Instituto Politécnico de Tomar
Escola Superior de Tecnológica de Tomar

Gonçalo Nunes Farinha

Melhoria da Qualidade de Imagem em Codificação com Múltiplas Descrições Usando Técnicas de Aprendizagem Profunda

Relatório de Projeto

Orientado por:
Pedro Daniel Frazão Correia

Relatório de Projeto apresentada ao Instituto Politécnico de Tomar para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrotécnica

Tomar, novembro de 2022

Dedico este trabalho aos meus pais.

RESUMO

As redes neuronais têm tido sucesso em vários ramos do processamento de informação em particular no processamento de imagem. As aplicações vão desde os métodos de melhoria de imagem, que incluem redução de ruído, correção de cor, restauro de imagem ou técnicas de super-resolução, as técnicas de visão computacional, que incluem a segmentação, a classificação de objetos, o seguimento de pessoas e veículos em vídeo, assim como aplicada aos métodos de codificação. Dentro das redes neuronais, as redes convolucionais (CNN) são as mais frequentemente utilizadas para obter soluções no processamento de imagem devido à sua capacidade de reconhecer padrões.

A codificação de imagem com múltiplas descrições (MDC), surge num contexto de esquemas de codificação para transmissão resiliente a erros de transmissão, onde uma imagem é dividida em várias componentes (descrições) que são codificadas de forma independente. No descodificador, as descrições são descodificadas de forma independente e combinadas entre si, produzindo uma representação da imagem de boa qualidade. Se cada descrição for descodificada de forma individual, a representação da imagem original tem uma qualidade inferior, embora aceitável.

Neste contexto, o presente relatório apresenta o estudo e aplicação de técnicas de aprendizagem profunda baseadas em CNN como método de pós processamento para melhoria da qualidade de imagens codificadas com múltiplas descrições (MDC), tendo em conta a possibilidade de descodificação conjunta ou independente das descrições constituintes da imagem.

O trabalho realizado apresenta primeiramente um esquema de codificação de imagem usando MDC com subamostragem por blocos para produzir várias descrições da imagem. Como pós processamento, são utilizadas de redes CNN pré-treinadas para melhoria da qualidade das descrições descodificadas onde a informação original é inexistente, utilizando técnicas de restauro de imagem (inpaiting) e de super- resolução. Os resultados obtidos apresentam melhorias significativas face a métodos tradicionais de interpolação.

Adicionalmente, é proposto um esquema MDC para duas descrições, usando MDC balanceado e não balanceado, isto é, com descrições codificadas com qualidades iguais e

com qualidades diferentes respetivamente. É apresentado um método de processamento para melhoria da qualidade das imagens descodificadas, baseado em métodos existentes para redução de ruído, no entanto treinados para o tipo de distorção introduzida pela codificação MDC. Os resultados obtidos revelam que se consegue melhorar a qualidade das imagens descodificadas, comparando com os métodos de interpolação tradicionais. Adicionalmente o método tem a capacidade de compensar a perda de qualidade de uma descrição nos casos de MDC não balanceado.

Palavras-chave: redes neuronais convolucionais, aprendizagem profunda, processamento de imagem, codificação de imagem, codificação com múltiplas descrições.

ABSTRACT

Neural networks have been successfully applied in several information processing areas, in particular, image processing. The application areas comprise image enhancement methods, which includes noise reduction, color correction, image inpainting or super-resolution techniques, computer vision techniques, which includes segmentation, object classification or people and vehicles tracking, as well as applied to the image coding methods. Regarding neural networks, convolutional networks (CNN) are the most frequently used to image processing due to their ability to recognize patterns.

Image multiple descriptions coding (MDC) is used in the context of coding schemes for resilient transmission due to transmission errors, where an image is divided into several components (descriptions) that are coded independently. At the decoder, the descriptions are independently decoded and combined, producing a representation of the image with good quality. If each description is decoded individually, the representation of the original image has a lower acceptable quality.

In this context, this report presents the study and application of deep learning techniques based on CNN as a post-processing method for improving the quality of MDC images, considering the possibility of joint or independent decoding of the image descriptions.

The present work firstly presents an image coding scheme using MDC with block subsampling to produce several image descriptions. As post-processing, a pre-trained CNN network is used to improve the quality of the decoded descriptions where the original information is non-existent, using image inpainting and super-resolution techniques. The obtained results show significant improvements compared to traditional interpolation methods.

Additionally, an MDC scheme for two descriptions is proposed, using balanced and unbalanced MDC, i.e. with descriptions encoded with equal and different qualities respectively. A post-processing method is proposed to improve the quality of the decoded images, using existing methods CNN for noise reduction, however trained for each type of distortion induced by MDC. The obtained results show that the quality of the decoded

images is improved, comparing with the traditional interpolation methods. Furthermore, the method can compensate the loss of quality in cases of unbalanced MDC.

Keywords: convolutional neural networks, deep learning, image processing, image coding, coding with multiple descriptions.

AGRADECIMENTOS

Nesta secção é dedicada a todos aqueles que me apoiaram ao longo destes anos, que possibilitaram a conclusão deste trabalho. Primeiramente ao orientador Prof. Pedro Correia pelo apoio e pela disponibilidade.

Agradecer ao Instituto Politécnico de Tomar e a todos os meus professores ao longo do meu percurso académico nesta instituição.

Queria também agradecer à minha família pela paciência e pelo apoio e agradecer aos meus amigos e colegas.

.

ÍNDICE

ÍNDICE DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CNN - *Convolutional neural network*

RD - *Rate - Distortion*

RNC - Rede neuronal convolucional

RNP - Rede neuronal profunda

JPEG - *Joint Photographic Experts Group*

HEVC - *High Efficiency Video Coding*

ML - *Machine learning*

DL - *Deep learning*

ANN - *Artificial neural network*

ReLU - *Rectified linear unit*

RL - *Rectified linear unit*

FNN - *Feed forward neural network*

RNN - *Recurrent neural networks*

AE - *Autoencoder*

TAID - *Task-aware image down-scaling framework*

LR - *Low-resolution*

SR - *Super-resolution*

TAD - *Task-aware down-scaled image*

TAU - *Task-aware up-scaled image*

VDSR - *Very deep convolutional networks*

RGB - *Red, blue, green*

Y - *Luminance*

CbCr - *Chrominances*

HD - *High-definition*

BPG - *Better portable graphics*

BN - *Batch normalization*

PSNR - *Peak signa-to-noise ratio*

Bpp - *Bits per pixel*

QF - *Quality factor*

QP - *Quantization parameter*

CPU - *Central processing unit*

GPU - *Graphics processing unit*

CNN-SR - CNN *for super-resolution*

DCT - *Discrete cosine transform*

GAN – *Generative Adversarial Networks*

RAN – *Recursive Attention Network*

RCAN – *Residual Channel Attention Network*

IBP – *Iterative Back-Projection*

MWRN – *Multi-level Wavelet Residual Network*

1. INTRODUÇÃO

Este capítulo apresenta a introdução do presente relatório, descrevendo a motivação e contexto, os objetivos e as contribuições resultante do trabalho desenvolvido.

1.1 Motivação e Contexto

As técnicas de melhoria de imagem têm sido desenvolvidas de modo a resolver problemas de aquisição de imagem, devido a limitação física das câmaras ou condições de iluminação deficientes, levando a imagens ruidosas ou com problemas com o contraste de cor. Também em aplicações para *smarphone* e outros dispositivos, podem ser aplicadas várias operações às imagens captadas pelas câmaras, desde filtros de cor ou alterações da resolução da imagem aplicados de forma automática ou semi-automática controladas pelo utilizador. Por outro lado, quando se aplica compressão e transmitem imagens à distância, é introduzida distorção que de alguma forma pode ser minimizada usando técnicas de melhoria de imagem. De igual forma as técnicas de técnicas de restauro de imagem podem ser usadas de modo a minimizar efeitos de erros de transmissão.

A utilização de técnicas de aprendizagem profunda ao processamento de imagem tem sido amplamente difundida, quer nas técnicas de melhoria de imagem, técnicas de visão por computador, em particular na classificação de objetos e mais recentemente para a codificação de imagem e vídeo, melhorando o desempenho destas aplicações relativamente às técnicas tradicionais existentes nas diferentes áreas de processamento de imagem. Não sendo solução para todos os problemas relacionados, a especialização em aprendizagem profunda aplicada a processamento de imagem torna-se um instrumento importante para nas diferentes áreas de conhecimento e aplicações, que podem ir desde a arqueologia até às ciências biomédicas.

No contexto da codificação de imagem, têm sido propostos vários esquemas de codificação de imagem baseadas em métodos de aprendizagem automática, que tentam incorporar os diferentes blocos de codificação e descodificação num mesmo processo de aprendizagem [1]. Uma outra abordagem possível, não sendo tão eficiente, consiste em

usar técnicas de pós-processamento para a melhoria da qualidade das imagens descodificadas resultantes de codificadores de imagem convencionais [2].

Em particular, num paradigma de Codificação com Múltiplas Descrições (MDC), uma imagem é dividida em várias componentes (descrições) que são codificadas de forma independente. No descodificador, cada descrição é descodificada de forma independente. Estas podem ser combinadas, produzindo uma representação da imagem de boa qualidade. Se cada descrição for descodificada de forma individual, a representação da imagem original tem qualidade inferior, embora aceitável [3].

O presente relatório apresenta a utilização de aprendizagem automática como pós processamento para a melhoria de imagens codificadas utilizando o paradigma de codificação com múltiplas descrições. Várias abordagens de codificação MDC são consideradas e em função disso, serão aplicadas técnicas de aprendizagem profunda para processos de restauro de imagem, super-resolução e redução de ruído.

1.2 Objetivos

Pretendem-se usar técnicas de codificação e descodificação de imagens com múltiplas descrições usando codificadores de standard, nomeadamente JPEG e JPEGXL.

Como pós processamento, pretende-se explorar técnicas de restauro de imagem, de super-resolução e de redução de ruído usando aprendizagem de profunda em codificação com múltiplas descrições (MDC), em diversas condições, nomeadamente, quando não existe no descodificador toda a informação original, mas também quando existe toda a informação codificada e se pretende melhorar a qualidade da imagem através da redução de ruído.

1.3 Contribuições

Do trabalho desenvolvido apresentam-se as seguintes contribuições:

- Estudo das técnicas de aprendizagem automática para melhoria de imagem utilizando *Convolutional Neural Networks* (CNN).

- Utilização de redes pré-treinadas para o restauro de imagens descodificadas onde a informação original é inexistente, utilizando técnicas de restauro de imagem (inpaiting) e de super- resolução, num contexto de codificação de imagem com múltiplas descrições.
- Codificador com múltiplas descrições com codificação balanceada e não balanceada, com melhoria da qualidade da imagem usando CNN para redução de ruído na descodificação MDC.

1.4 Estrutura do relatório

O presente relatório está organizado da seguinte forma. O capítulo 1 apresenta os objetivos, o contexto e motivação assim como as contribuições do trabalho desenvolvido. O capítulo 2 apresenta as técnicas de codificação de imagem, e o capítulo 3 descreve os conceitos principais de redes neuronais convolucionais assim como o estado da arte dos métodos de melhoria de imagem com CNN. O capítulo 4, explora as técnicas de melhoria de imagem codificadas em MDC usando redes pré-treinadas genéricas para melhoria de imagem de restauro e de super-resolução. O capítulo 5 apresenta o processo de codificação de imagem com múltiplas descrições com melhoria de imagem usando a rede DnCNN para redução de ruído, com os resultados obtidos usando o treino e teste da rede aplicado ao caso de estudo. O capítulo 6 conclui o relatório.

2. TÉCNICAS DE CODIFICAÇÃO E PROCESSAMENTO DE IMAGEM

Este capítulo aborda os aspetos relacionados com a codificação de imagem, incluindo os formatos de cor, as técnicas de codificação, assim como uma referência breve às diferentes normas de codificação de imagem, com perdas e sem perdas. Em particular, é apresentado uma descrição breve das normas JPEG e JPEG XL. Numa segunda parte do capítulo é descrito a codificação com múltiplas descrições (MDC).

2.1 Introdução de conceitos da codificação de imagem

Ao longo dos anos a codificação de imagem tem evoluído na medida da necessidade de comprimir a informação para ser processada, armazenada e transmitida pelos sistemas de comunicação.

Podemos ter dois tipos de codificação, sem perdas e com perdas. Na codificação sem perdas, é usada a estatística da fonte para permitir a codificação eficiente e assim reduzir a informação. Neste caso, a reconstrução da imagem é exatamente igual à original. A codificação com perdas, obedece a um esquema de codificação DPCM, usando quantificação associado a codificação de Huffman obtendo-se graus de compressão mais elevados.

2.1.1 Sistemas de cor

Para a representação de imagem, a codificação da cor obedece a uma combinação cores primárias, Vermelho, Verde e Azul, ou RGB, geradas pelos sistemas de reprodução de imagem. Deste modo, surgiram vários sistemas de cor que a partir da combinação das três cores, permitem gerar uma paleta de cores o mais aproximada possível das cores do espectro visível. Por outro lado, de modo a compatibilizar os sistemas atuais com os sistemas tradicionais de televisão analógica, existem outras formas de codificação da cor que dividem o sistema de cor em componentes de luminância (imagem a preto e branco) com as componentes de cor. Alguns exemplos de formatos de cor são o RGB, YCbCr, YUV e CMYK que são usados consoante o objetivo.

- **RGB** é usado para a apresentação de imagens digitais. Quando se observa alguma imagem num computador o mais provável é estar em estado RGB, uma vez que o monitor é composto por pixéis que são compostos por leds RGB, em que, é um composto das cores vermelho (*Red*), verde (*Green*), azul (*Blue*). Para alcançar a cor branca é necessário a combinação de todas as cores, enquanto que a cor preta é ausência de todas as cores.

- **YCbCr** é usado em codificação de imagem e vídeo. Y corresponde à componente de luminância e CbCr às componentes de cor, Cb os tons de azul e Cr os tons de vermelho. Este formato é importante para a codificação de imagem pois separa a luminância dos tons de cor, permitindo manipular esses valores de forma separada alcançando assim, uma maior eficiência de codificação uma vez que o sistema visual humano tem menor sensibilidade às componentes de crominância do que de luminância.

- O formato **YUV**, apesar de ser idêntico a YCbCr e serem relacionados linearmente, foi construído para implementar imagens coloridas numa infraestrutura (monitor) a tons de cinza. O formato YUV, permite várias resoluções de cada componente. Por exemplo, o formato YUV4:2:2 corresponde a codificadar por cada quatro píxeis de Y, dois píxeis de U e dois píxeis de V.

- **CMYK** é usado para a impressão, onde C corresponde ao Ciano, M a Magenta, Y a Amarelo e K a Preto. Na impressão já se assume o fundo a branco, logo não necessita dessa cor. O preto faz-se com a combinação de todas as cores, mas também tem o seu próprio 'tinteiro' uma vez que é bastante comum ser usado, tornando assim a impressão mais eficiente.

2.1.2 Sistema visual humano

Para a codificação de imagem, é necessário atender a algumas características de olho humano:

As mudanças de brilho são mais importantes que as mudanças de cor, em que a retina humana contém cerca de 120 milhões de células de bastonete sensíveis à luz e cerca de 6 milhões de células de cone sensíveis à luz.

Mudanças de baixa frequência são mais importantes que as mudanças de altas frequências. O olho humano é bom a notar as mudanças de baixas frequências (p.e. contornos dos objetos) e menos preciso a notar mudanças nas altas frequências (p.e. textura ou padrões). Por exemplo, a camuflagem resulta em parte porque as componentes de alta frequência vão perturbar as bordas das baixas frequências.

As normas de codificação de imagem exploram estas características para que as técnicas de codificação permitam extrair a informação da imagem que seja menos notada pela visão humana.

2.1.3 Transformadas

Uma técnica usada em codificação de imagem é o uso de transformadas. Estas tipicamente usam uma transformação da informação espacial (píxeis) para o domínio espectral (componentes de frequência), permitindo desta forma uma redução informação a codificar. Por exemplo, à semelhança da representação de sinais em Série de Fourier, com apenas um valor, é possível codificar a média dos píxeis de uma região da imagem (componente DC). Duas das transformadas mais utilizadas, são a DCT (*Discrete Cosine Transform*) e a DWT (*Discrete Wavelet Transform*).

- ***Discrete Cosine Transform* (DCT)**

Na codificação de imagem, a transformada DCT é aplicada tanto a blocos fixos de 8x8 pixéis de luminância, ou de dimensão variável dependendo do codec, depois da subamostragem das componentes de cor.

Na DCT de um bloco de pixéis 8x8, vão ser gerados 8x8 coeficientes de frequência, sendo que o primeiro (canto superior esquerdo) corresponde ao coeficiente DC e os coeficientes do canto inferior direito correspondem aos de frequências mais elevadas e de amplitudes mais baixas. A Figura 1, apresenta a distribuição espectral resultante da transformada DCT, onde o primeiro coeficiente (canto superior esquerdo) representa a componente DC, e os coeficientes próximos do canto inferior direito, representam os componentes de frequências mais elevadas. Por exemplo, numa imagem com bastantes componentes verticais iremos ter coeficientes com um maior valor na primeira linha, se for por exemplo um tabuleiro de xadrez iremos ter um maior valor de coeficientes no último coeficiente.

Associado ao processo da transformada existe a operação de quantização, que consiste em arredondar a zero os coeficientes de valor mais pequeno em função do seu passo. O passo de quantificação permite regular a característica débito-distorção e afeta tipicamente os coeficientes correspondentes a frequências mais elevadas. Como já foi mencionado anteriormente, o olho humano distingue pior as altas frequências de uma imagem, permitindo descartar essas componentes da imagem, comprometendo, no entanto, detalhe à imagem.

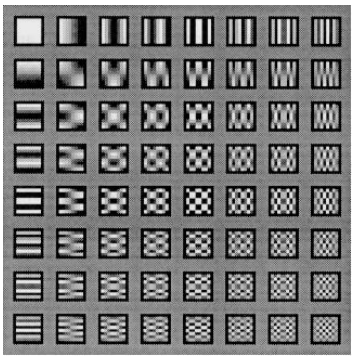


Figura 1 – Distribuição espectral resultante da DCT [47].

- ***Discrete Wavelet Transform* (DWT)**

A Figura 2 representa um esquema geral da transformada DWT 1-D para duas bandas. Nesta transformada para uma dimensão, à sequência x[n] é aplicada um banco de filtros, neste caso de duas bandas de frequência, onde h_0 vai corresponder a um filtro passa-baixo e h_1 corresponde a um filtro passa alto. A cada filtragem (convolução), é aplicada uma subamostragem. Este processo denomina-se de análise. Retirados os coeficientes da DWT e divididos pelas duas bandas, este processo e desenhado para que depois se tenha todos os coeficientes da sequência podendo assim reconstruí-la sem perdas.

A síntese é o processo inverso em que g_0 e g_1 vão corresponder a filtros passa-baixo e passa-alto para a reconstrução do sinal. Estes funcionam como filtros de interpolação, onde que se adiciona em ambas as bandas um zero entre cada amostra. No final as diferentes componentes são somadas.

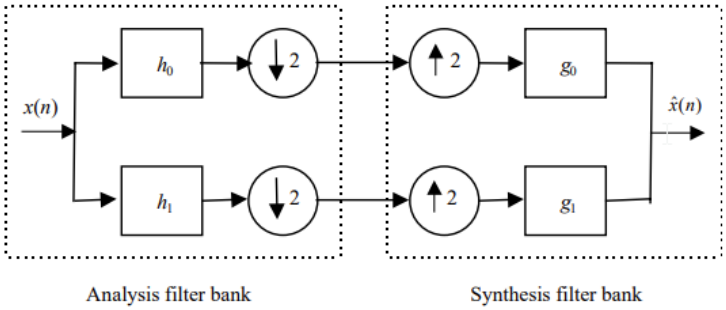


Figura 2- 1-D DWT com 2 bandas [4].

Na codificação de imagem é aplicado a DWT a duas dimensões, 2-D DWT. A Figura 3, representa a divisão em sub-bandas o resultante da DWT. Neste caso, o método é aplicado para ambas as dimensões (linhas e colunas), em que o resultado de aplicação de um nível resulta em 4 sub-bandas. Na figura, kHL, k corresponde ao nível, H vai corresponder a um filtro passa-alto aplicado às linhas e L corresponde a um filtro passa-baixo aplicado às colunas. Assim sendo, pode aplicar-se este processo sucessivamente até se parar de ver ganhos de eficiência.

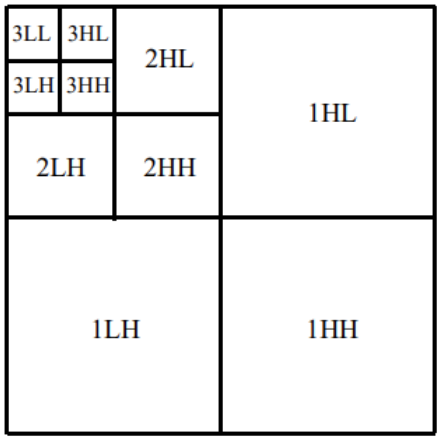


Figura 3 - 2-D com três níveis DWT [5].

2.1.4 Codificação de entropia

A codificação de entropia é uma componente importante nas técnicas de codificação e compressão de imagem. Esta pretende codificar de forma eficiente toda a sintaxe de codificação desde os coeficientes DCT aos cabeçalhos normalizados. Tipicamente pretende-se que a codificação de entropia use códigos de comprimento variável onde os

símbolos mais frequentes usam códigos mais curtos e símbolos menos frequentes usem códigos mais longos.

O tipo de codificação de entropia mais comum e conhecido é a codificação de Huffman. Esta usa símbolos consoante a frequência de certos eventos, por exemplo letras do teclado em que se usa mais umas que outras.

