve frames at 4K (3840×1920), six frames at 6K (6144×3072), and twelve frames at 8K (8192×4096). From now on, we will consider these frames simply as images.

For the 360° imagery case, we considered the Weighted-to-Spherically-Uniform PSNR (WS-PSNR) (SUN; LU; YU, 2017) and Weighted-to-Spherically-Uniform SSIM (WS-SSIM) (ZHOU et al., 2018) as image quality metrics. Here, we also consider bpp to measure bitrate and set the QF parameter as in the experiments from Section 5.1.

The proposed method uses the transform T_3 (OLIVEIRA et al., 2018) from Table 3.2 and Q_J (WALLACE, 1992) (Equation (2.5)) as a base quantization matrix. Additionally, it employs the function $np2^\circ$ (Equation (3.3)) (OLIVEIRA et al., 2017) to derive power-of-two values for the quantization process. The adopted parameter configuration does not guarantee the best results, but the best compression characteristics were selected to support our proposal. Given the number of available operations — i.e., transforms, quantization matrices, and the power-of-two functions — it would be necessary to evaluate $3! \times 3! \times 3! = 216$ combinations only for the parameters presented in this study (many other low-complexity transforms and base quantization matrices exist). Therefore, performing an exhaustive analysis of all possibilities would be computationally unfeasible.

in the context of this work. Thus, we chose to explore strategic configurations that maximized the balance between quality and computational cost, highlighting the potential of our approach in representative scenarios.

From now on, we will call our proposed method as the transform-quantization pair given by $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi = \mathcal{A}(\text{np}2^\circ(\mathbf{Q}_J \square \mathbf{Z}_3), \phi)$ (Permutation P1 from Table 4.1). Recall, however, that Permutation P5 is the one correct from a mathematical point of view and we use the notation $\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi*} = \text{np}2^\circ(\mathcal{A}(\mathbf{Q}_J, \phi) \square \mathbf{Z}_3)$ for that. Above and in the graphs next, for simplicity, we omit the notation related to the QF parameter selection (Equation (2.4)). Figure 5.2 depicts the average results when approximating the method from Simone et al. (2016) by $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi$ and $\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi*}$.

The results demonstrate the feasibility of integrating low-complexity techniques into the compression process while accounting for the geometric characteristics of ERP mapping. It is important to note that the bitrate range decreases and image quality increases when considering a set of higher-resolution ERP images. Larger images tend to exhibit greater information redundancy, allowing compression algorithms to represent these areas using fewer bits and more efficiently. It is important to note that the oscillations observed in the curves are caused by the rounding function to the quantization matrix, a feature inherited from the low-complexity methods of the planar counterpart. The two formulations analyzed yielded nearly identical results.

In terms of WS-PSNR, $\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi*}$ differs from $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi$ by not more than 1 dB. Additionally, when compared with Simone et al. (2016), $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi$ has a quality loss of no more than 4 dB at 0.5 bpp or less. For bitrates (bpp) above 0.5, the curve behavior is similar to that of methods employing both low-complexity transformation and low-complexity quantization for planar image compression. This can be interpreted as a residual effect of utilizing both techniques together. The results demonstrate that low-cost codec implementations can effectively preserve visual quality, mainly as measured by the WS-SSIM metric, which correlates more closely with human perception compared to WS-PSNR (ZHOU et al., 2018). These findings underscore the effectiveness of the proposed solutions in 360° image compression scenarios, at least for low bitrates.

BD-Rate calculation relies on curve interpolation to estimate the area between rate-distortion curves. However, interpolation can smooth out or distort critical features of the data, such as local valleys or sharp transitions, which often indicate meaningful performance differences. As a result, this process might obscure the true behavior of the codecs under specific conditions, potentially leading to less reliable or misleading