Char	Freq	Code
space	7	111
a	4	010
e	4	000
f	3	1101
h	2	1010
i	2	1000
m	2	0111
n	2	0010
s	2	1011
t	2	0110
l	1	11001
o	1	00110
p	1	10011
r	1	11000
u	1	00111
x	1	10010

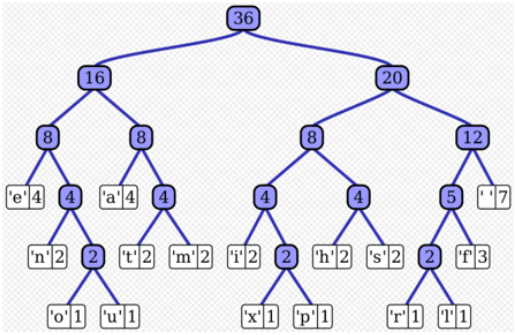


Figura 4 - Tabela de frequências e árvore de Huffman [46].

A Figura 4 apresenta um exemplo de codificação de Huffman. Neste exemplo, existem dezasseis caracteres, logo poder-se-iam representar com quatro bits usando código binário natural. Segundo a codificação de Huffman, como existem caracteres que se usam com mais frequência, é possível representar esses elementos com uma combinação mais pequena (três bits) e caracteres que se usam menos com uma combinação maior (cinco bits). A árvore de Huffman garante que cada combinação dá apenas um resultado. O argumento de Huffman é que se tem os três carateres com três bits com uma frequência de quinze e tem-se os seis caracteres com uma frequência de seis para uniformizar 15-6=9, que significa que escrevendo trinta e seis caracteres com os quatro bits tem-se 144bits. No entanto usando codificação de obtém-se 144-9=135bits, poupando aproximadamente 6% de informação sem perdas.

2.1.5 Predição espacial

Um preditor espacial é usado através de uma estimativa dos pixéis vizinhos. Subtraindo o valor estimado pelo valor real e codificando esses valores usando uma codificação entrópica como a de Huffman consegue-se assim uma melhor eficiência em termos de

espaço pois pode-se argumentar que na generalidade das imagens, os pixéis vizinhos têm valores próximos. Calculando essa diferença vamos usar na maior parte das vezes uma escala mais pequena. Pode ser usada DCT e codificação de Huffman às diferenças obtidas.

2.2 Formatos de Codificação

2.2.1 JPEG

O formato JPEG é um formato popular de codificação com perdas para imagem. A Figura 5 mostra o processo de codificação. Num primeiro passo, como brilho é mais importante para o olho humano, numa imagem RGB, o brilho vai estar distribuído pelas 3 cores, portanto é usado outra escala de cor YCbCr, em que Y vai corresponder ao brilho (Luminância), Cb aos tons de azul e Cr aos tons de vermelho. É realizada uma subamostragem da informação dos tons de vermelho e azul deixando o brilho ficar igual, logo retirando informação da imagem original uma vez que o sistema visual humano é menos sensível à cor do que à luminância.

A imagem é dividida em blocos de dimensão 8x8 pixeis, sobre os quais é aplicada a transformada **DCT** (*Discrete Cosine Transform*). Aos coeficientes DCT resultantes é aplicada a quantificação, que permite regular a qualidade de reconstrução da imagem. Os coeficientes são ordenados e codificados segundo a codificação de huffman, ou códigos de cumprimento variável. No processo de reconstrução, aos coeficientes codificados é aplicado a DCT inversa, reconstruindo os pixéis.

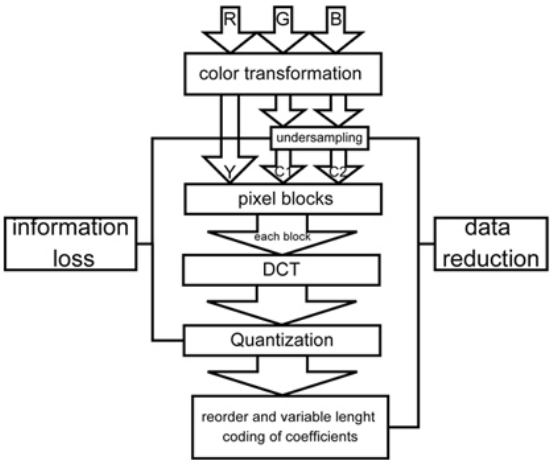


Figura 5 - Processo de codificação JPEG [44].

2.2.2 JPEG XL

A Figura 6 mostra a arquitetura da codificação JPGXL. Este *codec* é designado para ser um sucessor da codificação JPEG quer na *web*, como da fotografia profissional. Este *codec* apresenta melhor qualidade de imagem como maior número de recursos que permite um maior número de aplicações que o seu predecessor.

Algumas das características deste *codec* são ([10]): melhor eficiência e funcionalidade; transcodificação sem perdas; suporte para imagens fotográficas e sintéticas; boa qualidade a amplas gamas de *bitrates*; codificação sem perdas consegue reduzir o tamanho da imagem de 16% a 22%; consegue ser visualmente sem perdas até metade do *bitrate* do seu predecessor.

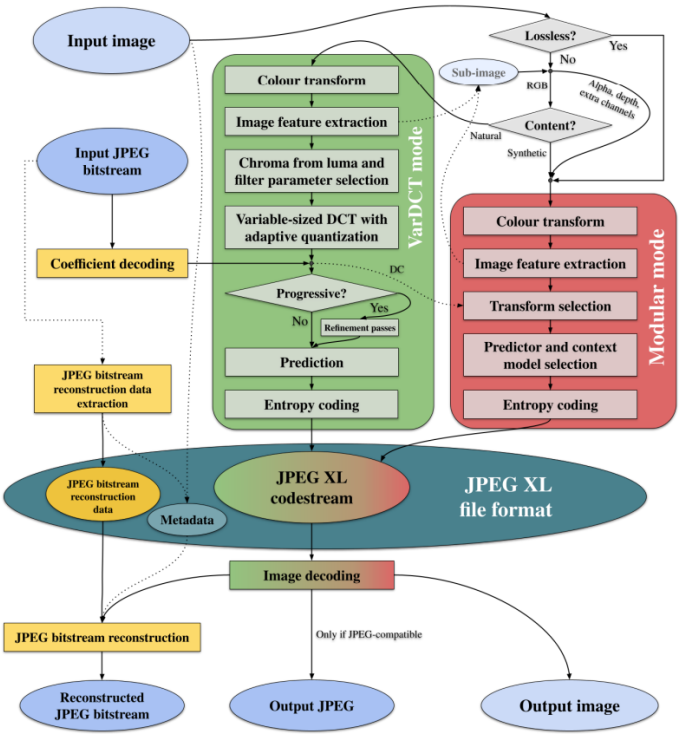


Figura 6 – Visão global da arquitetura do codificador JPEG XL [45].

2.3 *Multiple Description Coding* (MDC)

A codificação com múltiplas descrições (MDC) é um método de codificação, em que a fonte original, por exemplo uma imagem, é dividida em várias partes, chamadas descrições, que são codificadas e descodificadas de forma independente. Pretende-se que quando uma descrição é descodificada de forma independente, deve-se obtendo-se uma qualidade aceitável. Se existirem as duas descrições, estas podem ser combinadas de modo a obter uma qualidade superior. Neste processo, a eficiência de codificação é degradada quando comparada com a codificação com descrição única (SDC), no entanto espera-se que o esquema seja mais eficiente em ambientes com elevadas taxas de erro uma vez que cada descrição deve ser transmitida por canais distintos e assim ficar menos dependente dos erros existentes em cada canal.

A

Figura 7, apresenta o esquema genérico MDC aplicado a uma imagem. Neste caso é aplicada uma forma de construção das descrições, que pode ser por subamostragem [3], MDSQ [6], entre outras. Cada descrição é codificada, transmitida e descodificada por canais distintos. No descodificador, cada descrição pode ser descodificada de forma independente, obtendo-se as descrições D1 e D2, as descrições laterais, e podem ser combinadas, obtendo-se a descrição D0, ou descrição central.

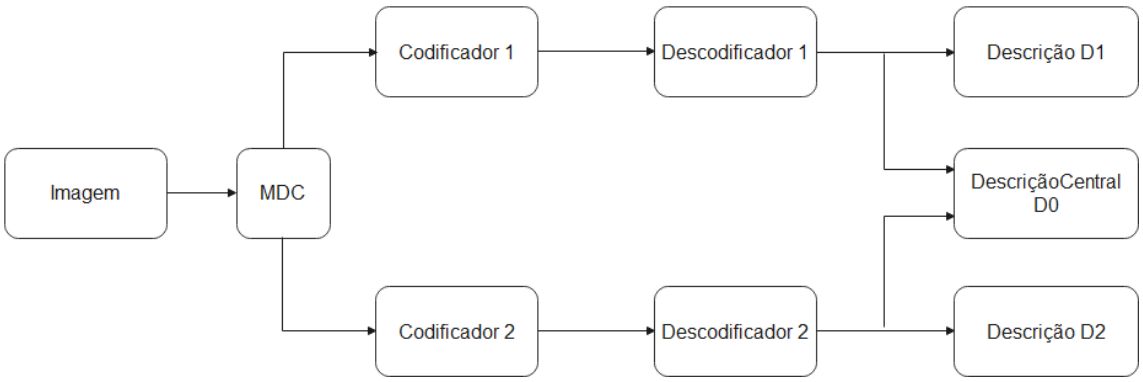


Figura 7 - Esquema de codificação MDC

2.4 Técnicas de processamento de imagem

Esta secção descreve de forma sucinta os conceitos de processamento de imagem que foram objeto de estudo para aplicação em redes neuronais profundas, nomeadamente, *Single Image Super Resolution* (SISR), Restauro (Inpainting) e Remoção de Ruído (Denoising).

2.4.1. *Single Image Super Resolution* (SISR)

O processo de Super-resolução vai consistir em recuperar uma imagem de maior resolução (HR) a partir de uma imagem com menor resolução (LR) [7]. Numa típica estrutura SISR, que está descrito na Figura 8, em que a imagem y (LR) é modelada da seguinte forma:

$$y = (x \otimes k) \downarrow_s + n,$$

Em que $(x \otimes k)$ vais corresponder à convolução entre x (HR) e a distorção associada pelo *kernel*, s é o fator de *downsampling*, e n é o ruído independente.

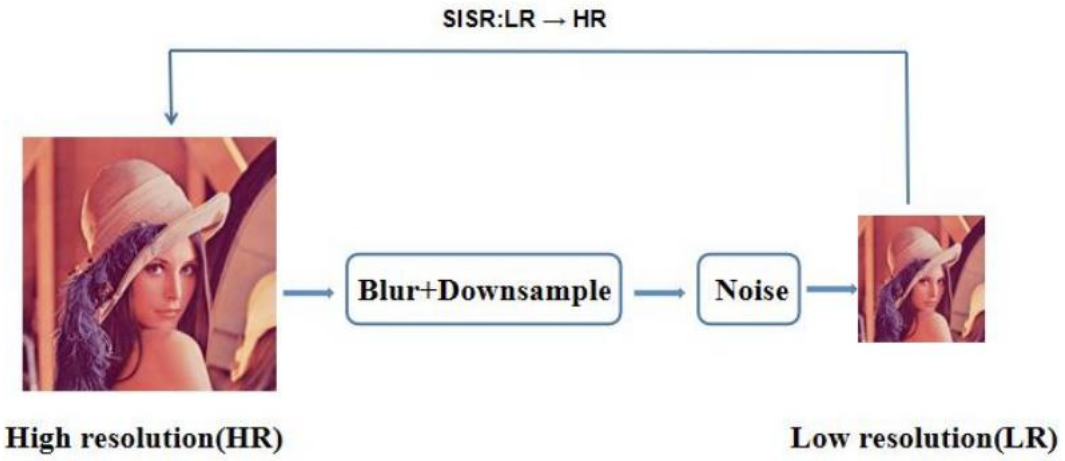


Figura 8 - Diagrama de Super-resolução [8]

2.4.2. Restauro (*Inpainting)*

O método de Restauro ou *Inpainting*, consiste no preenchimento da imagem quando não se tem acesso à informação de certos pixéis, ou seja, pode ser visto como uma máscara aplicada à imagem original que correspondente aos pixéis desconhecidos [9].

Como representado na Figura 9, a máscara vai indicar todos os pixéis por preencher, que neste caso estão distribuídos aleatoriamente.

Figura 9 - Exemplo do processo de *Inpainting* [10].

2.4.3. Remoção de Ruído *(Denoising)*

O processo de remoção de ruído ou *Denoising*, como o nome indica, consiste em gerar uma imagem com ruído minimizado a partir de uma imagem ruidosa, ao que segue o seguinte modelo:

$$y = x + n,$$

Ao que x é o objetivo, ou seja a imagem sem ruído, n corresponde ao ruído e y a imagem ruidosa [11]. A Figura 10 apresenta um exemplo de uma imagem ruidosa à esquerda e o seu melhoramento à direita

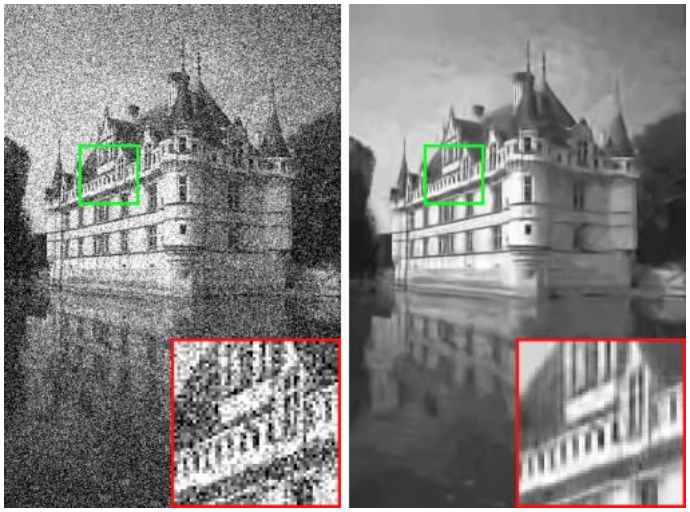


Figura 10 – Exemplo do processo de *Denoising* [12].

3. CONVOLUTIONAL NEURAL NETWORKS (CNN)

Este capítulo apresenta as noções de redes neuronais convolucionais, que são utilizadas para processamento de imagem nos vários tipos de aplicações, incluindo classificação de objetos, mas também técnicas de melhoria de imagem desde restauro (*inpaiting*), redução de ruido (*denosing*), super-resolução (SISR) e desfocagem (*debluring*). São apresentadas as constituintes das CNN e das operações envolvidas.

3.1 Modelo da rede neuronal artificial

Uma rede neuronal artificial é composta por várias camadas compostas de elementos básicos denominados de neurónio. A Figura 11 apresenta o chamado modelo de Perceptron, em que para cada nó (neurónio) está associado um conjunto de pesos relativos aos nós de camadas anteriores e o seu exclusivo "*bias*" em que passa por uma função de ativação e se tem a saída.

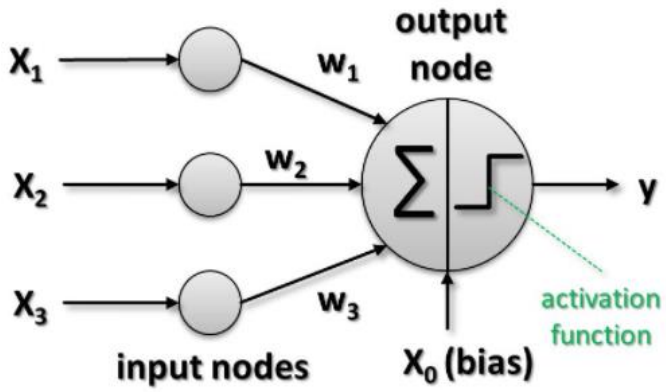


Figura 11 – Modelo do Perceptron [13].

A saída y será em função dos pesos, do *bias* e da função de ativação, ou seja, $y = \varphi(u)$ onde:

$$u = \sum_{i}^{m} x_i w_i + x_o$$

Onde w_i são os pesos, x_o corresponde ao *bias* e x_i são as entradas. A função $\varphi(u)$, é a função de ativação que corresponde a uma função não linear aplicada a cada neurónio.

A Figura 12 apresenta uma representação em camadas de uma rede neuronal, com uma camada de entrada, duas camadas escondidas e uma camada de saída de apenas um elemento. Neste caso a rede irá apresentar apenas uma saída com a resposta 0 ou 1, porque apesar de ser uma função, o seu processamento permitir dar respostas concretas em função da sua entrada.

Cada nó da primeira camada vai ter associado um valor dado por todos os valores da camada anteriores, multiplicado por peso (w), e somando um *bias* (b). Por exemplo, chamando a primeira camada h1 e a entrada i:

$$h1^{(1)} = i^{(1)}w^{(1)} + i^{(2)}w^{(2)} + i^{(3)}w^{(3)} + b$$

Note-se que (1) corresponde à posição do valor na camada de entrada. Associada a uma camada existe normalmente a respetiva ativação de valores, por exemplo a função *sigmoid* (σ), que transforma os valores para um intervalo de [0;1] ficando,

$$h1^{(1)} = \sigma(i^{(1)}w^{(1)} + i^{(2)}w^{(2)} + i^{(3)}w^{(3)} + b)$$

Esta operação é realizada sucessivamente para os outros nós. São esses valores w e b que vão ser atualizados durante o treino da rede, pois normalmente eles são inicializados aleatoriamente.

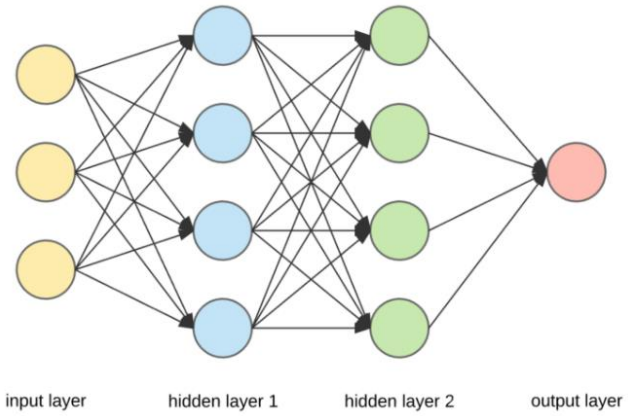


Figura 12 - Representação de uma rede neuronal [14].

Durante o treino, o que é feito é atualizar os valores dos pesos (w) e o bias (b) em função do Custo (C). O custo consiste em comparar os resultados dados pela rede com a verdade. Por exemplo, a função de custo pode ser a média dos erros quadrados, $C =$

$\frac{1}{N}\sum_{i=1}^{N}(o_i - v_i)^2$, com **c** sendo a função de custo ou erro, **o** é o output e **v** a verdade, isto pode mudar consoante a função de custo.

A atualização dos pesos e *bias* é feita com uma função que vai minimizar os resultados da função de custo, ou seja, minimizar o erro.

Para isso tem-se os gradientes de otimização, usando o exemplo em cima, mas imaginando que temos apenas um nó em cada camada, para efeitos de simplificação. Para saber a influência que uma mudança do peso relativo à camada do *output* (entre a segunda *hidden layer* e o *output*) tem no custo final temos:

$$\frac{\partial C(W)}{\partial w3(w)} = \frac{\partial C(W)}{\partial O} * \frac{\partial O}{\partial w3(w)}$$

Agora para saber a influência que uma mudança do peso relativo à primeira *hidden layer* (entre o *input* e a primeira camada) tem no custo final temos:

$$\frac{\partial C(W)}{\partial w1(w)} = \frac{\partial C(W)}{\partial O} * \frac{\partial O}{\partial w3(w)} * \frac{\partial w3(w)}{\partial h2} * \frac{\partial h2}{\partial w2(w)} * \frac{\partial w2(w)}{\partial h1} * \frac{\partial h1}{\partial w1(w)}$$

Vai ser neste processo de otimização que vai acontecer o fenómeno de '*back propagation*' pois o cálculo destes valores (w e b) vai ser feito começando nas camadas mais perto do output até às mais perto do *input* através das derivadas.

No processo de otimização também existe um parâmetro chamado de '*learning rate*' ou LR que vai ser importante. Em que a atualização dos pesos é dado por:

$$W \leftarrow W - \eta \frac{\partial C(W)}{\partial W}, \text{ sendo } \eta \text{ o LR.}$$

Visto que os pontos de inicialização dos pesos (w) são escolhidos aleatoriamente, como se pode ver na Figura 13, existem bastantes mínimos locais, o que com um LR baixo pode não ser possível obter assim o mínimo absoluto. Por outro lado, um LR alto pode resultar num *overshoot*. Pode ser visto na forma que a derivada determina a direção do caminho para o mínimo e o LR o tamanho do passo que se dá nessa direção.

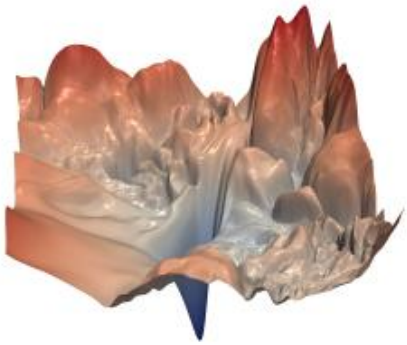


Figura 13 - Relevo dos custos [15].

3.2 Blocos constituintes das CNNs

Esta seção apresenta os blocos constituintes das CNN e as operações envolvidas. O modelo de neurónio apresentado anteriormente, quando aplicado a imagens, pode ser visto como um filtro espacial (convolução) que permite extrair uma determinada característica da imagem. Existindo várias camadas escondidas, compostas por vários neurónios, estes irão constituir por um conjunto de filtros que irão permitir extrair diferentes características da imagem, de modo a ser otimizados e aprendidos em função de um determinado objetivo e função de erro. As redes CNN são do tipo *Feedforward Neural Networks* (FNNs), ou seja não possuem realimentação [16].

3.2.1. Camada Convolucional

A camada convolucional vai servir para detetar padrões (através de filtros) e dar um sentido aos mesmos. Esta camada recebe uma entrada, transformando-a e enviando-a como saída para a próxima camada.

Uma rede neuronal convolucional pode ser pensada como uma composição de um número de funções.

$$f(x) = f_L(\dots f_2(f_1(x; w_1); w_2) \dots); w_L)$$

Em que cada função toma uma entrada x_l e um parâmetro w_l, em que este parâmetro vai ser atualizado com o processo de aprendizagem para resolver um problema.

Por exemplo, a Figura 14 apresenta um exemplo de filtragem ou convolução. Neste caso tem-se a entrada à esquerda, um filtro e a saída à direita. O filtro vai ser caracterizado pelo conjunto de parâmetros w que vai ser atualizado no processo de treino. Quando se seleciona a 'primeira' matriz 3x3 da entrada multiplicando elemento a elemento pelo filtro, e somando, obtém-se um valor para a nossa saída. A saída fica com as dimensões da entrada menos as dimensões do filtro - 1, neste caso, 28x28 -> 26x26, considerando um passo (*stride*) de 1. Como se pode ver na saída os dados vão até valores superiores a 1, e também podem ir até valores inferior a 0, dependendo do filtro. Por isso, depois de uma camada convolucional está normalmente presente uma ou mais camadas para normalização desses valores.

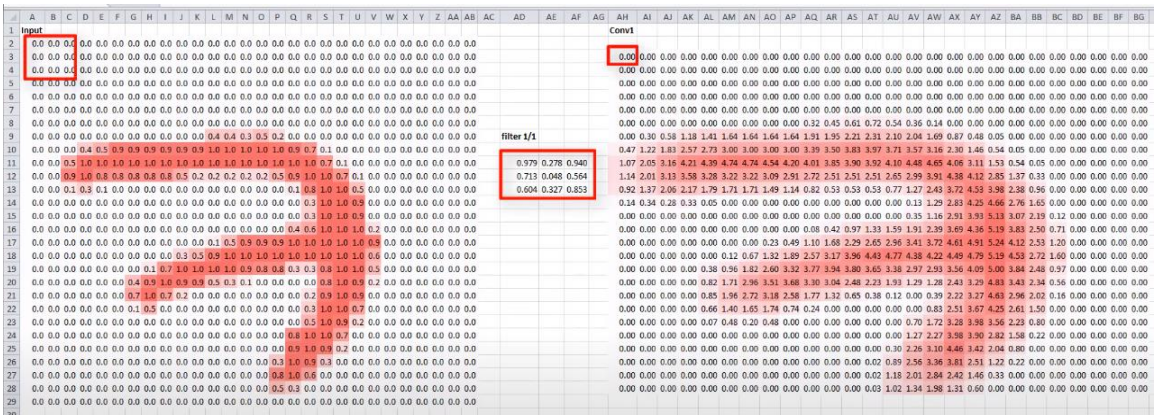


Figura 14 - Exemplo de filtragem [17].

Passo (stride) – É a distância que o centro do filtro vai tomar em cada filtragem na imagem. Por exemplo, na imagem da Figura 14, o *stride* é 1. Se fosse 2, a dimensão da saída seria metade resolução da entrada.

Padding – Acontece quando o filtro chega a valores que estão fora das dimensões do input, e aí temos de recorrer ao *padding* para preencher com valores, as estratégias mais usadas são: o *zero padding* e o *simetrical padding*, um preenche com zeros e o outro preenche de forma simétrica, respetivamente.

Dilatação – É outro parâmetro importante numa CNN. Na Figura 15 pode-se ver no lado direito uma dilatação com o fator 1 pois existe 1 espaço entre cada valor do filtro na aplicação.

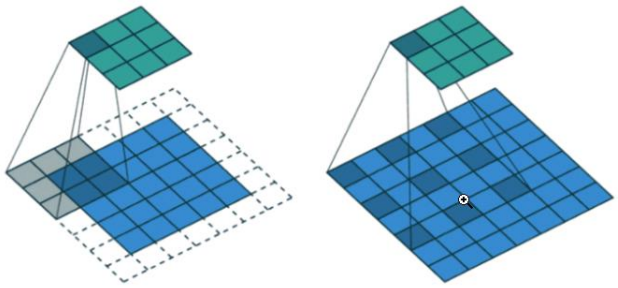


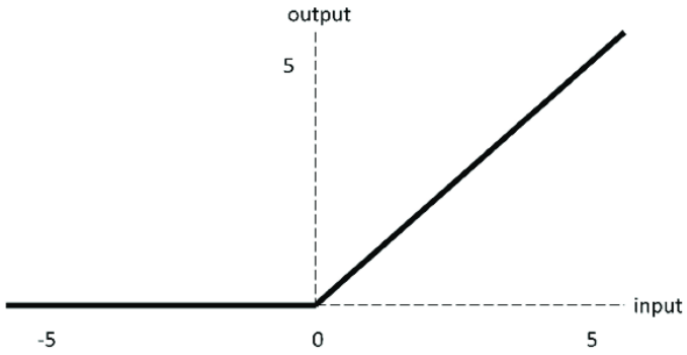
Figura 15 - Exemplo de dilatação [18].

A dilatação é importante pois em redes mais profundas é mais eficiente no processamento, sendo necessário também usar um *padding* adequado.

3.2.2. Funções de ativação não linear

Um exemplo de uma função de ativação não linear é a **ReLU** *(Rectified Linear Unit*)

$$y_{ijk} = \max(0, x_{ijk})$$



Esta operação atua como uma forma de controlo dos valores que são transferidos para a próxima camada, que vai facilitar o processo de treino.

3.2.3. Pooling

A camada ***Max pool*** vai servir mais para selecionar informação. Tem como equação:

$$y_{ijk} = \max(y_{i'j'k} : i \leq i' \leq i + p, j \leq j' \leq j + p)$$

A Figura 16 apresenta um exemplo de *pooling*. Como se pode observar, a aplicação de uma camada *max pool* que vai cortar o tamanho da entrada para metade na saída. O que faz consiste em selecionar o valor maior de cada quadrado 2x2. Existem também outras camadas semelhantes, como seja o *average pool* faz a média dos valores em vez de selecionar o maior. Tem o objetivo também de facilitar o treino.

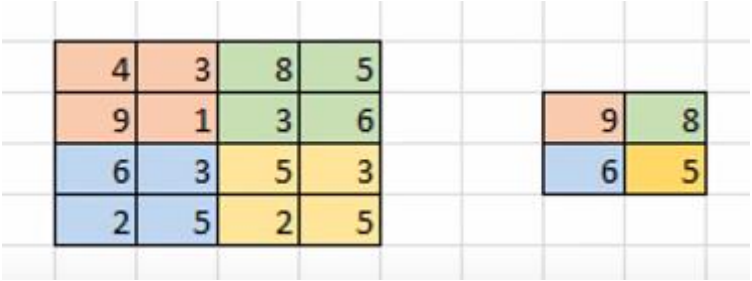


Figura 16 - Exemplo de pooling [19].

3.2.4. Normalização

Batch Norm (Batch Normalization) – A função desta camada, normalmente em conjunto com a ReLU, é normalizar os dados, pois numa rede podemos ter uma gama de valores mais elevada que pode causar instabilidade na rede.

Numa primeira parte pode usar a seguinte equação: $z = \frac{x-m}{d}$, em que **m** é a média, **d** o desvio padrão, **x** é a entrada e **z** a saída. Num segundo passo estes valores vão ser multiplicados por uma variável **g** e num terceiro passo vai se somar outra variável **h.**

Estes valores, m (média), d (desvio padrão), vão sendo atualizados em conjunto com o resto dos pesos e respetivos bias (Figura 11), reduzindo assim a influência de certos desequilíbrios na rede.

3.3 Trabalhos relacionados com técnicas de melhoria de imagem

O artigo [20], apresenta o método MIRNET-V2, onde os seus autores afirmam como sendo o estado da arte dos métodos baseados em aprendizagem profunda para as diferentes tarefas, desde redução de ruído, *defocus*, *deblurring*, *image denoising*, *super-resolution, and image enhancement*. Sendo um campo muito vasto de propostas, apresenta-se uma síntese de algumas propostas significativas das diferentes técnicas. Os quadros seguintes fazem um resumo dos métodos de processamento de imagem baseados em aprendizagem automática.

- **Redução de Ruído *(Denoising)***

Com o algoritmo NLM [21] já se consegue demonstrar uma performance promissora de redução de ruído ainda sem redes profundas. Recentemente com modelos de redes profundas [22] e [23] fizeram avanços significativos na redução de ruído de imagem, obtendo resultados mais favoráveis.

Tabela 1 – Exemplos de literatura utilizando o método de redução de ruído (*Denoising*)

Método	Arquitetura	Tipo de Dados
[21] – Buades et al. (2005)	CNN	Greyscle
[22] – Zamir et al. (2020)	CNN (MirNET)	RGB
[23] – Barnes et al. (2020)	CNN (RIDNet)	RGB
[24] – Brooks et al. (2018)	CNN	RGB
[25] – Guo et al. (2019)	CNN (CBDNet)	RGB
[26] – Kim et al. (2020)	CNN	RGB

- **Melhoria de Imagem (*Image enhancement*)**

Em algumas ocasiões, as camaras produzem imagens que carecem de contraste ou de cor. Para aprimoramento de imagem, a equalização de histograma é uma abordagem mais comumente usada. No entanto, frequentemente produz imagens sub ou superaprimoradas. A teoria de Retinex [27] com o uso de CNNs, recentemente motivou trabalhos que alcançaram bons resultados [28] [29] [30].

Tabela 2 - Exemplos de literatura utilizando o método de melhoria de imagem (*Image enhancement*)

Método	Arquitetura	Tipo de dados
[27] – Land et al. (1977)	--	--
[30] – Ignatov et al. (2019)	*	RGB
[28] – Zhang et al. (2019)	CNN	RGB
[29] – Shen et al. (2017)	CNN	RGB

*CNN/RAN/RCAN/IBP/MWRN

- **Super-Resolução**

Os primeiros métodos [31] e [32] usam uma imagem de baixa resolução (LR) como entrada e aprendem a formar diretamente sua versão de alta resolução (HR). Em contraste com a produção direta de uma imagem HR latente, as redes SR sucessoras [33], [34] empregam a estrutura de aprendizagem por residuo [35] para aprender os detalhes da imagem de alta frequência.

Tabela 3 - Exemplos de literatura utilizando o método de Super-Resolução

Método	Arquitetura	Tipo de dados
[31] – Dong et al. (2015)	CNN	RGB
[32] – Dong et al. (2014)	CNN	RGB
[33] – Kim et al. (2016)	CNN	RGB
[34] – Tai et al. (2017)	CNN	RGB
[35] – He et al. (2016)	CNN	RGB

- **Restauro de Imagem *(Inpainting)***

Começando pelo trabalho de IRCNN [12] que foi baseado num capítulo deste trabalho, já se consegue demonstrar bons resultados. Sucessivamente a rede encoder-decoder baseadas em CNN [36] e [37] conseguem recuperar bem os blocos perdidos com boa precisão em termos de estrutura e textura. Em Zhao et al. [38], foi proposto um trabalho para preenchimento de imagens médicas X-ray e [39] Nakamura et al, foi o trabalho proposto para remover texto de imagens.

Tabela 4 - Exemplos de literatura utilizando o método de Restauro de Imagem (*Inpainting*)

Método	Arquitectura	Data type
[38]– Zhao et al. (2018)	CNN	X-ray images
[39] – Nakamura et al (2017)	CNN	Images with text
[37]– Hertz et al (2019)	Encoder-Decoder	RGB
[36] – Yan et al (2018)	Encoder-Decoder	RGB

4. APLICAÇÃO DE CNN EM MELHORIA DE IMAGEM

Este capítulo apresenta um método para melhoria de imagem baseada no método de restauro de imagem, IRCNN, apresentada em [12], aplicado a um esquema de codificação de imagem com múltiplas descrições (MDC). Baseado numa CNN pré-treinada para melhoria de imagem, este método é aplicado na descodificação MDC por subamostragem, usando as técnicas de restauro de imagem (*image inpainting*) e de super-resolução. São apresentados resultados obtidos da aplicação dos métodos.

4.1. CNN para melhoria da qualidade de imagem - IRCNN

O processo de restauro de imagem consiste em gerar uma imagem limpa **x**, a partir de uma imagem degradada $\boldsymbol{y} = \boldsymbol{Hx} + \boldsymbol{v}$, onde **H** é a matriz de degradação e **v** é ruído aditivo branco gaussiano com desvio padrão σ. Em função da matriz de degradação, podem ser definidas diferentes funções de melhoria de imagem desde a redução de ruído ("*denoising*"), restauro ("*inpainting*"), super-resolução, ou "*debluring*".

Estes métodos pretendem sintetizar uma imagem limpa a partir de uma função de otimização que permita minimizar o erro

$$\hat{\boldsymbol{x}} = \arg\min_{x} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{Hx}\|^2 + lF(\boldsymbol{x})$$

Para resolver o problema definido pela equação, podem existir dois caminhos distintos. O primeiro consiste em usar modelos de otimização que resolvam diretamente a equação e permitam extrair a imagem com a menor distorção possível, envolvendo um método interativo de inferência computacionalmente pesado. A segunda metodologia consiste em usar aprendizagem discriminativa. Neste caso, pretende-se determinar (aprender) o conjunto de parâmetros **Θ** e uma inferência compacta, a partir da otimização de uma função de custo a partir de um conjunto de treino contendo pares de imagens limpas-degradadas. Esta função de custo é dada por

$$\min_{\mathbf{Q}} z(\hat{\boldsymbol{x}}, \boldsymbol{x}) \;\; s.t. \hat{\boldsymbol{x}} = \arg\min_{x} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + lF(\boldsymbol{x}; \mathbf{Q})$$

onde ζ é a função de custo.

Em [12], é apresentada uma CNN para restauro de imagem aplicada aos vários problemas inerentes, definidos por diferentes matrizes de degradação $\boldsymbol{H}$. A Figura 17, apresenta a arquitetura da rede CNN proposta para aplicações de restauro de imagem. Esta é composta por 7 camadas convolucionais, associadas a uma camada ReLU e 5 camadas BNorm+ReLU. Na figura, a-DConv com a=1,2,3,4,3,2,1 é referente ao fator de dilatação das camadas convolucionais, as camada ReLUs(max(0,x)) as camadas BNorm (*Batch Normalization*) + ReLU entre as camadas convolucionais.

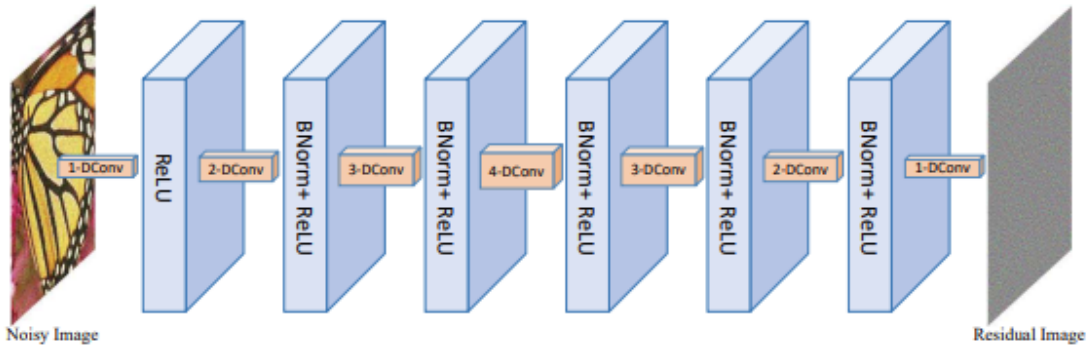


Figura 17 - Arquitetura da rede neuronal IRCNN

A Figura 18 - Aplicação de IRCNN para restauroFigura 18 mostra um exemplo da aplicação de restauro de uma imagem com uma máscara aleatória correspondentes aos pontos pretos na imagem à esquerda e a sua reconstrução à direita.

Figura 18 - Aplicação de IRCNN para restauro [12]

Tendo em conta estas funcionalidades e a partir da rede previamente treinada para cada aplicação disponível em [12], foi aplicado o método à utilização em MDC, sem treino específico para cada aplicação, que será descrito na secção seguinte.

4.2. Aplicação de IRCNN em MDC

A Figura 19 apresenta diagrama de blocos do esquema MDC associado ao método IRCNN descrito anteriormente para melhoria da qualidade da imagem reconstruída. Os métodos de melhoria de imagem são aplicados às imagens descodificadas, quando existe uma descrição ou então quando as descrições são combinadas entre si. Neste caso são apresentados dois métodos de melhoria de imagem, restauro (*inpainting*) e super-resolução, usando CNN pré-treinadas para processos genéricos de melhoria de imagem sem ter em conta as especificidades do processo de codificação.

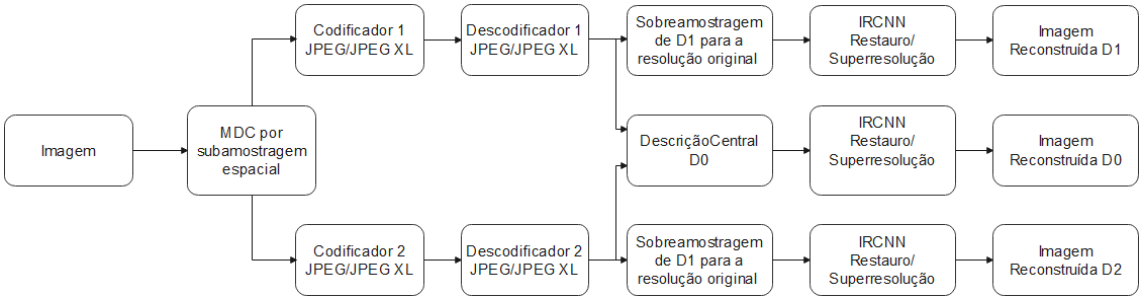


Figura 19 – Diagrama de blocos do processo de reconstrução da imagem.

De acordo com a Figura 19, a cada imagem é aplicada um método MDC por subamostragem, produzindo duas descrições. Essas descrições vão ser codificadas usando um *codec* JPEG ou JPEG XL, obtendo-se um determinado débito. Este será usado para aferir os resultados débito-distorção, expresso em PSNR versus bits por pixel (bpp).

O esquema MDC usado, é baseado numa subamostragem espacial, distribuído os pixéis originais pelas diferentes descrições, resultando em imagens codificadas com metade da resolução em largura e em altura. Note-se que este não é um esquema puro MDC uma vez que metade dos pixéis originais não são codificados, permitindo assim reduzir o débito resultante do esquema. São definidos dois métodos de subamostragem: i) por pixel e ii) por bloco. Espera-se que o método de pós processamento IRCNN permita melhorar a qualidade da imagem de reconstrução a níveis aceitáveis, incluindo alguns artefactos que resultam do facto do processo de subamostragem ser realizado sem filtragem.

Cada descrição é descodificada de forma independente. Existindo as duas descrições, estas são combinadas para obter uma imagem de melhor resolução. São aplicados dois métodos: restauro de imagem (*image inpainting*) e super-resolução. No método de restauro de imagem, os pixéis que foram retirados à imagem original para gerar as descrições vão ser colocados no lugar original, deixando os restantes pixéis em falta. O método de restauro tem a capacidade de preencher a informação em falta baseada nos pixéis vizinhos e assim melhorar a qualidade da imagem reconstruída. O método de super-resolução, usa a informação das descrições com menor resolução, e através de métodos de interpolação, neste caso interpolação bicúbica [40], restabelece a resolução original sendo aplicada de seguida o método de otimização de melhoria da qualidade da imagem.

A Figura 20 mostra um exemplo para uma imagem. O processo MDC gera duas descrições que são codificadas. Estas são combinadas, existindo ainda espaços da imagem em falta. Aplicando o método IRCNN, a imagem é reconstruída.

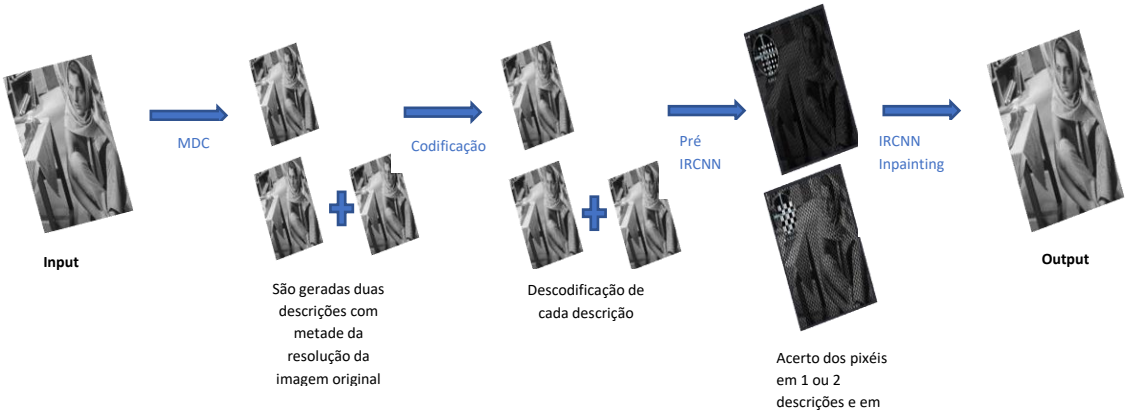
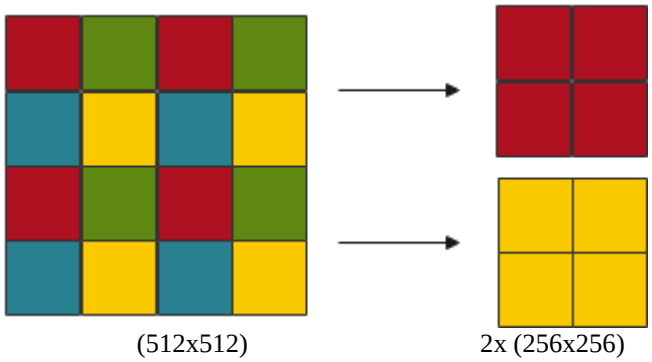


Figura 20 - Processo de MDC com restauração de imagem (*inpaiting*)

4.2.1. Codificação MDC por subamostragem

- **Subamostragem por pixel.**

Na subamostragem por pixel, por cada bloco de 2x2 pixeis, cada descrição é composta por um pixel desse bloco. Os outros dois pixéis não são codificados. A Figura 21, apresenta um exemplo. Cada descrição resulta em um quarto da resolução original, usando duas descrições ficando assim com uma imagem com metade da resolução original.