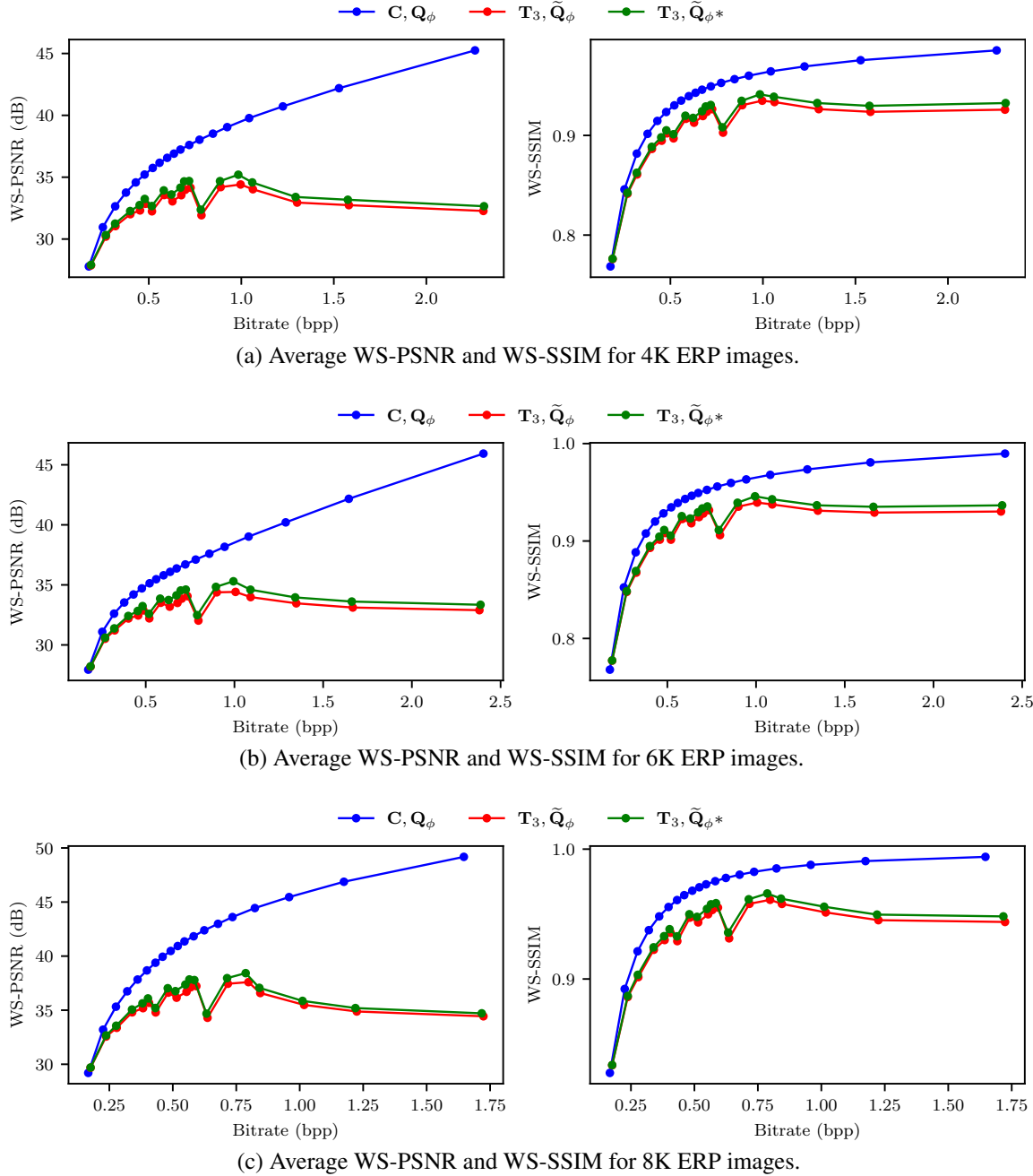


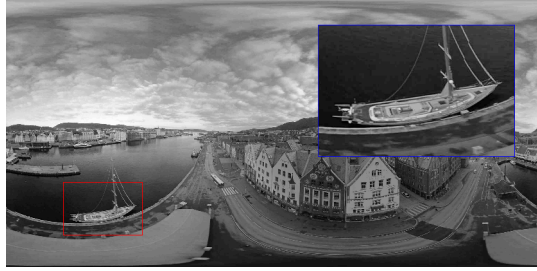
Figure 5.2 – Average RD curves for spherical image compression. Comparison between Simone et al. (2016) (C, Q_ϕ) and the proposed methods (T_3, \tilde{Q}_ϕ and T_3, \tilde{Q}_ϕ^*).

Table 5.1 – Arithmetic complexity comparison.

Approx. block	Mult.	Add.	Bit-shift	Rounding
\mathbf{C}, \mathbf{Q}_J (WALLACE, 1992)	288	928	0	64
$\mathbf{C}, \mathbf{Q}_\phi$ (SIMONE et al., 2016)	288	928	0	64
$\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi = \mathcal{A}(\text{np}2^\circ(\mathbf{Q}_J \square \mathbf{Z}_3), \phi)$	0	768	320	0
$\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi^*} = \text{np}2^\circ(\mathcal{A}(\mathbf{Q}_J, \phi) \square \mathbf{Z}_3)$	0	768	320	0



(a) Original



(b) WS-PSNR: 35.48 dB and WS-SSIM: 0.91 at 0.38 bpp



(c) WS-PSNR: 33.65 dB and WS-SSIM: 0.89 at 0.40 bpp



(d) WS-PSNR: 35.09 dB and WS-SSIM: 0.89 at 0.40 bpp

Figure 5.3 – Compressed Aerial City frame (BOYCE; ALSHINA; ABBAS, 2017) at roughly 0.4 bpp with (b) Simone et al. (2016) ($\mathbf{C}, \mathbf{Q}_\phi$), (c) $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi$ and (d) $\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi^*}$. Results are best viewed with digital zoom.

interpretations. For these reasons, we avoided the BD-Rate calculation in this analysis.

Table 5.1 details the number of operations required to perform per-block compression (transform and quantization) for each approach considered. Our approaches are multiplication-free and show much lower complexity compared to traditional methods. While $\mathbf{T}_3, \tilde{\mathbf{Q}}_{\phi^*}$ achieves the best image quality results, it would require a massive LUT to be implemented with the claimed arithmetic complexity (each ERP block latitude needs to store a quantization matrix). Conversely, despite yielding slightly poorer results, $\mathbf{T}_3, \tilde{\mathbf{Q}}_\phi$ is the most cost-effective for implementation (a single mapping relating the ERP block latitude to eight column indices is stored in the LUT). Thus, selecting an approach that effectively balances compression efficiency with operational simplicity is essential.

As shown in Figure 5.3, the proposed approach exhibits quality losses not exceeding 4 dB in low-bitrate scenarios. Furthermore, the visual quality degradation is minimal, if not imperceptible, as illustrated in Figure 5.3 with the help of a highlighted region of

interest (RoI). The use of our method might be recommended for devices requiring low power consumption and not demanding super high-fidelity representation.

5.3 Complementary Experiments and Results

To demonstrate the plug-and-play nature of our methodology, we adjust its parameters (transform and quantization matrices and power-of-two rounding function) and assess the results using the subset of 4K ERP images from the CTC360 dataset (BOYCE; ALSHINA; ABBAS, 2017) (noting that the results for 6K and 8K ERP images are expected to be similar as in Section 5.2). The QF parameter range and the metrics used are the same as in Section 5.2. To simplify the graphs analyses, as before, the notation associated with the QF parameter selection (Equation (2.4)) is omitted.

Figure 5.4a shows the results when considering transforms T_1 (OLIVEIRA et al., 2017) and T_2 (BRAHIMI et al., 2020) (and corresponding adjusting matrices Z_1 and Z_2), instead of T_3 (OLIVEIRA et al., 2018), while fixing the other parameters of our method.

Note that all curves fluctuate between the range of 25 and 35 dB for the WS-PSNR metric, with the curve that uses T_3 above the others in both metrics analyzed for most bpp values. Compared to Simone et al. (2016), $T_1, \mathcal{A}(\text{np}2^\circ(\tilde{Q}_J \square Z_1), \phi)$ has a quality loss of at most 4 dB, while $T_2, \mathcal{A}(\text{np}2^\circ(\tilde{Q}_J \square Z_2), \phi)$ has a quality drop no greater than 3 dB and $T_3, \mathcal{A}(\text{np}2^\circ(\tilde{Q}_J \square Z_3), \phi)$ deviates by only 2.5 dB, all for bpp values smaller than 0.5.