(512x512) 2x (256x256)

Figura 21 - Esquema MDC para duas descrições – Subamostragem por pixel.

- **Subamostragem por bloco.**

Na subamostragem por blocos, por cada bloco de 4x4 pixéis, é retirado dois blocos 2x2 pixéis para construir as duas descrições, isto porque consoante a resolução da imagem pode-se obter melhores resultados. A Figura 22 apresenta um exemplo de subamostragem por bloco.

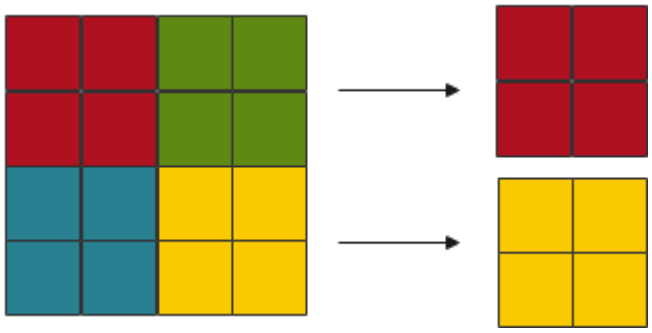


Figura 22 - Esquema MDC para duas descrições – Subamostragem por bloco.

A Figura 23, mostra um exemplo de uma descrição com um quarto da resolução espacial da imagem original.

Figura 23 – Imagem de uma subamostragem por pixel usando uma descrição.

A cada descrição foi aplicado uma codificação JPEG ou JPEG XL. A Figura 24 apresenta um resultado de uma codificação de uma descrição codificada em JPEG.

Figura 24 - Codificação da imagem subamostrada.

4.3. Descodificação MDC com IRCNN por restauro da imagem

A descodificação de uma descrição, ou das duas descrições, para aplicação do método IRCNN por restauro (*inpainting*), obedece a uma composição de imagem denominada de pré-CNN. Este método consiste em colocar os pixéis na posição original da imagem. Desta forma, obtém-se uma imagem com a resolução original, no entanto, com parte da imagem em falta, que será preenchida pelo processo de IRCNN. A Figura 25 mostra o processo de composição para duas descrições quando estas são construídas por subamostragem por pixel e por bloco respetivamente.

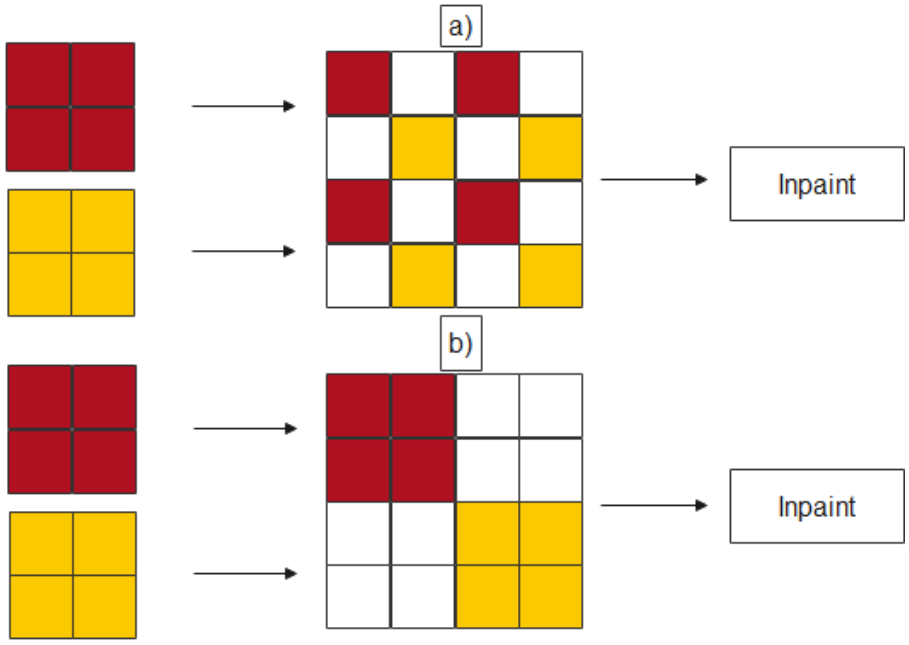


Figura 25 - Processo de pré-CNN com duas descrições. a) subamostragem por pixel; b) subamostragem por bloco.

A Figura 26 apresenta o processo de Pré-CNN em duas situações. Através da lupa, na imagem do meio (pré-CNN1) apresenta uma descrição com blocos 2x2 com uma descrição e subamostragem por pixel, ou seja, por cada quatro pixéis, existe apenas informação de um. Na imagem da direita (pré-CNN2) observa-se as duas descrições e uma subamostragem por bloco. Os pixéis a preto não existem e vão ser gerados pelo processo IRCNN.

Figura 26 - Pré-CNN no processo de inpainting

A Figura 27 apresenta o resultado da aplicação do método IRCNN, para as duas situações expressas na Figura 26. Observamos uma composição da imagem das regiões em falta. Os artefactos derivam da distorção resultante da codificação, assim como do conteúdo da imagem, que neste caso é composta por conteúdos de alta frequência, que podem não favorecer o processo. De notar que a rede aplicada, não foi treinada para este método específico.

Figura 27 – Aplicação do processo de inpainting.

4.3.1. Resultados Débito-Distorção

Esta secção apresenta da aplicação do método descrito nas seções anteriores, usando dois tipos de codificação de imagem, JPEG e JPEG XL. O método MDC apresentado usa subamostragem por pixel e subamostragem por bloco. O método IRCNN utilizado está disponível em [10], e está implementado usando o package de MATLAB, Matconvnet [41]. É usada a rede pré-treinada existente.

Para comparação dos resultados a medida de distorção utilizada é o PSNR (*Peak Signal to Noise Ratio*), definido como

$$PSNR = 10log_{10}\frac{255^2}{MSE}\ dB$$

onde

$$MSE = \frac{1}{M\times N}\sum_{i=1}^{N}\sum_{j=1}^{M}\left[I(i,j) - I_{Ref}(i,j)\right]^2$$

é o erro médio quadrático entre a imagem descodificada e a imagem de referência.

O número de bit por pixel (bpp) considerado é a relação do débito resultante da codificação de duas descrições, independente do caso de descodificação, relativamente à resolução original da imagem, ou seja,

$$bpp = \frac{R_{D1} + R_{D2}}{M \times N}$$

Os testes são realizados com as imagens apresentadas na Figura 28 - Set de imagens usadas nos testes de restauro.

Figura 28 - Set de imagens usadas nos testes de restauro de imagem.

As imagens apresentam-se por ordem de 1 a 6, em que: imagem 1, 'Lena' cinzento (512x512); imagem 2, 'Baby', cor (512x512); imagem 3 e 4, 'Monarch', cinza e cores respetivamente (256x256); imagem 5, 'Man eating', cinza (1024x1024); imagem 6, 'Xunantunich', cor (1024x1024).

São apresentados resultados da descodificação de uma descrição e de duas descrições, usando os métodos de subamostragem por pixel ou por bloco.

4.3.2. JPEG XL

Nesta seção são apresentados os resultados da reconstrução das imagens codificadas em JPEG XL [38], com uma descrição por pixel (1DPnn) e por bloco (1DBnn) e com duas descrições, por pixel (2DPnn) e por bloco (2DBnn). Os resultados comparam com a codificação JPEG XL da imagem ($SDCJPEG_{XL}$), assim como com a imagem obtida a partir de uma descrição usando apenas interpolação bicúbica ($1DP_b$, $1DB_b$, $2DP_b$, $2DB_b$).

Da Figura 29 à Figura 33, são apresentados os resultados débito distorção obtidos. Considerando a referência $SDCJPEG_{XL}$, os resultados obtidos dos métodos de restauro são cerca de 2dB abaixo em média, o que é natural, devido tanto à falta de informação original existente no descodificador, assim como pela redundância introduzida pela codificação das descrições.

No entanto verifica-se que comparada com o método de interpolação bicúbica, os métodos de restauro funcionam de forma bastante interessante uma vez que se obtêm ganhos de qualidade de reconstrução da ordem dos 3 a 4 dB, quando comparamos entre as várias formas codificação com o respetivo número de descrições.

Outro aspeto que se pode concluir dos resultados, prende-se com a qualidade obtida em função do tipo de codificação. Neste caso, verifica-se que a codificação por pixel é mais eficiente na reconstrução do que a descodificação por bloco, obtendo-se melhores resultados de débito-distorção. Este facto explica-se pelo facto de no caso de codificação por bloco, a área de imagem a reconstruir nas áreas que faltam, é maior, sendo mais difícil uma aproximação ao original do que comparada com a codificação por pixel.

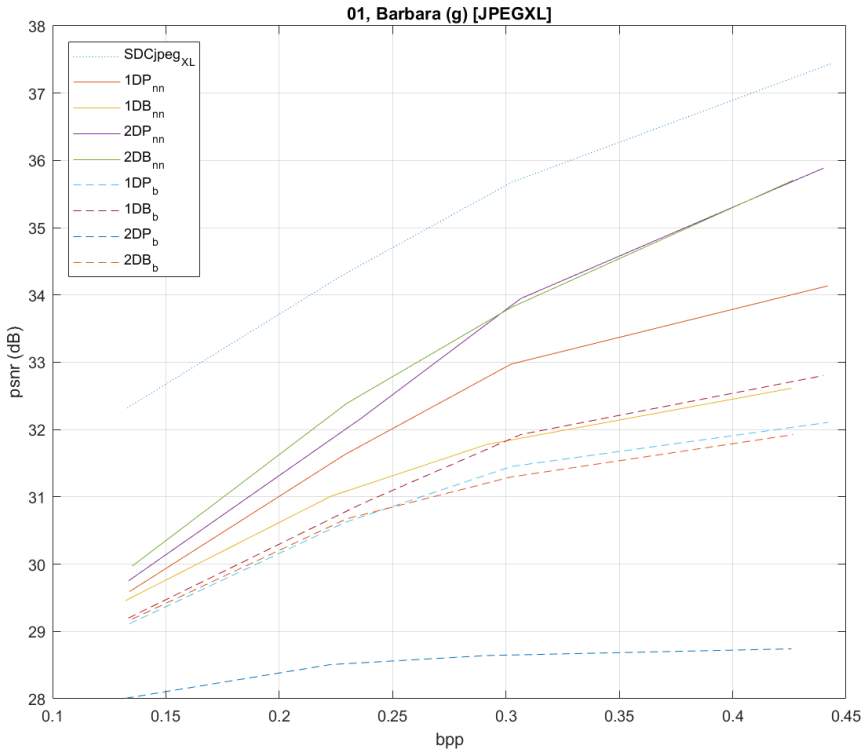


Figura 29- Resultados debito-distorção imagem 1, JPEG XL

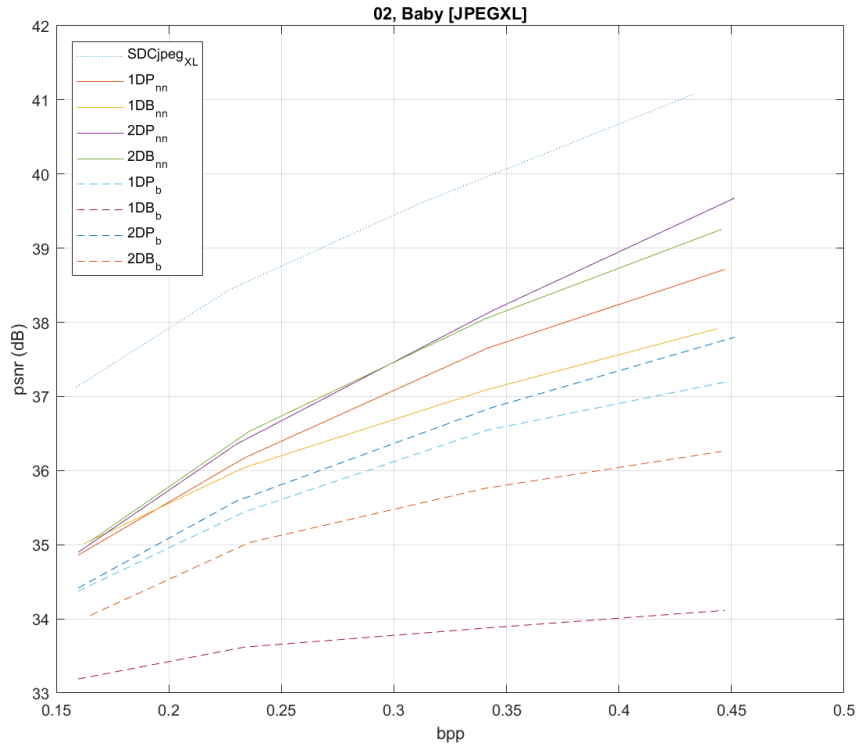


Figura 30 - Resultados debito-distorção imagem 2, JPEG XL

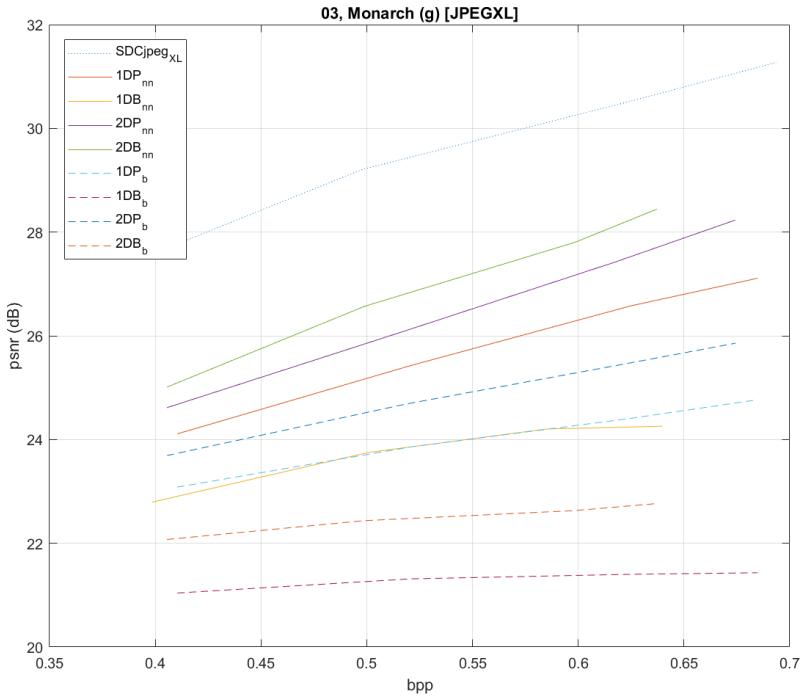


Figura 31 - Resultados debito-distorção imagem 3, JPEG XL

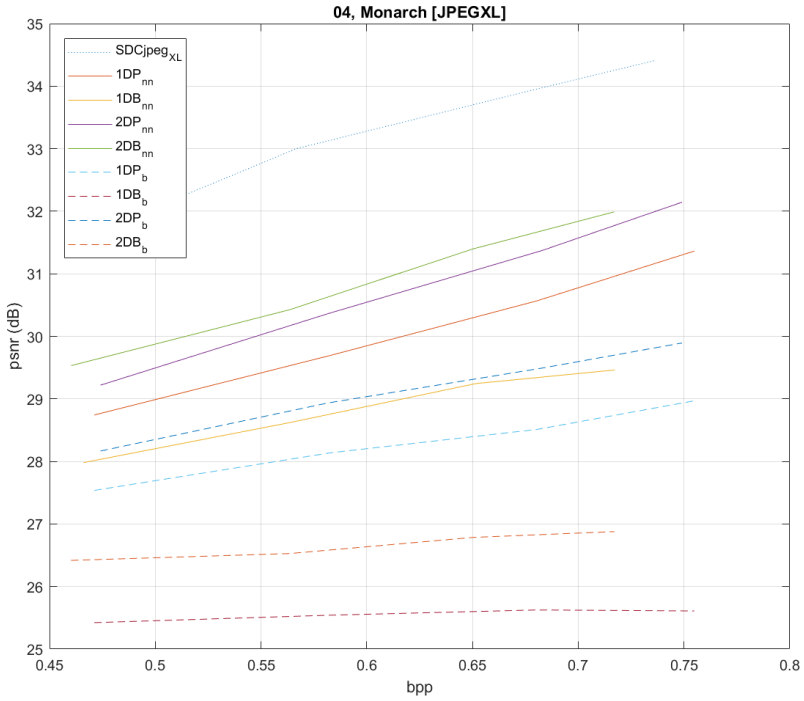


Figura 32 - Resultados debito-distorção imagem 4, JPEG XL

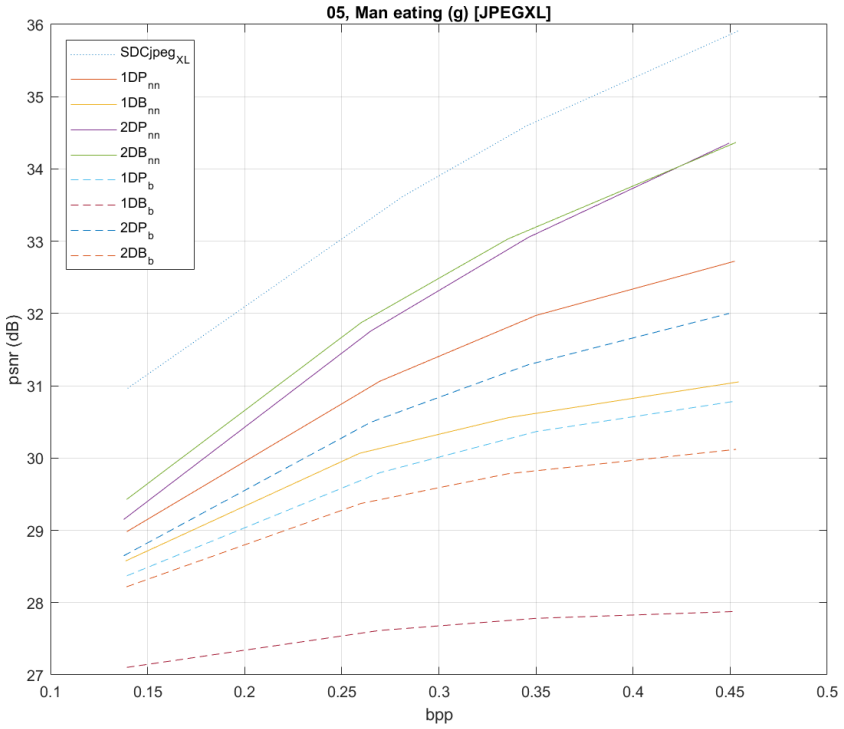


Figura 33 - Resultados debito-distorção imagem 5, JPEG XL

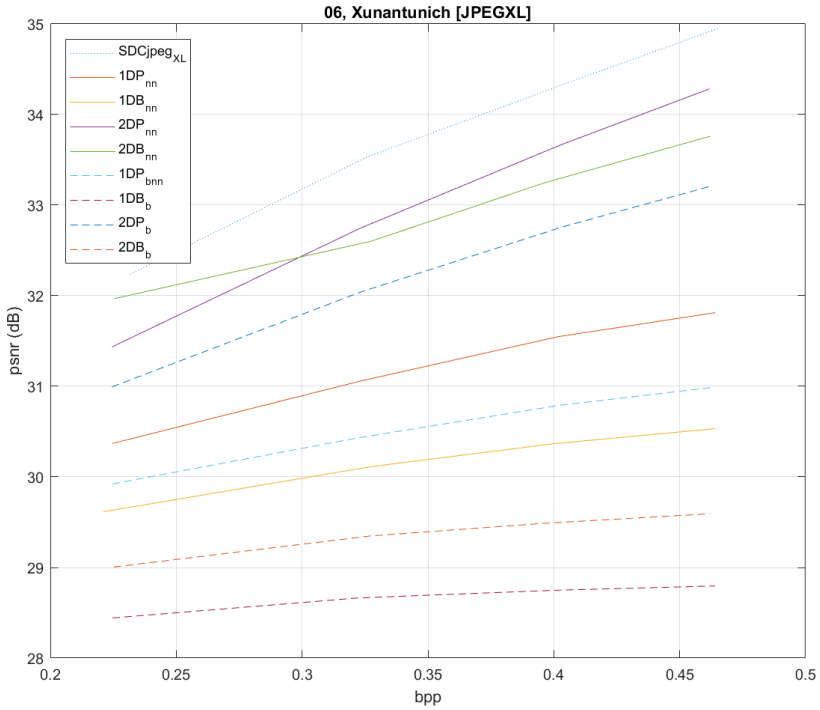


Figura 34 - Resultados debito-distorção imagem 6, JPEG XL

4.3.3. JPEG

Nesta seção são apresentados os resultados da reconstrução das imagens codificadas em JPEG, com uma descrição por pixel (1DPnn) e por bloco (2DPnn) e com duas descrições, por pixel (2DPnn) e por bloco (2DBnn). Os resultados comparam com a codificação JPEG da imagem (SDCJPEG), assim como com a imagem obtida a partir de uma descrição usando apenas interpolação bicúbica (1DPb, 1DBb, 2DPb, 2DBb).

Da Figura 35 à Figura 40, são apresentados os resultados débito distorção obtidos usando codificação JPEG. Considerando a referência SDCJPEG, o resultado obtido mantém o padrão obtido usando a codificação JPEXL, no entanto verifica-se que para débitos muito baixos, os métodos de restauro conseguem obter resultados de qualidade muito próximos da SDCJPEG ou mesmo com melhor qualidade. Estes resultados devem-se ao facto da distorção introduzida pelo *codec* JPEG ser elevada e ao mesmo tempo, a melhoria imposta pelo método de restauro conseguem minorar a distorção original da codificação.

Os resultados obtidos pelo método de restauro em comparação com o método de interpolação bicúbica, mantém o padrão comparativamente aos resultados obtidos com a codificação JPEGXL, assim como entre os vários métodos de codificação MDC utilizados.

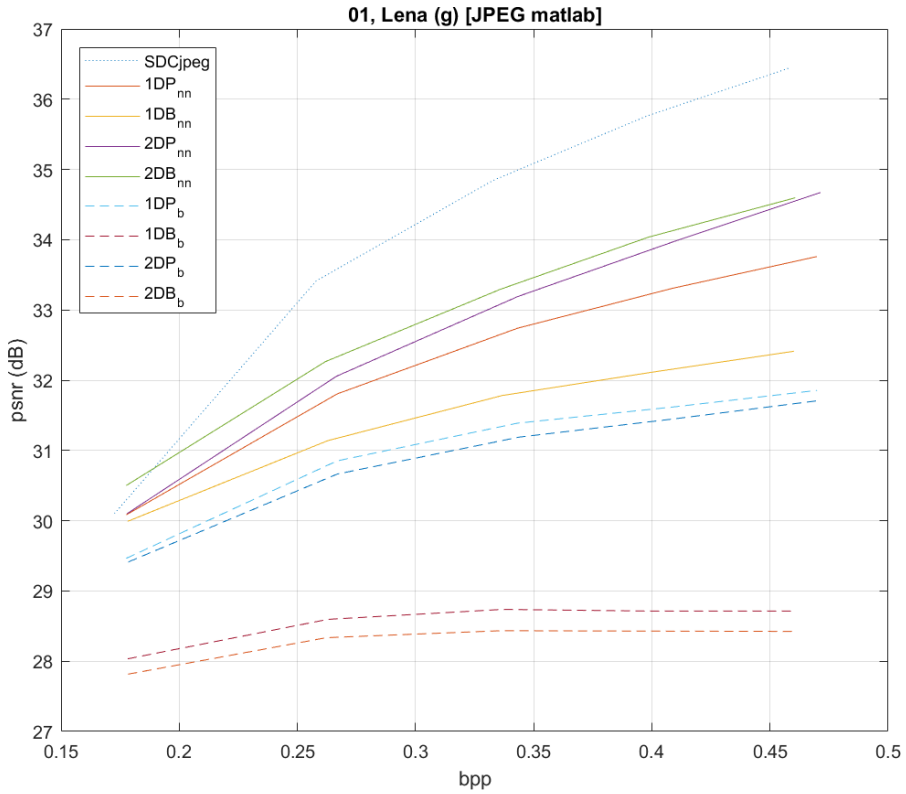


Figura 35 - Resultados debito-distorção imagem 1, JPEG

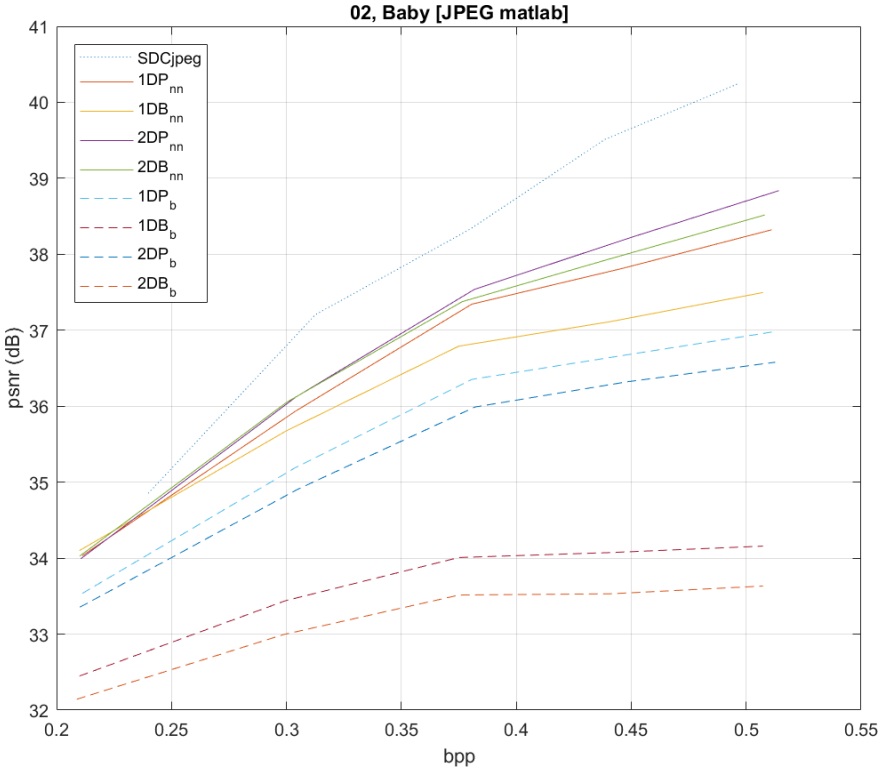


Figura 36 - Resultados debito-distorção imagem 2, JPEG

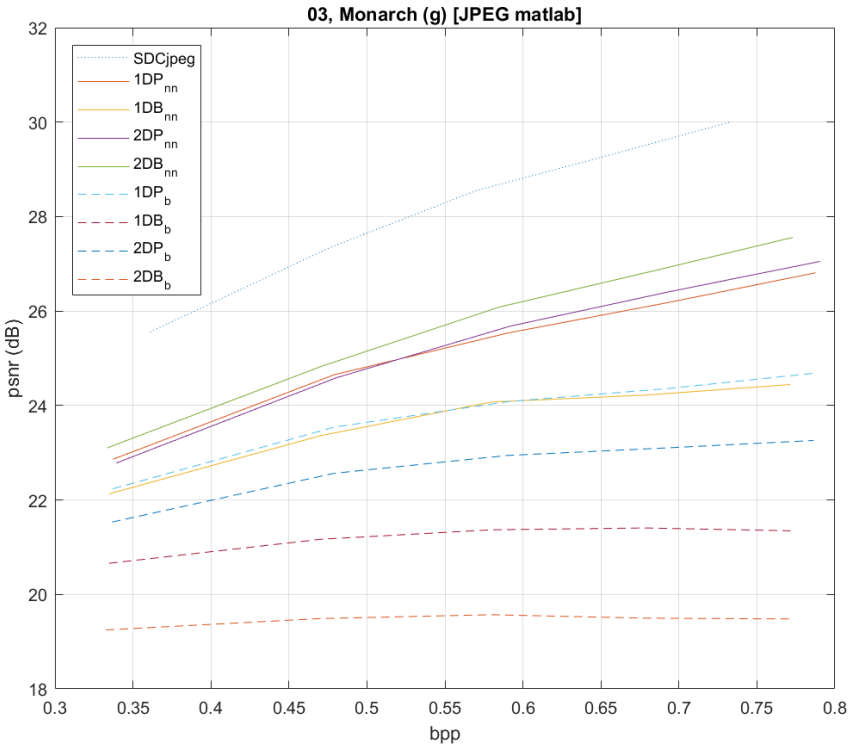


Figura 37 - Resultados debito-distorção imagem 3, JPEG

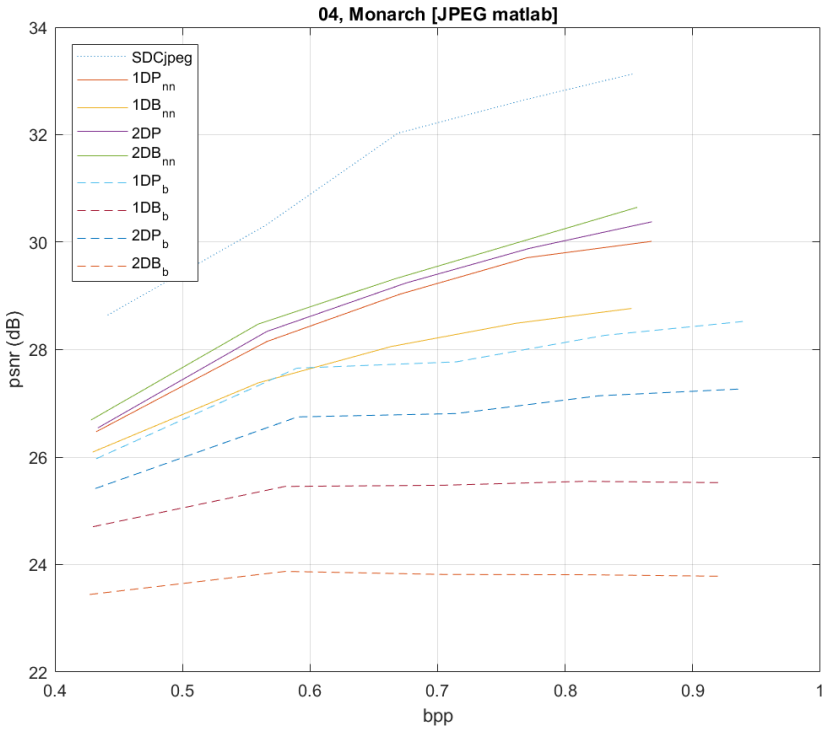


Figura 38 - Resultados debito-distorção imagem 4, JPEG

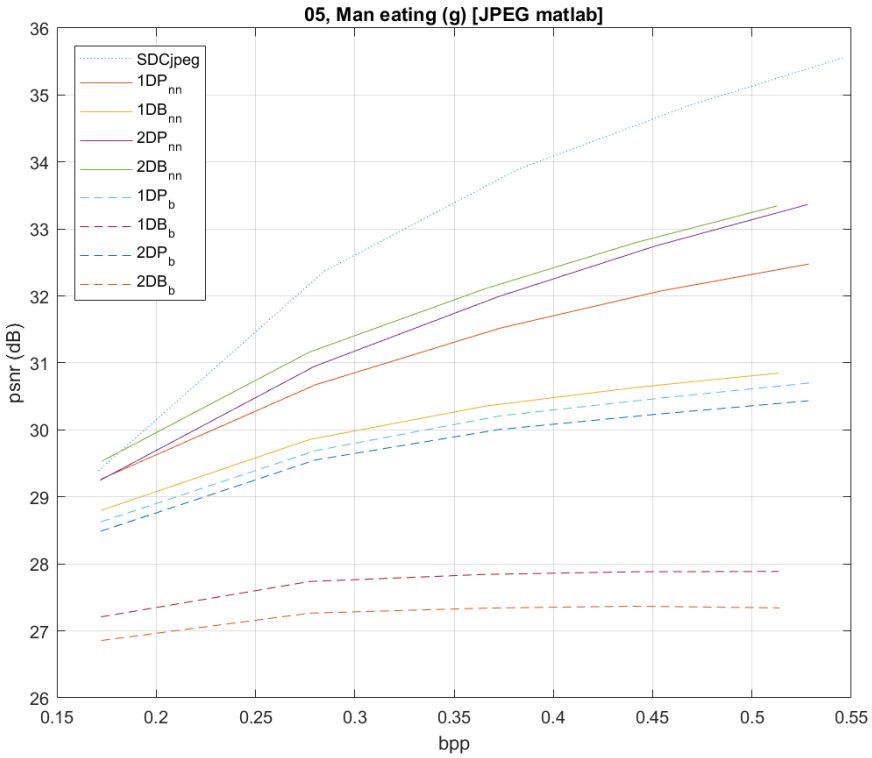


Figura 39 - Resultados debito-distorção imagem 5, JPEG

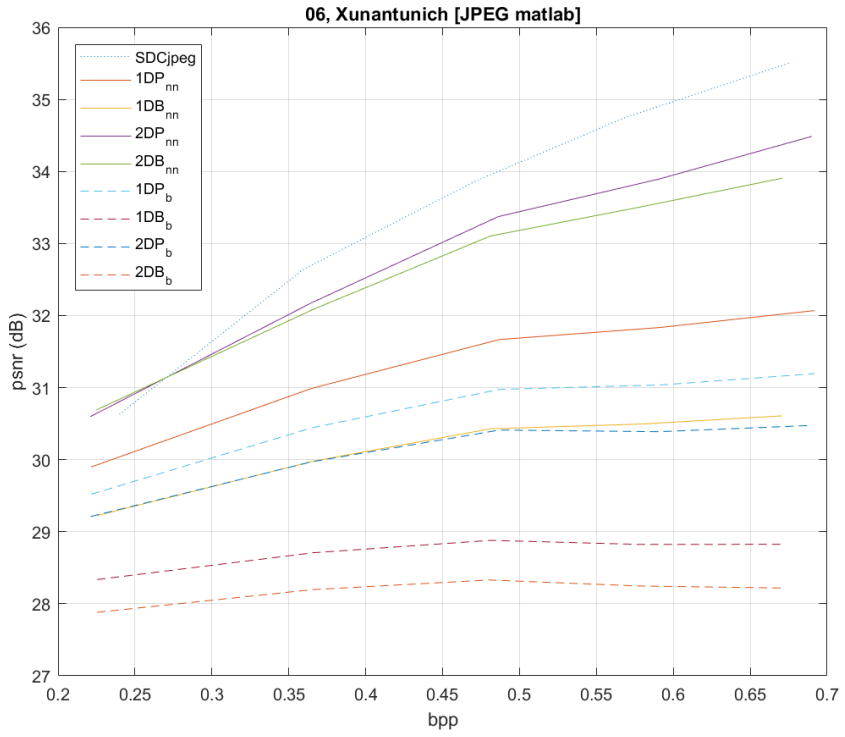


Figura 40 - Resultados debito-distorção imagem 6, JPEG

4.4. Descodificação MDC com IRCNN por Super-resolução

De acordo com a Figura 19, o descodificador recebe uma ou duas descrições e aplica o método IRCNN de super-resolução.

Adaptando o esquema geral da Figura 19, a Figura 41 apresenta o método usado para melhoria de imagem por super-resolução (SISR). Este, aplica uma interpolação bicúbica a cada descrição para restabelecer a resolução original e depois aplica a melhoria da imagem a partir da rede pré-treinada.

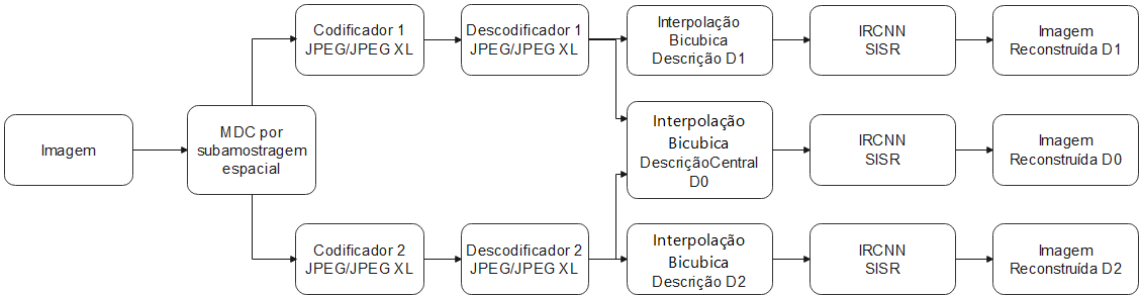


Figura 41 - Esquema geral do esquema de reconstrução com super-resolução (SIRS)

A Figura 28 - Set de imagens usadas nos testes de restauro de imagem. mostra as imagens usadas para testar o método de reconstrução com super-resolução. As imagens usadas são de cor e apresentam-se por ordem de 1 a 5, em que: imagem 1, 'Lena' (512x512); imagem 2, 'Baby', cor (512x512); imagem 3 'Monarch(g)' (256x256) , imagem 4, 'Monarch' (256x256); 'Man eating', cinza (1024x1024); imagem 6, 'Xunantunich', cor (1024x1024)

4.4.1 Resultados Débito - Distorção

- JPEG XL

Nesta secção são apresentados resultados para o método de reconstrução usando a técnica de super-resolução (SISR), usando codificação JPEGXL e usando apenas o método MDC com uma descrição por pixel (1DP) e duas descrições por pixel (2DP), uma vez que o método de uma descrição por bloco apresenta uma incompatibilidade com a rede usada levando a distorções de imagem incompatíveis. Os resultados comparam com a

codificação JPEG XL da imagem (SDCJPEGXL), assim como com a imagem obtida a partir de uma ou duas descrições usando interpolação bicúbica (1DPb, 2DPb).

Da Figura 42 - Resultados debito-distorção imagem 1, JPEG, SISRà Figura 47, são apresentados os resultados débito distorção obtidos usando a codificação JPEGXL. Considerando a referência $SDCJPEG_{XL}$, os resultados obtidos dos métodos de super-resolução são cerca de 3dB a 4 dB abaixo em média, o que é natural, devido tanto à falta de informação original existente no descodificador, assim como pela redundância introduzida pela codificação das descrições.

No entanto verifica-se que comparada com o método de interpolação bicúbica, os métodos de super-resolução funcionam de forma bastante interessante uma vez que se obtêm ganhos de qualidade de reconstrução da ordem dos 1 a 2 dB.

Ao comparar os resultados obtidos na reconstrução com os métodos de restauro em comparação com os métodos de super-resolução, verifica-se que o método de restauro funciona de forma mais efetiva na melhoria da imagem. Este resultado prende-se pelo facto de sendo uma rede pré-treinada, o tipo de distorção introduzida pelos métodos de restauro no treino estar mais próxima do método de MDC utilizado do que a distorção introduzida pelo treino de super-resolução. Recorde-se que este método MDC utilizado retira informação espacial original que necessita de ser reconstruída pela rede, o que é muito semelhante à distorção que se encontra em situações de restauro de imagem.