Qualitative results when considering transforms T_1 and T_2 are shown in Figures 5.5a and 5.5b. Their results differ by 2.23 dB and 1.09 dB, respectively, when compared with our main proposal with T_3 . Considering other transforms than T_3 , alters the arithmetic complexity shown in Table 5.1. When considering T_1 in our method, the arithmetic complexity per block becomes 704 additions and 128 bit-shifts; while T_2 modifies the per-block cost to 512 additions and 128 bit-shifts only.

Figure 5.4b presents a quantitative analysis for our method when switching the base quantization matrix Q_J (WALLACE, 1992) by Q_B (BRAHIMI et al., 2021) and Q_H (ARAAR; CHABBI, 2022) and keeping the other parameters unaltered. Modifying the quantization matrices in our proposal yields favorable behavior. The base quantization matrix Q_H exhibits a more elongated bitrate range. This configuration demonstrates an advantage for lighter compression scenarios, whereas the base quantization matrices Q_J and Q_B achieve superior performance under higher compression levels.

Qualitative results for the proposed methodology when considering Q_H and Q_B

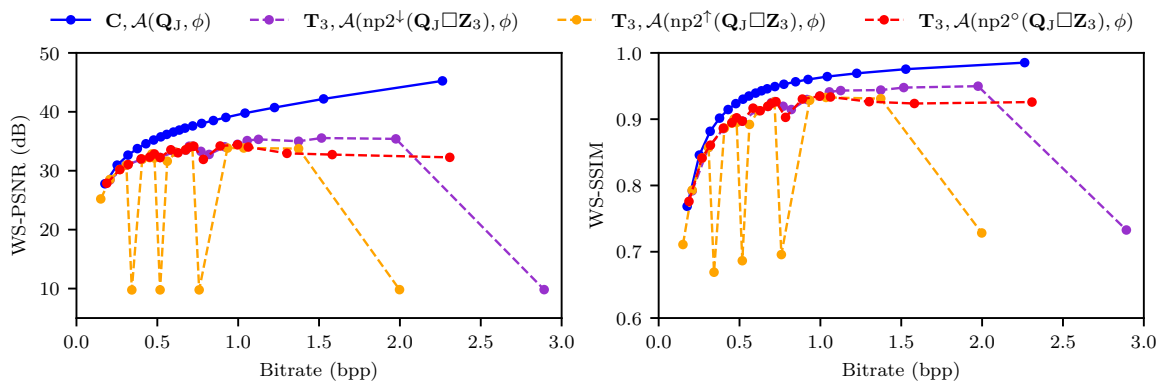
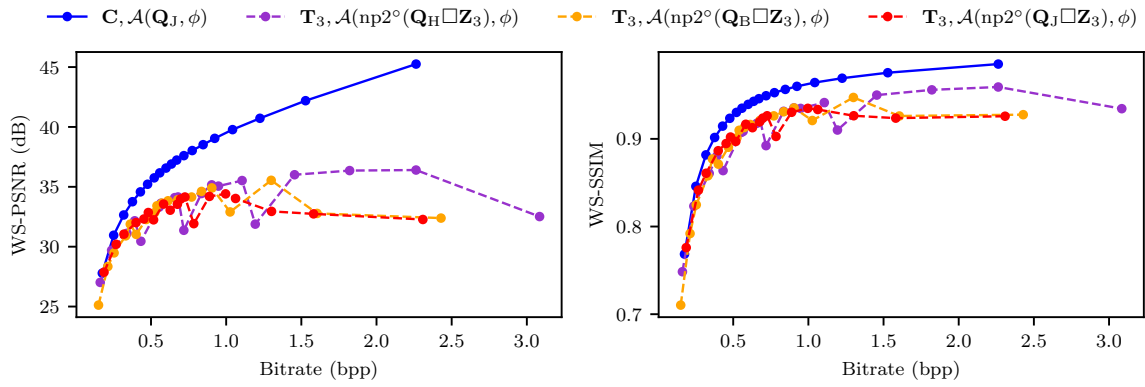
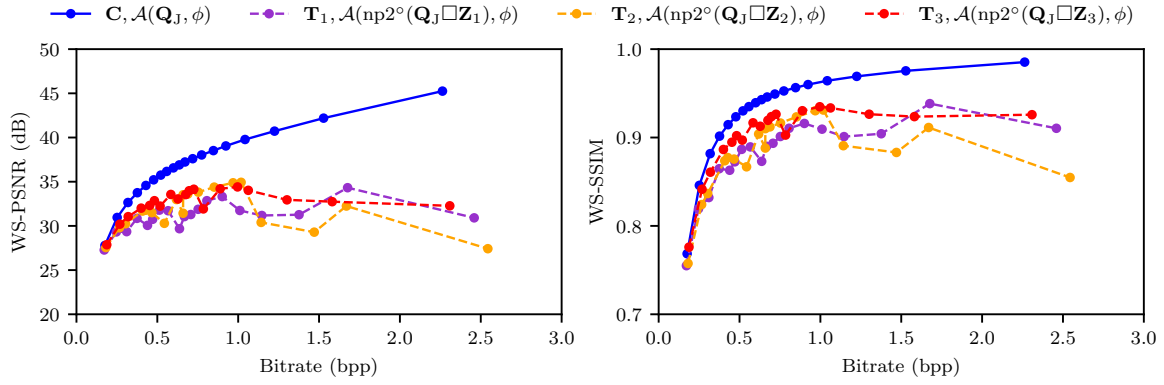