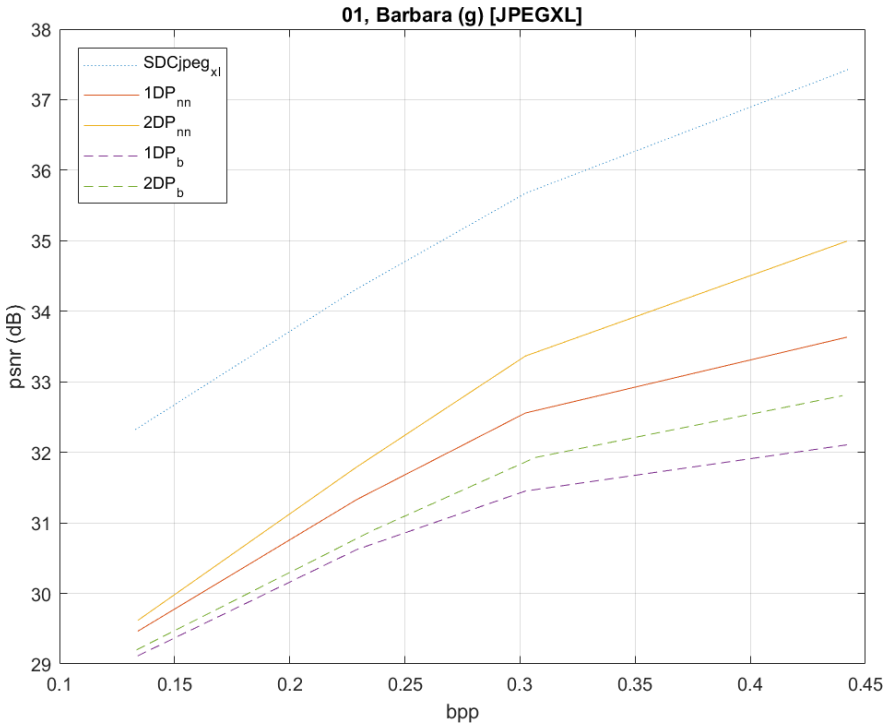


Figura 42 - Resultados debito-distorção imagem 1, JPEG, SISR

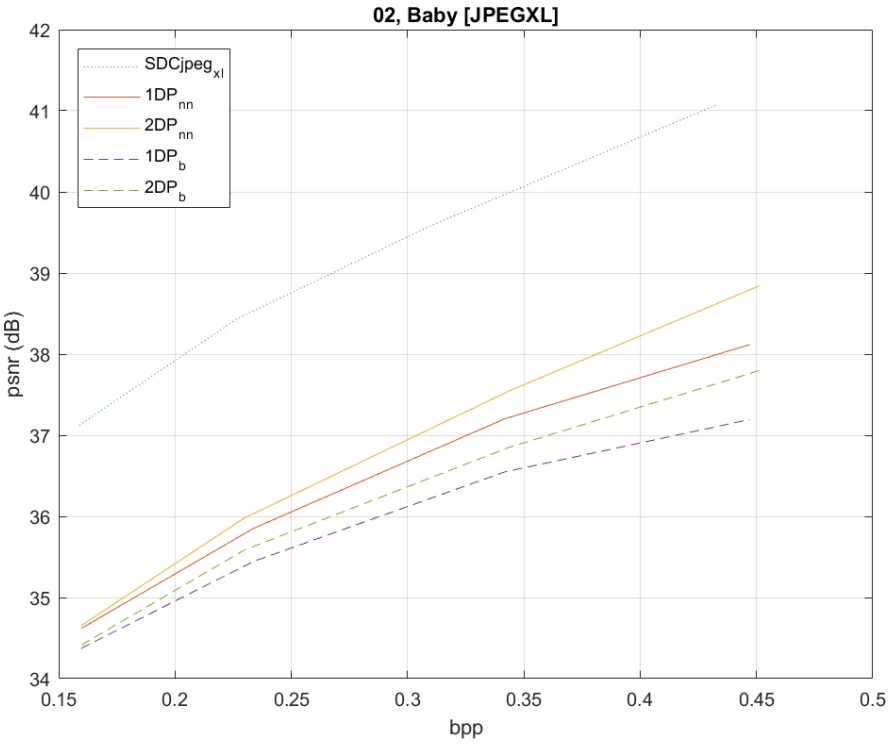


Figura 43 - Resultados debito-distorção imagem 2, JPEG, SISR

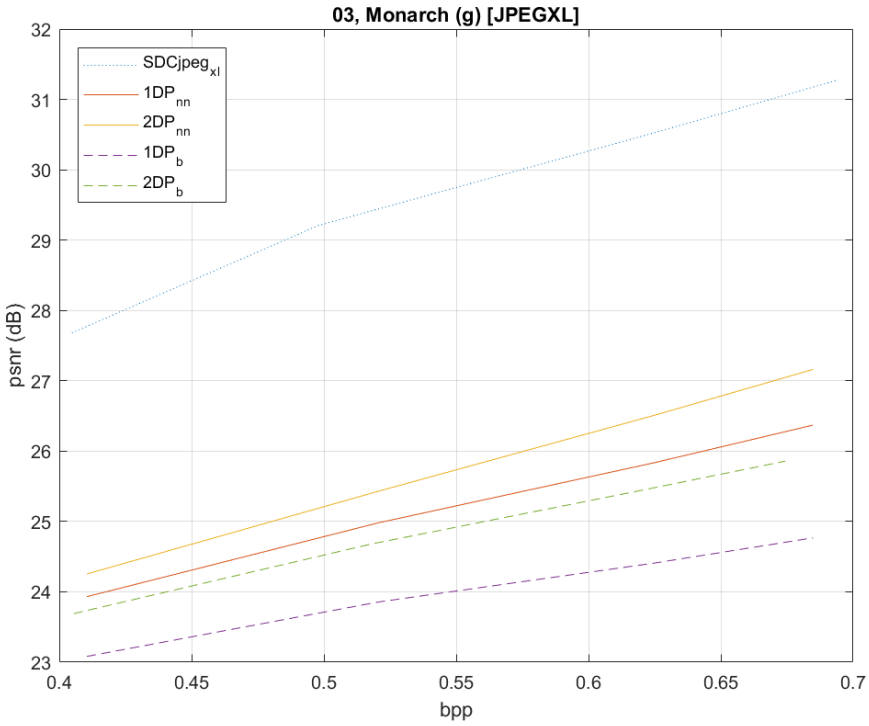


Figura 44 - Resultados debito-distorção imagem 3, JPEG, SISR

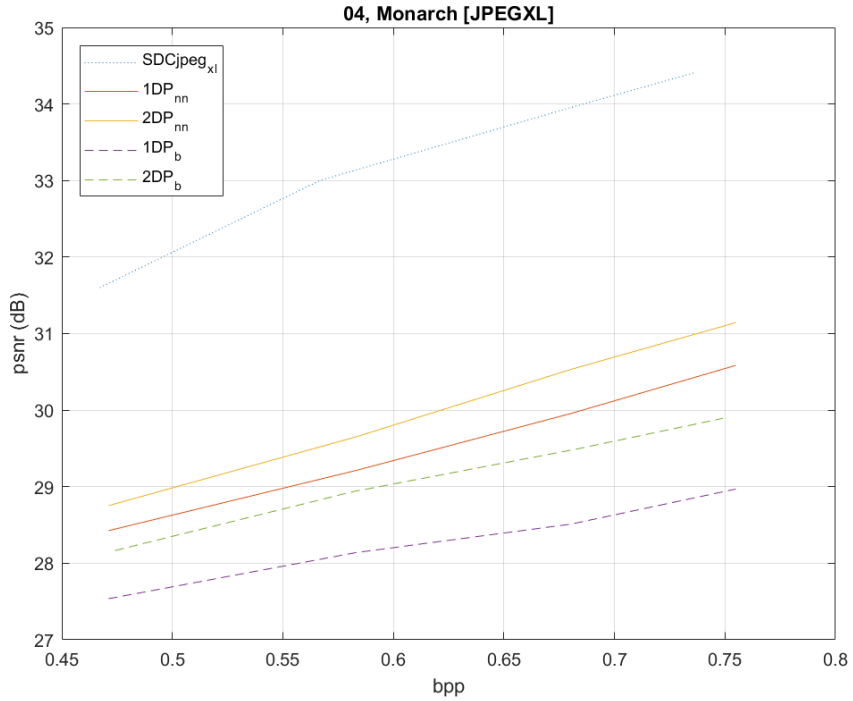


Figura 45 - Resultados debito-distorção imagem 4, JPEG, SISR

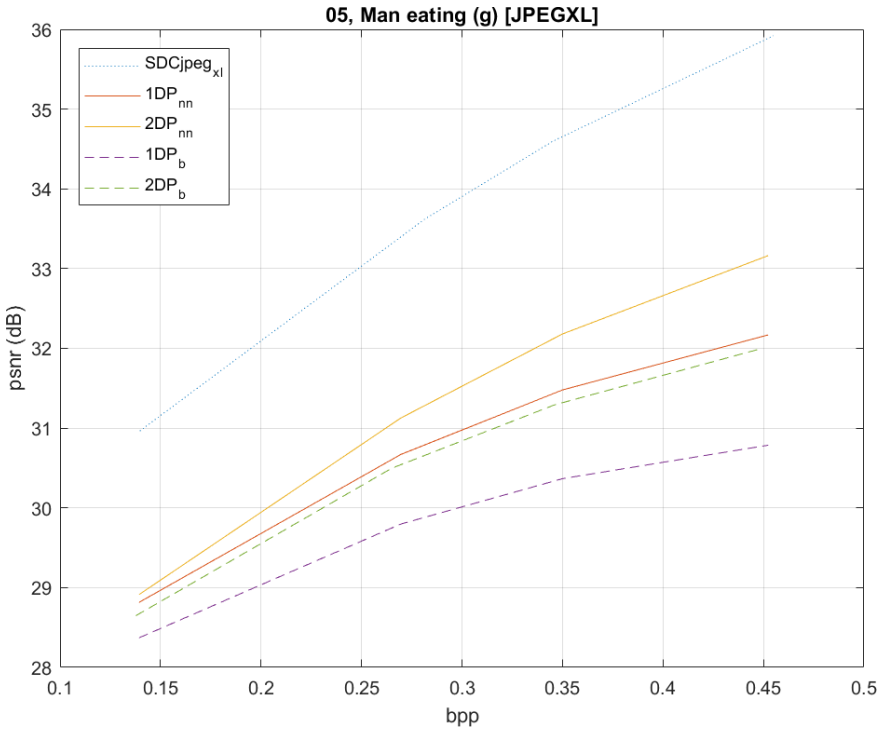


Figura 46 - Resultados debito-distorção imagem 5, JPEG, SISR

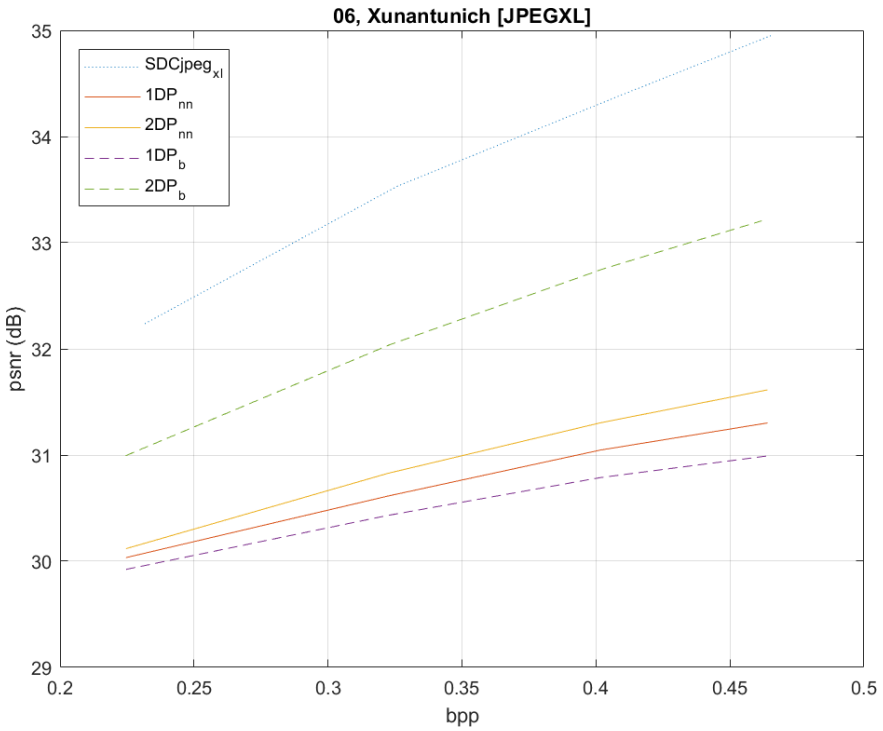


Figura 47 - Resultados debito-distorção imagem 6, JPEG, SISR

- **JPEG**

Nesta seção são apresentados os resultados da reconstrução das imagens por super-resolução codificadas em JPEG, com uma descrição por pixel (1DPnn) e com duas descrições por pixel (2DPnn). Os resultados comparam com a codificação JPEG da imagem (SDCJPEG), assim como com a imagem obtida a partir de uma descrição usando apenas interpolação bicúbica (1DPb, 2DPb).

Da Figura 48 à Figura 53, são apresentados os resultados débito distorção obtidos usando codificação JPEG. Considerando a referência SDCJPEG, o resultado obtido mantém o padrão obtido usando a codificação JPEGXL, no entanto verifica-se que para débitos muito baixos, os métodos de restauro conseguem obter resultados de qualidade muito próximos da SDCJPEG ou mesmo com melhor qualidade. Estes resultados devem-se ao facto da distorção introduzida pelo *codec* JPEG ser elevada e ao mesmo tempo, a melhoria imposta pelo método de super-resolução conseguem minorar a distorção original da codificação.

Os resultados obtidos pelo método de super-resolução em comparação com o método de interpolação bicúbica, mantém o padrão comparativamente aos resultados obtidos com a codificação JPEGXL, assim como entre os vários métodos de codificação MDC utilizados.

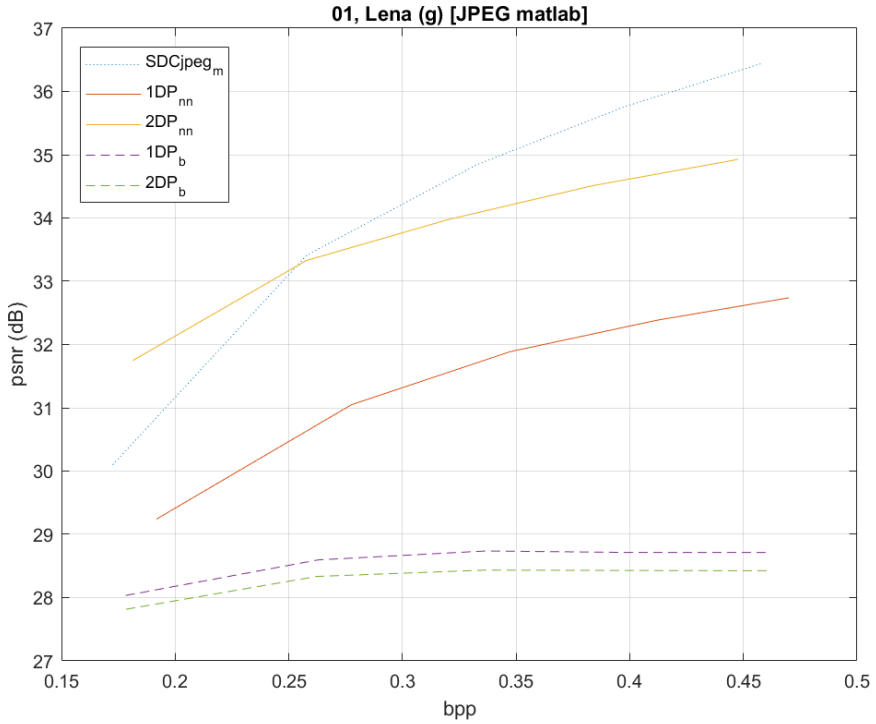


Figura 48- Resultados debito-distorção imagem 1, JPEG XL, SISR

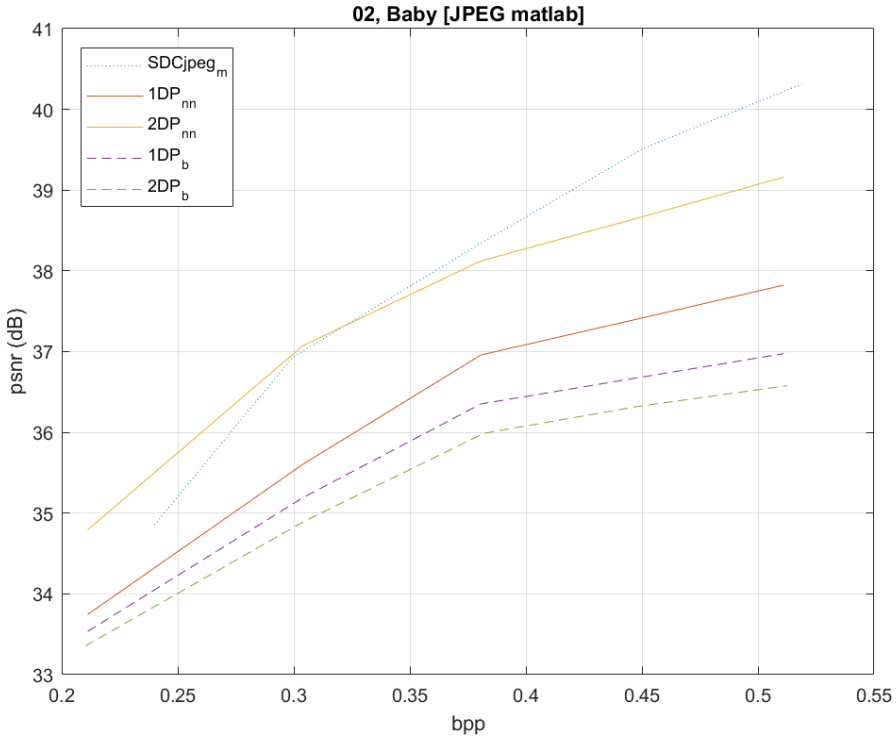


Figura 49 - Resultados debito-distorção imagem 2, JPEG XL, SISR

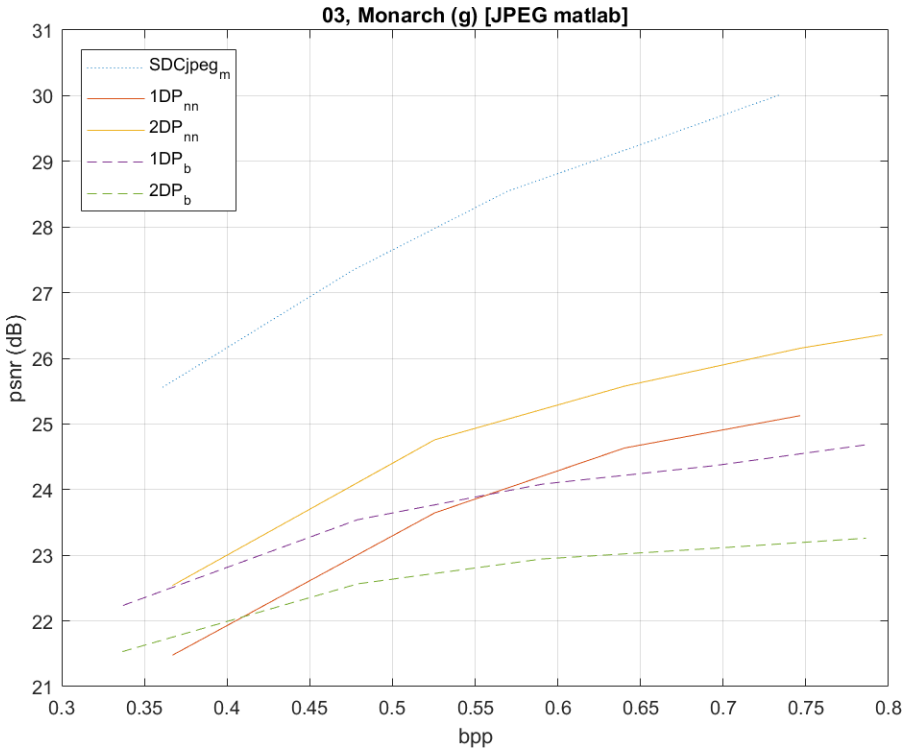


Figura 50 - Resultados debito-distorção imagem 3, JPEG XL, SISR

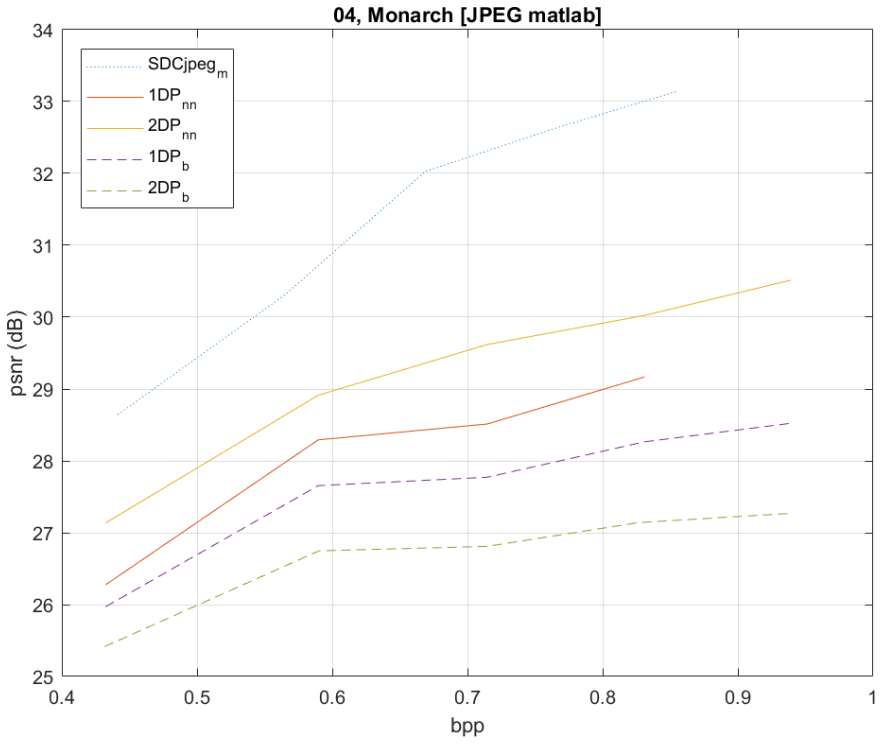


Figura 51 - Resultados debito-distorção imagem 4, JPEG XL, SISR

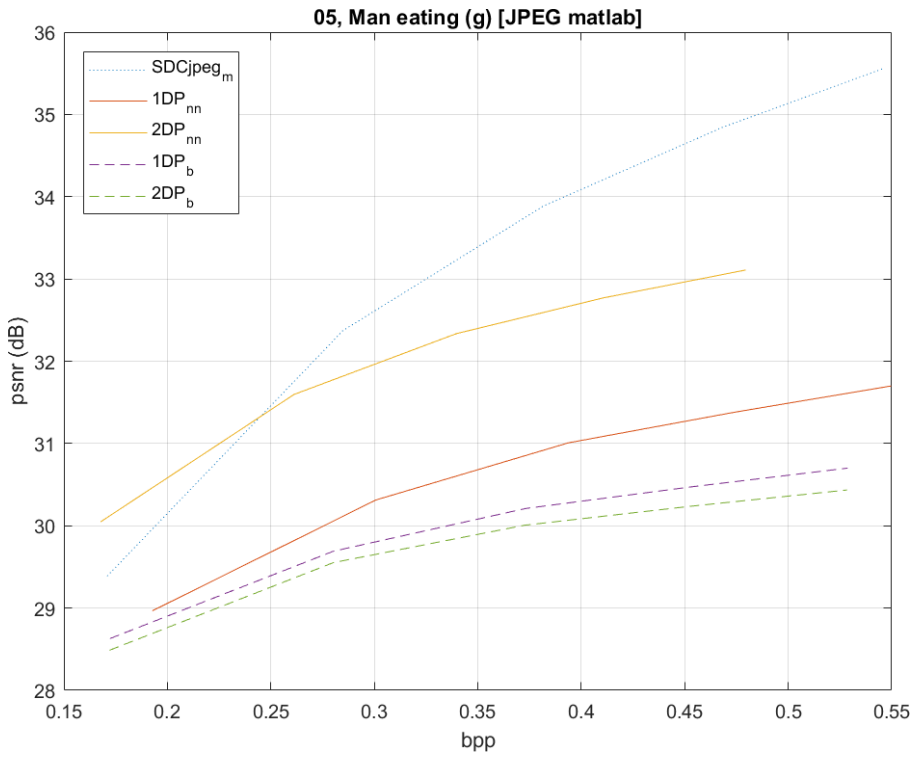


Figura 52 - Resultados debito-distorção imagem 5, JPEG XL, SISR

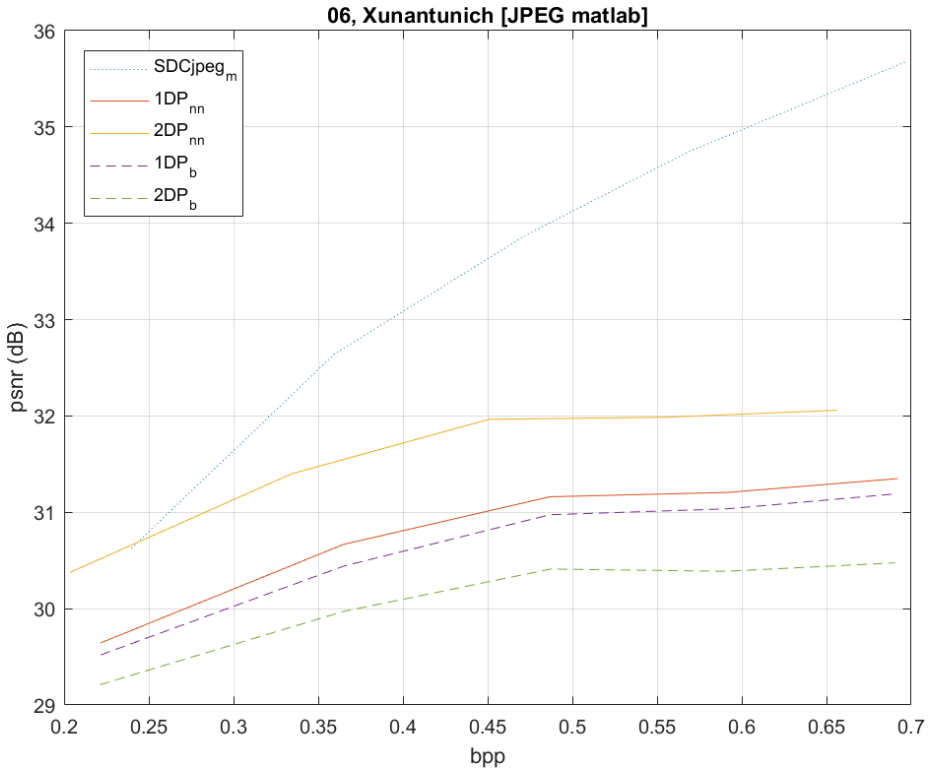


Figura 53 - Resultados debito-distorção imagem 6, JPEG XL, SISR

4.5. Conclusão

Este capítulo apresenta o estudo da utilização de métodos de reconstrução de imagem para codificação de imagem MDC onde apenas metade ou um quarto da informação original está disponível no descodificador.

São usadas rede CNN pré-treinadas para utilização de técnicas de reconstrução por restauro e por super-resolução. Dos testes efetuados, conclui-se que o treino original consegue corrigir a distorção introduzida pela codificação MDC melhorando substancialmente a qualidade das imagens reconstruídas, quando comparadas com os métodos clássicos de interpolação, nomeadamente a interpolação bicúbica.

Sendo este um estudo exploratório, conclui-se que treinando as redes para o tipo de distorção introduzida pela codificação MDC, os resultados obtidos podem ser melhorados. Paralelamente, podem ser exploradas outras técnicas MDC de modo a em conjunto com as CNN usadas na melhoria de imagem em pós-processamento possam produzir melhores resultados. Este estudo é apresentado no capítulo 5.

5. REDUÇÃO DE RUÍDO EM CODIFICAÇÃO DE IMAGEM COM MÚLTIPLAS DESCRIÇÕES

Este capítulo aborda a melhoria de imagens para MDC, usando um esquema MDC diferente do descrito no capítulo anterior. Neste caso o pós-processamento é baseado no método de redução de ruído (*denoising*) DnCNN [42]. A aplicação de DnCNN a imagens MDC, implica o treino e teste da rede CNN a este novo contexto. Assim, neste capítulo é apresentado o esquema MDC implementado, descrito o teste e treino da DnCNN adaptado ao esquema MDC utilizado e os resultados obtidos pela implementação no método de melhoria de imagem em aplicações MDC balanceadas e não balanceadas, isto é, com descrições codificadas com o mesmo débito ou com débitos diferentes respetivamente.

5.1. Esquema MDC

A Figura 54 apresenta do esquema geral da codificação MDC usada no contexto deste trabalho. Para cada imagem, são geradas duas descrições resultantes da subamostragem espacial, neste caso realizando uma subamostragem por linhas ou por colunas. Ou seja, do esquema MDC resultam duas imagens complementares, com metade da resolução horizontal ou vertical. Cada descrição pode ser descodificada de forma independente. Se existir apenas uma descrição disponível, é realizada uma interpolação bicúbica para a resolução original, seguida do método DnCNN para melhoria da qualidade da imagem. Se existirem as duas descrições, estas são combinadas de acordo com o esquema original e é aplicada o mesmo método para melhoria de imagem.

De notar que o esquema permite que as descrições sejam codificadas com a mesma qualidade (MDC balanceado), ou então com qualidades diferentes (não balanceado). Um aspeto a considerar neste esquema, é perceber qual a influencia da DnCNN na melhoria da qualidade da imagem resultante do método não balanceado, uma vez que apesar de existir uma descrição ter pior qualidade, logo mais distorção, pode fornecer informação importante à rede CNN, de modo a melhor tirar partido do treino nestas condições.

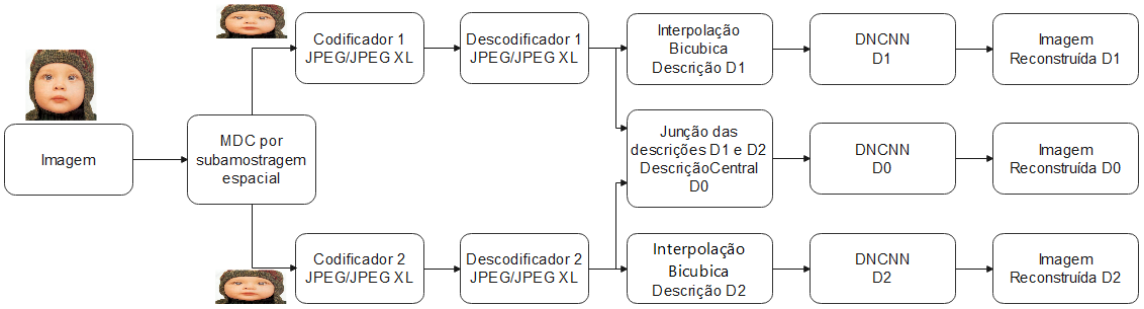


Figura 54 - Esquema geral do esquema de reconstrução com DnCNN

A Figura 55, apresenta o esquema MDC utilizado. Neste caso, por exemplo, as linhas de pixel pares constituem a descrição D1, e as linhas de pixel ímpares constituem as linhas ímpares, resultante em dias imagens com metade da resolução vertical. Poder-se-ia usar um critério idêntico para as colunas. Cada um dos codificadores é independente, pelo que os parâmetros de codificação de cada descrição podem ser iguais (MDC balanceado) ou diferentes (MDC não balanceado).

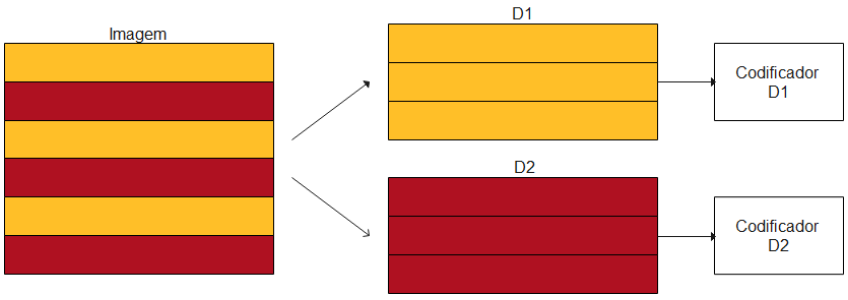


Figura 55 – Esquema de codificação MDC.

A Figura 56, apresenta o esquema de descodificação MDC usado. Quando existe apenas uma descrição, a resolução da imagem original é realizada a partir de interpolação bicúbica. De seguida é aplicada a DnCNN para melhoria da qualidade de imagem. Quando existem as duas descrições, os pixéis de cada imagem são colocados nas posições originais. Sendo cada descrição codificada de forma independente, espera-se que a distorção introduzida pelo codificador, assim como pelo método MDC em si, seja minimizada com o esquema DnCNN.

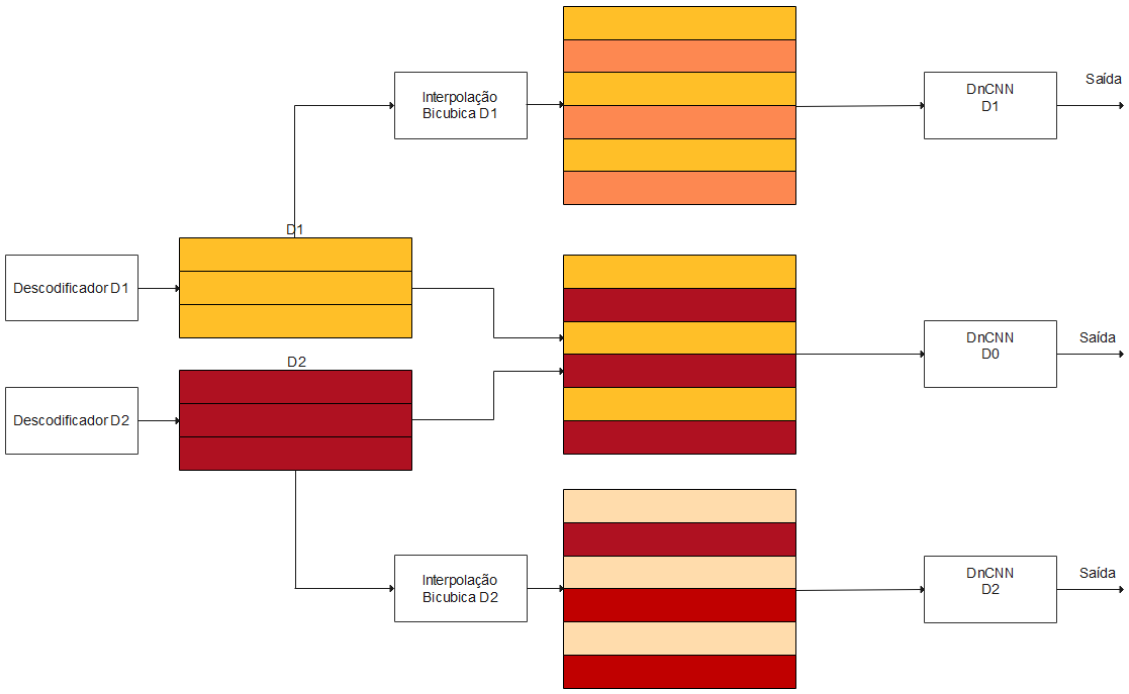


Figura 56 – Esquema de Descodificação MDC com melhoria da imagem com DnCNN.

5.2. Melhoria de qualidade de imagem usando CNN para redução de ruido

No contexto deste trabalho, é usada a rede CNN para redução de ruído proposta em [42]. Este apresenta uma rede *Feed-Forward* de redução de ruído, capaz de remover ruído gaussiano sem saber à partida o seu nível. Utiliza uma estratégia de aprendizagem de resíduo resultante entre a imagem ruidosa e a imagem correspondente sem ruído. Assim, a DnCNN consegue remover de forma implícita o ruído existente numa imagem, depois de treinada para extrair o nível de ruído existente quando comparada com uma imagem limpa. Deste modo, esta abordagem permite que o método possa ser aplicado em diferentes casos de melhoria de imagem, desde a redução de ruído propriamente dito, mas também a distorção introduzida num processo de interpolação em aplicações de super-resolução, de uma codificação JPEG ou na reconstrução de imagens resultantes de MDC, cujo estudo é abordado neste capítulo.

A Figura 57 apresenta o esquema da rede DnCNN proposta em [42]. Esta permite extrair a imagem de resíduo a partir de uma imagem ruidosa, permitindo a partir daí extrair a imagem com qualidade melhorada.

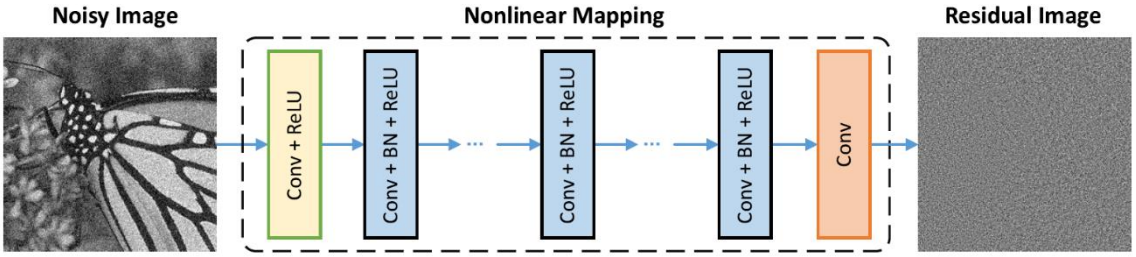


Figura 57-Esquema da Rede DnCNN. [42]

A entrada da DnCNN é a imagem ruidosa, que pode ser expressa por

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{v}$$

Onde $\boldsymbol{x}$ é a imagem sem ruido e $\boldsymbol{v}$ é o ruido existente.

A rede DnCNN pretende treinar o mapeamento de residuo $\mathcal{R}(y) \approx v$ por forma a que

$$\boldsymbol{x} = \boldsymbol{y} - \boldsymbol{\mathcal{R}}(\boldsymbol{y})$$

O sistema de treino funciona com a função de custo baseada na média do erro quadrático entre a imagem residual desejada e a entrada ruidosa, ou seja, a função de custo é dada por

$$\boldsymbol{l}(\boldsymbol{Q}) = \frac{\boldsymbol{1}}{\boldsymbol{2N}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \|\boldsymbol{\mathcal{R}}(\boldsymbol{y_i}); \boldsymbol{Q}\ ) - (\boldsymbol{y_i} - \ \boldsymbol{x_i})\|_{\boldsymbol{F}}^{\boldsymbol{2}}$$

onde Q é o conjunto de parâmetros da rede a determinar para o objetivo.

O conjunto $\{(y_i - x_i)\}_{i=1}^{N}$ representa o conjunto dos N pares imagens (*patches*) de treino ruidosas-limpas.

A arquitetura desta rede neuronal está descrita na Figura 57 com as três diferentes secções a cores diferentes. A primeira secção (cor amarela) é composta por uma camada convolucional de dimensões 3_3_c_64 em que c vai definir se estamos a trabalhar com imagens a cores ou a preto e branco, que no nosso caso é a preto e branco (c=1) e uma camada de ativação ReLU. Na segunda secção (azul) temos 15 vezes: uma camada convolucionais + *batch normalization* + ReLU. Por fim, na terceira (cor de laranja) temos apenas uma camada convolucional com c filtros de dimensões 3_3_64 para reconstruir a saída.

Em suma, este modelo tem duas principais características: a formulação de aprendizagem residual é usada para aprender R(y) (a diferença). A incorporação de *Batch Normalization* para acelerar o treino e melhorar os resultados da rede de redução de ruído.

5.3. Treino DnCNN em MDC

- ***Dataset***

Depois da rede estar aplicada para o treino é necessário definir o *dataset* de imagens de treino. O *dataset* é constituído por 400 imagens que é considerado reduzido para este processo de treino. Assim, o habitual consiste em fazer o *data_augmentation* que vai usar essas imagens e transformar uma imagem em oito, obtendo assim, um total de 3200 imagens. Este processo consiste em rodar as imagens a ângulos múltiplos de 90º e inverter. A Figura 58 apresenta uma amostra de 40 imagens do dataset de treino cada uma dessas quatro imagens.

Figura 58 - Amostra do dataset de treino.

- **Pré CNN**

O processo de gerar os dados, vão ser todas as imagens de treino correspondendo à sua referência (*labels*). Como queremos treinar as imagens com o processo de codificação associado, o processo de *data augmentation* é aplicado tanto aos *labels*, assim como o *dataset* codificado de acordo com os casos de melhoria que se pretendem treinar e processado pela rede. Dependendo do tipo de codificação, quer seja uma descrição ou duas, balanceado e não balanceado, são gerados os diferentes tipos de input. O treino foi concebido definindo a forma de como se ia estudar as imagens em que:

- **Uma descrição**

No processo de uma descrição, o que foi feito tal como nos outros, agrupam-se imagens em blocos de 2x2 pixéis, retiram-se 2 pixéis dos 4 de cada um desses blocos e desenvolve-se a imagem a partir daí, como está descrito na secção 5.1. Depois de codificada e descodificada, para tornar a imagem com a sua resolução original é realizada um '*resize'* usando a interpolação bicúbica.

- **Balanceado**

Este processo consiste em usar duas descrições descodificadas e combinando-as entre si de modo a obter uma imagem de uma determinada qualidade, como descrito na secção 5.1.

- **Não Balanceado**

Neste caso usam-se diferentes qualidades de codificação das descrições, isto é, uma descrição com mais qualidade que a outra.

- **Não Balanceado específico**

A diferença entre este modo e o não balanceado consiste em usar a rede treinada para cada qualidade de codificação que vamos usar, isto é, usando um fator 50 de codificação numa descrição e usando o fator 10 até 40 na outra descrição. A rede é treinada em todas estas diferentes combinações.

Note-se que o treino neste caso foi feito usando todas as diferentes qualidades de codificação, do não balanceado, em que foi feita com um treino para uma das qualidades. Isto para se poder inferir a diferença da qualidade de imagem quando a rede é treinada para diferentes qualidades de cada descrição.

A implementação base para o presente trabalho está em [43] implementada em MATLAB.

5.4. Testes DNN com MDC

No processo de teste vai-se fazer o respetivo processamento que foi feito no treino (patches_generation), por exemplo, usando o treino feito para duas descrições não balanceadas (i.e. diferentes fatores de qualidade (QF) entre as descrições), o input na nossa rede vai ser igualmente o label com duas descrições com cada descrição com o seu QF associado. Vai ser feito uma avaliação de desemprenho consoante algumas variáveis, sendo possível, permitir escolher os valores mais favoráveis.

5.4.1. Avaliação de desempenho

Tendo em conta as necessidades de redução de ruído para os diferentes tipos, foram feitos testes para verificar se com determinados ajustes poderiam se obter melhores resultados. Em variáveis como o LR (*learning rate*) e o número de épocas. O dataset de teste (set 12) é apresentado na Figura 59.

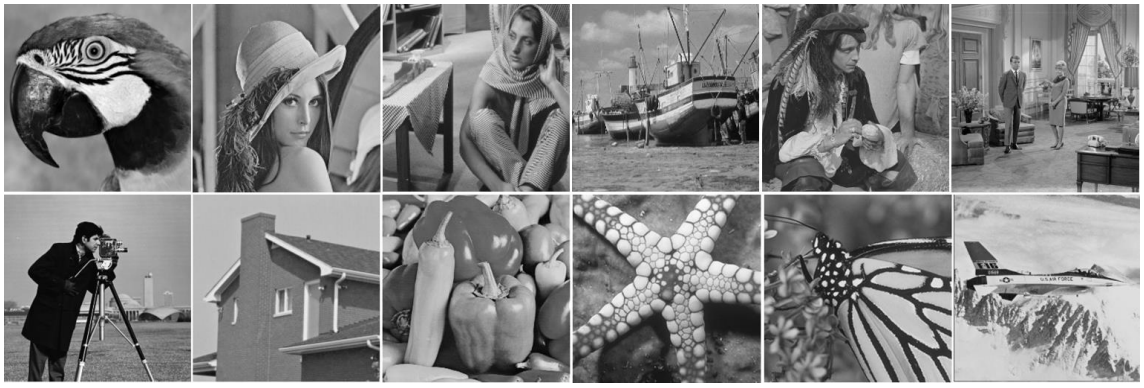


Figura 59 – Set12, imagens usadas nos testes de avaliação de desempenho.

- *Learning Rate*

Para o LR como predefinição já estava definido como 0.001 nas primeiras 20 épocas e 0.0001 para as outras. Sendo que o treino só foi realizado para 10 épocas, o LR de predefinição mantém-se sempre 0.001, foram feitos mais 2 treinos com:

- Lr = logspace(-2.5,-3.5,10)
- Lr = logspace(-3,-4,10)

'Logspace(a,b,c)' significa que são criados c número de pontos, do ponto a ao ponto b, isto espaçado logaritmicamente.

Os resultados obtidos estão representados na Figura 60. Com diferentes valores de LR, verifica-se que a média dos valores de PSNR vs. Bit por pixel (bpp), não é significativa.

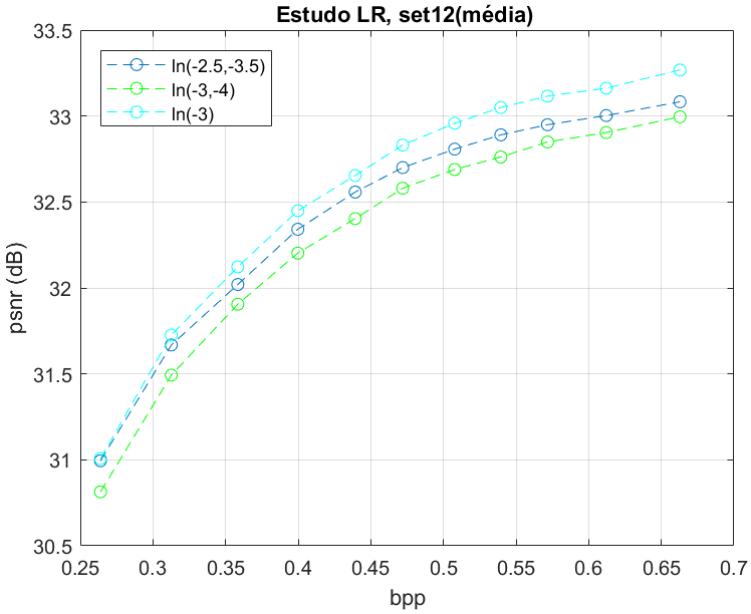
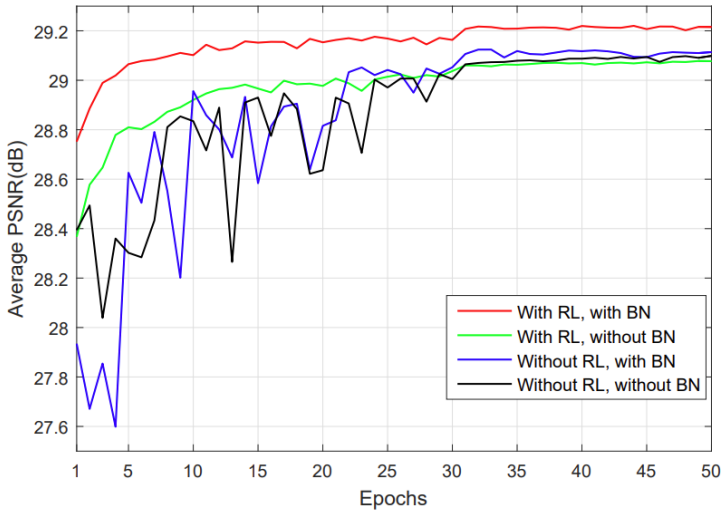


Figura 60 – Avaliação de desempenho de LR.