Figure 5.4 – Average RD curves for spherical image compression. Comparison between Simone et al. (2016) (C, Q_ϕ) and low-complexity variants of the proposed method.

are shown in Figures 5.5c and 5.5d. Compared to our method when using \mathbf{Q}_J , \mathbf{Q}_H implies in a performance loss of 1.83 dB, while \mathbf{Q}_B incurs a quality loss of 0.08 dB for the selected sample image. These results underscore the trade-offs between compression intensity and perceptual quality across the tested configurations. It is worth mentioning that switching the base quantization preserves the arithmetic complexity of our method.

Figure 5.4c shows the impact of considering different power-of-two rounding functions while keeping the other parameters of our method fixed. Note that $\text{np}2^\uparrow(\cdot)$, adopted in (BRAHIMI et al., 2021), performs poorly compared to $\text{np}2^\circ(\cdot)$, adopted in (OLIVEIRA et al., 2017; ARAAR; CHABBI, 2022). Also, $\text{np}2^\downarrow(\cdot)$ has inferior results than $\text{np}2^\circ(\cdot)$ for high bitrates. Although the $\text{np}2^\uparrow(\cdot)$ exhibit noticeable fluctuations in Figure 5.4c, the selected sample image captures the visual quality with no visual degradation (Figure 5.5e). Similarly, Figure 5.5f shows a qualitative results for when $\text{np}2^\downarrow(\cdot)$ is considered as power-of-two rounding function in our method. Compared to our method when using $\text{np}2^\circ(\cdot)$, the results from $\text{np}2^\uparrow(\cdot)$ and $\text{np}2^\downarrow(\cdot)$ vary in -0.08 dB and $+0.39$ dB for the selected sample image, respectively. The arithmetic complexity of our method keeps unchanged regardless of the adopted power-of-two rounding function. It is worth mentioning that we also tested combining different power-of-two rounding functions for the forward and backward quantization matrices. However, using $\text{np}2^\circ(\cdot)$ in both cases still performs the best.



(a) WS-PSNR: 30.87 dB and WS-SSIM: 0.86 at 0.39 bpp



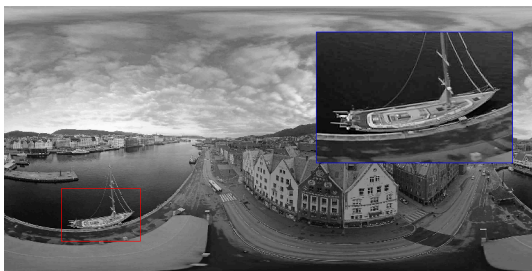
(b) WS-PSNR: 32.01 dB and WS-SSIM: 0.87 at 0.41 bpp



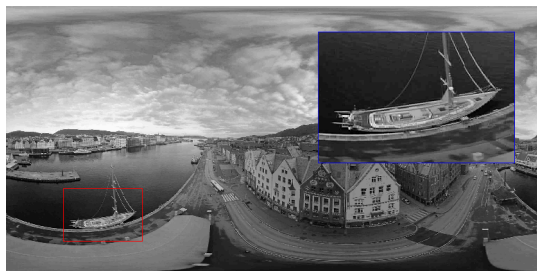
(c) WS-PSNR: 31.27 dB and WS-SSIM: 0.86 at 0.36 bpp



(d) WS-PSNR: 33.02 dB and WS-SSIM: 0.88 at 0.41 bpp



(e) WS-PSNR: 33.02 dB and WS-SSIM: 0.89 at 0.40 bpp



(f) WS-PSNR: 33.49 dB and WS-SSIM: 0.89 at 0.41 bpp

Figure 5.5 – Compressed Aerial City (BOYCE; ALSHINA; ABBAS, 2017) frame at roughly 0.4 bpp with varying transforms — (a) T_1 (OLIVEIRA et al., 2017) and (b) T_2 (BRAHIMI et al., 2020) —, base quantization matrices — (c) Q_H (ARAAR; CHABBI, 2022) and (d) Q_B (BRAHIMI et al., 2021) —, and power-of-two rounding functions — (e) $(np2^\uparrow(\cdot))$ (BRAHIMI et al., 2021) and (f) $(np2^\downarrow(\cdot))$. Results are best viewed with digital zoom.

6 CONCLUSION

To our knowledge, this is the first work addressing low-complexity compression of 360° images. We employed both approximate transforms and latitude-adaptive quantization. Our results demonstrate that the proposed method incurs a quality loss of no more than 4 dB at low bit rates (0.5 bpp or less) compared to its exact counterpart method, which involves complex arithmetic operations. We verified that our method is parametric, accepting different transforms and base quantization matrices. We showed that our approach is multiplication-free and offers a practical, hardware-friendly implementation requiring only additions and bit-shifting operations.

A first future investigation, which applies to planar low-complexity image compression, is how to avoid the valleys in the RD curves due to aggressive power-of-two rounding. Plus, although a hardware-friendly implementation was sought, future work may explore additional optimizations to the design. For instance, implementing the method on an FPGA would be a promising approach to assess its practical efficiency, enabling direct comparisons with current compression methods regarding chip area, throughput, energy consumption, etc. Another interesting avenue of research would involve selecting a QF -like parameter as the final step of the method, adding flexibility to our method, as well as the decomposition of the quantization matrix into sums of powers of two. Additionally, expanding this study to encompass 360° video compression would be relevant, given the specific challenges of this type of media, such as preserving visual quality across frames. Furthermore, exploring the impact of using different transform sizes in this context could provide valuable insights, as the choice of transform size can significantly influence compression efficiency and the retention of perceptual quality in varying regions (SILVEIRA et al., 2022).

It is worth mentioning that part of the results presented in this work has been accepted for presentation at the 16th IEEE CASS Latin American Symposium on Circuits and Systems (LASCAS). The article, titled "Low-Complexity Compression of 360° Still Images", authored by Bruno M. Bastos, Enzo B. Segala, and Thiago L. T. da Silveira, highlights the scientific relevance of this research by proposing efficient techniques for 360° image compression. These advancements are directly aligned with the objectives of this study, reinforcing its contribution to the field of visual data compression.

REFERENCES

- ARAAR, C.; CHABBI, S. Low-complexity JPEG quantization table requiring only bit-shift operations. **The Imaging Science Journal**, Taylor & Francis, v. 70, n. 8, p. 556–563, 2022. Available from Internet: <<https://doi.org/10.1080/13682199.2023.2173545>>.
- AZEVEDO, R. G. A. et al. Visual distortions in 360° videos. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 30, n. 8, p. 2524–2537, 2020.
- BAYER, F. M.; CINTRA, R. J. DCT-like transform for image compression requires 14 additions only. **Electronics Letters**, v. 48, n. 15, p. 919, 2012. ISSN 00135194.
- BIDGOLI, N. M. et al. OSLO: On-the-sphere learning for omnidirectional images and its application to 360-degree image compression. **IEEE Transactions on Image Processing**, IEEE, v. 31, p. 5813–5827, 2022. ISSN 19410042.
- BIDGOLI, N. M.; MAUGEY, T.; ROUMY, A. Intra-coding of 360-degree images on the sphere. **2019 Picture Coding Symposium, PCS 2019**, IEEE, p. 1–5, 2019.
- BOYCE, J.; ALSHINA, E.; ABBAS, A. **JVET Common Test Conditions and Evaluation Procedures for 360° Video**. Torino, Italy: [s.n.], 2017. Document JVET-G1030v2.
- BRAHIMI, N. et al. Designing multiplier-less jpeg luminance quantisation matrix. In: _____. [S.l.]: Springer International Publishing, 2021. v. 3, p. 683–691. ISBN 9783030923105.
- BRAHIMI, N. et al. A novel and efficient 8-point DCT approximation for image compression. **Multimedia Tools and Applications**, v. 79, p. 7615–7631, 2020. ISSN 15737721.
- CINTRA, R. J.; BAYER, F. M. A dct approximation for image compression. **IEEE Signal Processing Letters**, 2011. ISSN 10709908.
- EDER, M. et al. **Tangent Images for Mitigating Spherical Distortion**. arXiv, 2019. Available from Internet: <<https://arxiv.org/abs/1912.09390>>.
- FERREIRA, L. S.; SACHT, L.; VELHO, L. Local moebius transformations applied to omnidirectional images. **Computers and Graphics**, Elsevier BV, v. 68, p. 77–83, nov. 2017. ISSN 0097-8493. Available from Internet: <<http://dx.doi.org/10.1016/j.cag.2017.08.005>>.
- HE, Y. et al. Content-adaptive 360-degree video coding using hybrid cubemap projection. In: **2018 Picture Coding Symposium (PCS)**. [S.l.: s.n.], 2018. p. 313–317. ISSN 2472-7822.
- HUYNH-THU, Q.; GHANBARI, M. Scope of validity of PSNR in image/video quality assessment. **Electronics Letters**, v. 44, p. 800 – 801, 02 2008.
- JABALLAH, S.; BHAVSAR, A.; LARABI, M. Perceptual versus latitude-based 360-deg video coding optimization. In: **2020 IEEE International Conference on Image Processing (ICIP)**. [S.l.]: IEEE, 2020. p. 3423–3427.

LEE, S.-H. et al. Omnidirectional video coding using latitude adaptive down-sampling and pixel rearrangement. **Electronics Letters**, v. 53, n. 10, p. 655–657, 2017.

LEE, Y. et al. Spherephd: Applying cnns on 360° images with non-euclidean spherical polyhedron representation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 44, n. 2, p. 834–847, Feb 2022. ISSN 1939-3539.

LI, M. et al. End-to-end optimized 360° image compression. **IEEE Transactions on Image Processing**, Institute of Electrical and Electronics Engineers (IEEE), v. 31, p. 6267–6281, 2022. ISSN 19410042.

LIEBERMAN, K. et al. Neural image compression: Generalization, robustness, and spectral biases. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2023. v. 36, p. 77201–77241.

OLIVEIRA, P. et al. JPEG quantisation requires bit-shifts only. **Electronics Letters**, v. 53, p. 588–590, 4 2017. ISSN 0013-5194.

OLIVEIRA, R. S. et al. Low-complexity 8-point DCT approximation based on angle similarity for image and video coding. **Multidimensional Systems and Signal Processing**, Springer Science and Business Media LLC, v. 30, n. 3, p. 1363–1394, jul 2018.

POURAZAD, M. T. et al. HEVC: The new gold standard for video compression: How does HEVC compare with H.264/AVC? **IEEE Consumer Electronics Magazine**, v. 1, n. 3, p. 36–46, 2012.

RAO, K. R.; YIP, P. **Discrete Cosine Transform: Algorithms, Advantages, Applications**. [S.l.]: Academic Press, 1990. 512 p. ISSN 1098-6596. ISBN 0080925340.

RIZKALLAH, M. et al. Rate distortion optimized graph partitioning for omnidirectional image coding. **European Signal Processing Conference, EURASIP**, v. 2018-Septe, p. 897–901, 2018. ISSN 22195491.

SHAFI, R.; SHUAI, W.; YOUNUS, M. U. 360-degree video streaming: A survey of the state of the art. **Symmetry**, MDPI AG, v. 12, n. 9, p. 1491, sep. 2020. ISSN 2073-8994. Available from Internet: <<http://dx.doi.org/10.3390/sym12091491>>.

SILVEIRA, T. L. T. da et al. A class of low-complexity DCT-like transforms for image and video coding. **IEEE Transactions on Circuits and Systems for Video Technology**, Institute of Electrical and Electronics Engineers (IEEE), v. 32, n. 7, p. 4364–4375, jul 2022.

SILVEIRA, T. L. T. da; JUNG, C. R. Omnidirectional visual computing: Foundations, challenges, and applications. **Computers & Graphics**, Elsevier BV, v. 113, p. 89–101, jun. 2023.

SIMONE, F. D. et al. Geometry-driven quantization for omnidirectional image coding. In: **2016 Picture Coding Symposium (PCS)**. [S.l.]: IEEE, 2016. p. 1–5.

STANDARDISATION, I. O. for. **Information technologies – JPEG systems – Part 6: JPEG 360**. [S.l.], 2019. v. 2019. <https://www.iso.org/standard/75846.html>.

SUN, Y.; LU, A.; YU, L. Weighted-to-spherically-uniform quality evaluation for omnidirectional video. **IEEE Signal Processing Letters**, Institute of Electrical and Electronics Engineers (IEEE), v. 24, n. 9, p. 1–1, 2017. ISSN 1070-9908.

TORII, A.; IMIYA, A.; OHNISHI, N. Two- and three- view geometry for spherical cameras. In: . [S.l.: s.n.], 2005.

ULLAH, F.; KWON, O. J.; CHOI, S. Generation of a panorama compatible with the JPEG 360 international standard using a single PTZ camera. **Applied Sciences (Switzerland)**, v. 11, n. 22, 2021. ISSN 20763417.

USC-SIPI. **USC-SIPI Image Database**. Los Angeles, CA, USA: [s.n.], 1997. Available at <<http://sipi.usc.edu/database/>>.

WALLACE, G. K. The JPEG still picture compression standard. **IEEE Transactions on Consumer Electronics**, v. 38, n. 1, p. xviii–xxxiv, 1992.

WANG, F.-E. et al. **Self-Supervised Learning of Depth and Camera Motion from 360° Videos**. arXiv, 2018. Available from Internet: <<https://arxiv.org/abs/1811.05304>>.

WANG, Z. et al. Image quality assessment: from error visibility to structural similarity. **IEEE Transactions on Image Processing**, v. 13, n. 4, p. 600–612, April 2004. ISSN 1941-0042.

XIU, X.; HE, Y.; YE, Y. An adaptive quantization method for 360-degree video coding. In: **Applications of Digital Image Processing XLI**. [S.l.]: SPIE, 2018. v. 10752, p. 317–325.

YOUVALARI, R. G.; AMINLOU, A.; HANNUKSELA, M. M. Analysis of regional down-sampling methods for coding of omnidirectional video. **2016 Picture Coding Symposium, PCS 2016**, IEEE, p. 1–5, 2017.

ZHANG, M. et al. An efficient coding algorithm for 360-degree video based on improved adaptive QP compensation and early CU partition termination. **Multimedia Tools and Applications**, n. 1, p. 1081–1101, 2019. ISSN 15737721.

ZHAO, X. et al. Transform coding in the VVC standard. **IEEE Transactions on Circuits and Systems for Video Technology**, Institute of Electrical and Electronics Engineers (IEEE), v. 31, n. 10, p. 3878–3890, oct. 2021.

ZHOU, Y. et al. Weighted-to-spherically-uniform SSIM objective quality evaluation for panoramic video. In: **IEEE International Conference on Signal Processing**. [S.l.]: IEEE, 2018. p. 54–57.