- Épocas

A Figura 61 apresenta um estudo realizado em [43] sobre o valor médio de PSNR obtido em função do número de épocas de treino. Pelos resultados é possível observar que a diferença do desempenho entre a época 10 e a época 50 é de apenas 0.1dB. Note-se que o estudo apresentado na Figura 61, foi feito sobre haver a camada ReLU e BN. No presente trabalho foram também usadas as duas camadas.



(b) Adam

Figura 61 – Estudo apresentado por épocas [43].

A Figura 62 apresenta o estudo feito por época treinando para uma descrição, em que apenas a época 18 apresenta algumas melhorias. No entanto na época 22 o desempenho volta ao regular, correspondendo ao desempenho da época 10.

Tendo a informação de ambas figuras decidiu-se usar 10 épocas no treino, em que os resultados não se alteram muito de um número mais elevado, para além de ser mais eficiente pelo tempo.

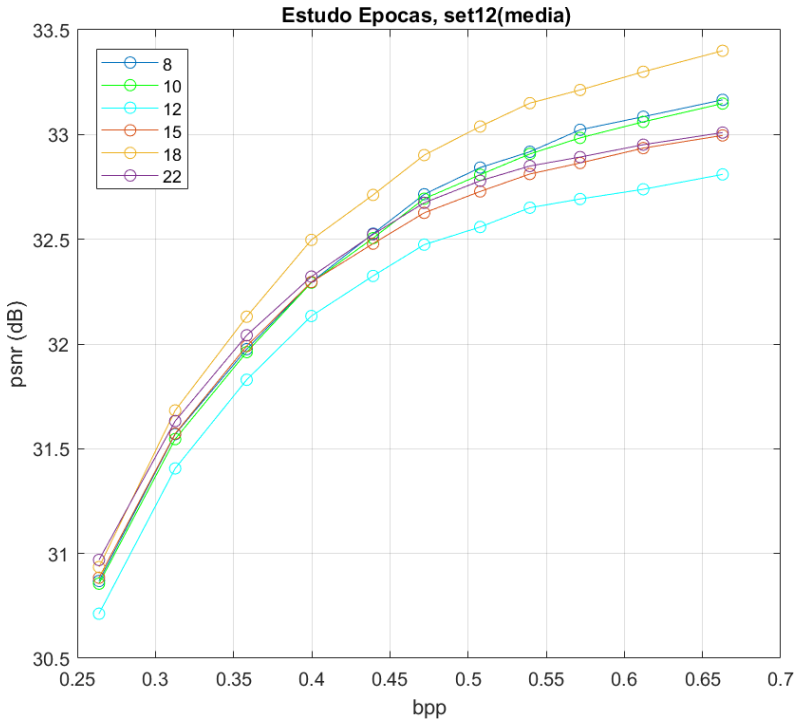


Figura 62 - Estudo por épocas na rede treinada.

- *Quality Factor* (QF)

A rede foi treinada com diferentes fatores que qualidade, para depois ser usada para qualquer parâmetro de qualidade às imagens de teste. O presente estudo pretende avaliar como o QF escolhido no treino, influencia os resultados obtidos no teste.

Em relação aos resultados da Figura 63 temos um compromisso entre ter melhores resultados quando se tem menos informação, treinando para menores QF. Treinando para um QF mais elevado vamos ter melhores resultados com mais informação comprometendo assim os resultados. Dos resultados obtidos, é espectável que com o treino realizado com QF de 50, correspondendo a metade da escala, este se adapta melhor a todas as situações, perdendo algum desempenho mais significativo em débitos mais elevados, correspondendo a imagens de melhor qualidade.

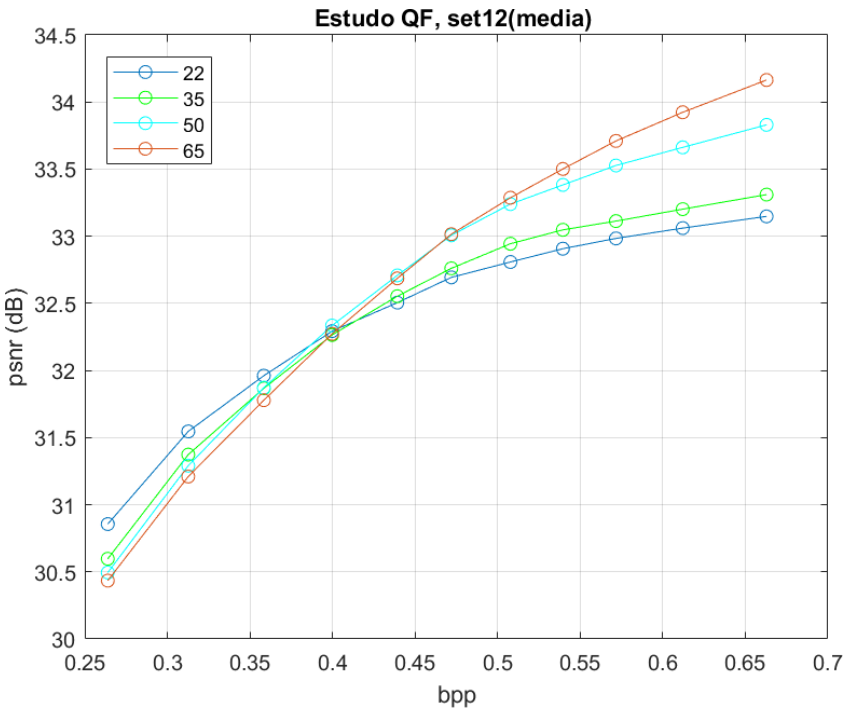


Figura 63 – Estudo por QF na rede treinada.

5.4.2. Estudo por imagem com Duas Descrições

Neste estudo aplicamos resultados a especificas imagens usando os diferentes modos que foram treinados, recorrendo às imagens da Figura 64, que contêm diferentes resoluções espaciais. Da esquerda para a direita temos: imagem 1, Lena (512×512); imagem 2,

Monarch (256×256); imagem 3, Man eating (1024×1024); imagem 4, Barbara (512×512); imagem 5, Photographer (256×256).

Figura 64 - Imagens usadas para os testes das redes de 1 e 2 descrições.

São apresentados os resultados obtidos na descodificação das imagens com duas descrições (Figura 65 a Figura 69). Para comparação dos resultados a medida de distorção utilizada é o PSNR (*Peak Signal to Noise Ratio*), definido como

$$PSNR = 10log_{10}\frac{255^2}{MSE}\ dB$$

onde

$$MSE = \frac{1}{M\times N}\sum_{i=1}^{N}\sum_{j=1}^{M}\left[I(i,j) - I_{Ref}(i,j)\right]^2$$

é o erro médio quadrático entre a imagem descodificada e a imagem de referência.

O número de bit por pixel (bpp) considerado é a relação do débito resultante da codificação de duas descrições, usando codificação balanceada e não balanceada, relativamente à resolução original da imagem, ou seja,

$$bpp = \frac{R_{D1}+R_{D2}}{M\times N}\ \text{bit/pixel}$$

Os resultados da Figura 65 à Figura 69 comparam a codificação MDC balanceada (Bal), MDC não balanceada (UnBal) com treino balanceado, MDC não balanceada com treino não balanceado (UnBalEx). São apresentados também os resultados da descodificação das descrições balanceadas (Bal_{antes}) e não balanceadas ($Unbal_{antes}$), obtidos antes do método de melhoria de imagem. Estes métodos são comparados com JPEG (SDC_{jpeg}) e com JPEG usando o método de melhoria de imagem ($JPEG_{nn}$).

Em relação aos resultados obtidos, constata-se que o método de image DnCNN consegue melhorar significativamente a imagem JPG SDC significativamente até 1dB. Este resultado permite fazer uma comparação justa com os métodos MDC utilizados, que pela sua natureza impõem redundância, piorando o desempenho em termos de relação débito-distorção.

Relativamente aos métodos MDC, verifica-se que valores de *bpp* mais baixos, os métodos MDC não balanceado atinge maior qualidade que o método balanceado. Este facto deve-se a que a descrição com melhor qualidade é usada pelo método de melhoria de imagem para melhorar as regiões da imagem correspondentes à descrição com pior qualidade. Verifica-se assim que o método não balanceado pode ser um método interessante para melhorar a relação débito-distorção geral. Por outro lado, também se verifica que quando a rede é treinada em específico para os métodos não balanceados, que a qualidade de descodificação aumenta significativamente em débitos mais elevados.

Também se justifica uma vez que com maior débito, estão disponíveis mais detalhes da imagem para melhorar o treino e consequentemente, o desempenho final. Nota-se também que, em débitos mais baixos o esquema MDC não balanceado apresenta um desempenho melhor que JPEG SDC.

O facto de o método não balanceado não atingir débitos mais baixos do que os outros métodos, deve-se ao facto de tendo os níveis de QF 8 e 25 das duas descrições, o esquema MDC balanceado tem ambas descrições no nível 10, alcançando valores mais baixos de bpp.

No caso de MDC não balanceado, os ganhos do método DnCNN, este vão desde cerca de 3-4dB em débitos baixos (0.4bpp) e 1-2dB para débitos mais elevados (0.8bpp). Considerando MDC balanceado, os ganhos são entre 1dB para débitos baixos (0.4bpp) e 1-3.5 dB para débitos altos (1.0 bpp).

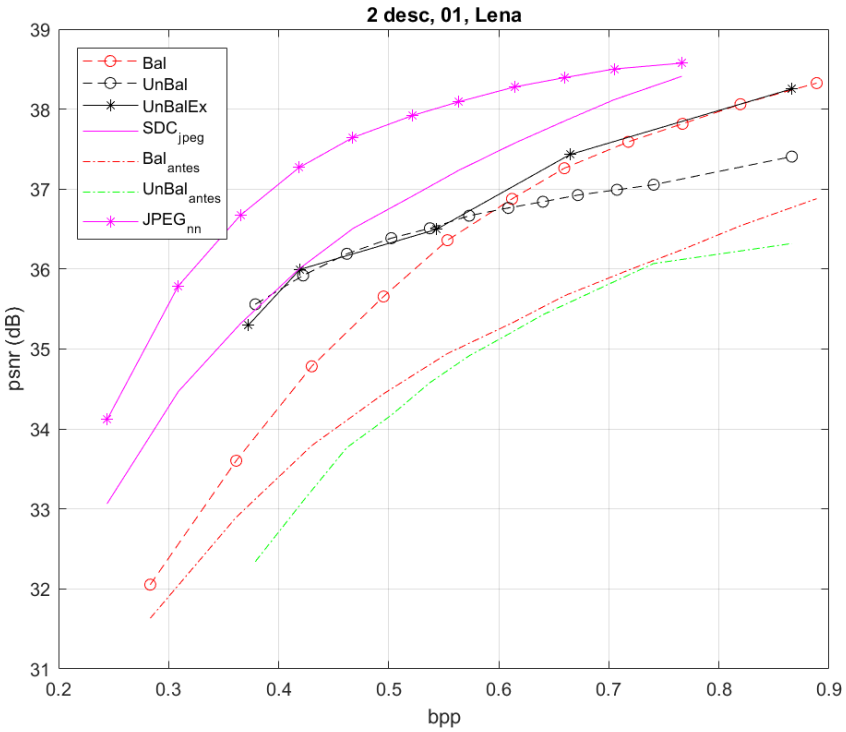


Figura 65 - Resutados Dnn em MDC, imagem 1

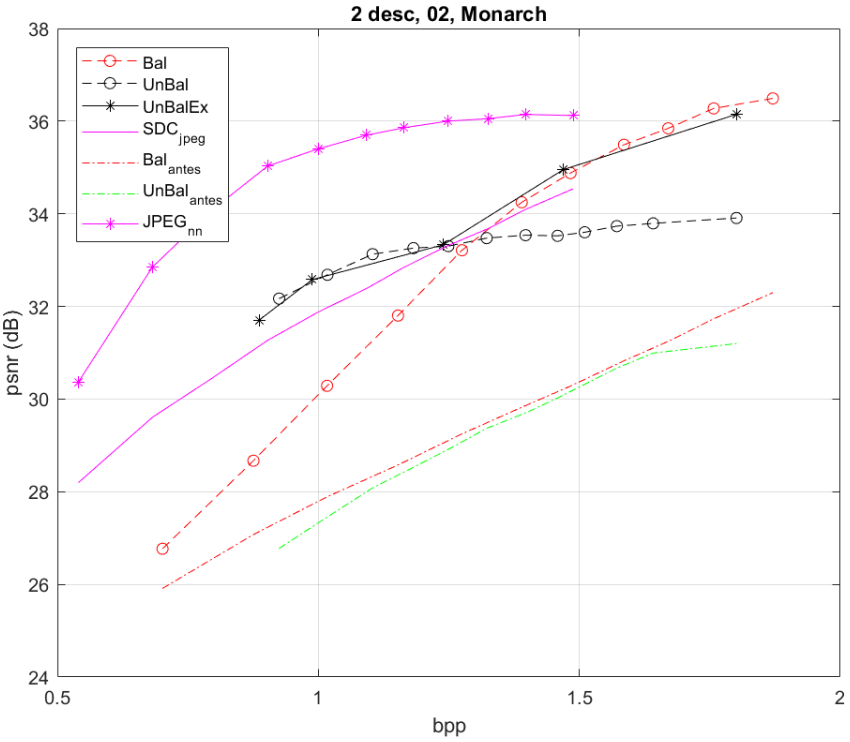


Figura 66 - Resutados Dnn em MDC, imagem 2

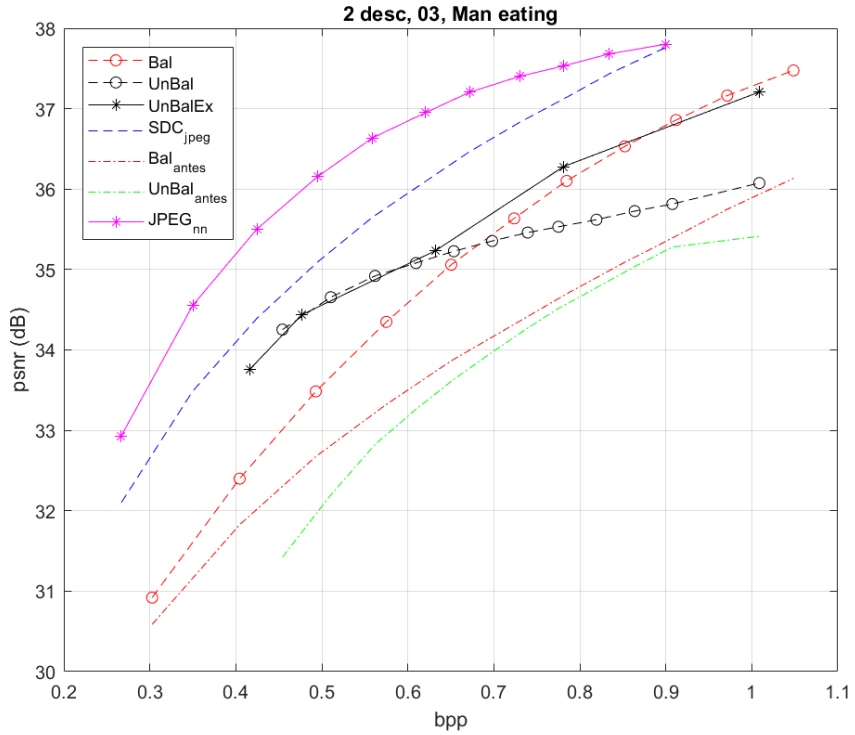


Figura 67 - Resutados Dnn em MDC, imagem 3

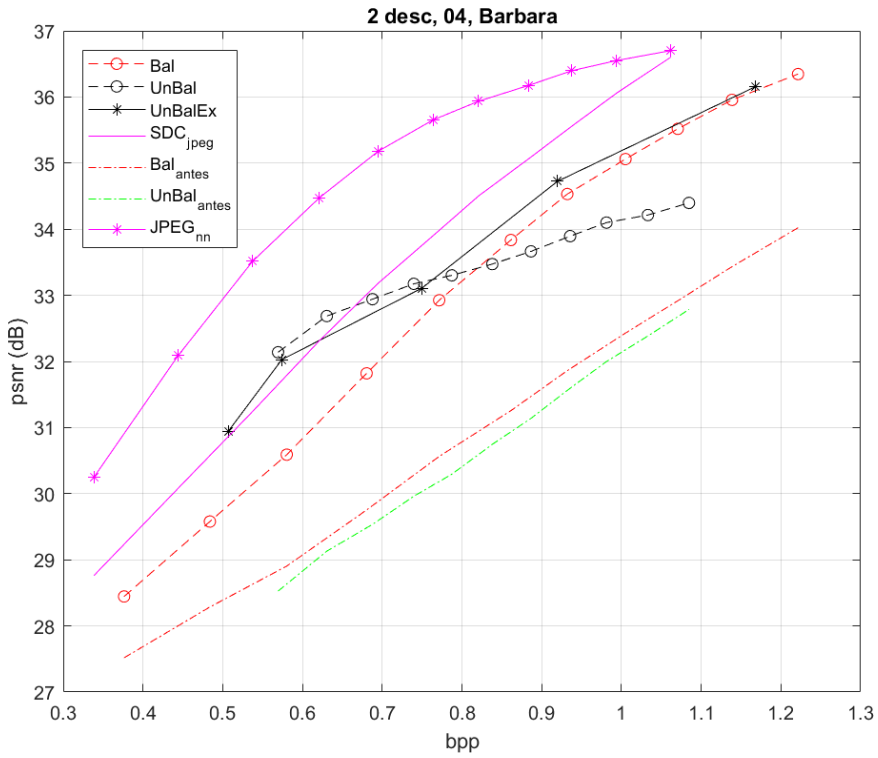


Figura 68 - Resutados Dnn em MDC, imagem 4

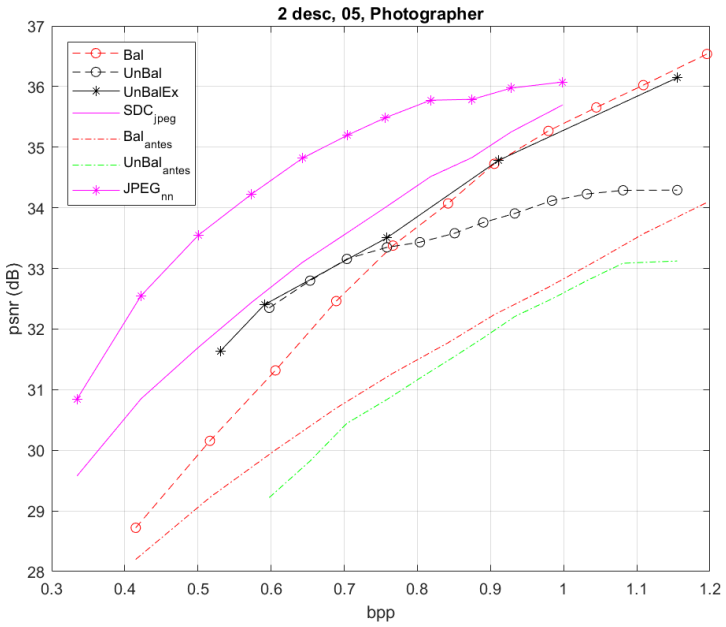


Figura 69 - Resutados Dnn em MDC, imagem 5

A Figura 70 apresenta os resultados subjetivos obtidos e observados para o método não balanceado. Nota-se uma melhoria substancial na qualidade visual obtida pelo método usado de redução de ruído.

Figura 70 - Resultado do método não balanceado específico (bpp = 0,3727). Da esquerda para a direita: input, pré-dncnn (psnr = 32,31 dB), output (psnr = 35,30 dB).

A Figura 71, apresenta os resultados obtidos e observados para o método balanceado. Uma vez que os débitos são baixos, a melhoria não é tão significativa, ainda assim, nota-se uma melhoria na qualidade visual obtida pelo método usado de redução de ruído. Neste caso os ganhos são mais significativos em débitos mais elevados.

Figura 71 - Resultado do método balanceado (bpp = 0,3614). Da esquerda para a direita: input, pré-DnCnn (psnr = 32,87 dB), output (psnr = 33,60 dB).

A Figura 72, apresenta os resultados subjetivos obtidos utilizando o método de redução de ruído a uma codificação normal JPEG. Visualmente verifica-se a melhoria obtida.

Figura 72 - Resultado do método usando apenas JPEG e JPEGnn (melhoramento do JPEG) (bpp = 0,3652). Da esquerda para a direita: input, JPEG (psnr = 35,32), JPEGnn (psnr = 36,67).

Nas Figura 71 à Figura 75 estão apresentados alguns exemplos dos resultados visuais para a imagem 3.

Figura 73 - Resultado do método não balanceado específico (bpp = 0,4160). Da esquerda para a direita: input, pré-DnCnn (psnr = 31,26 dB), output (psnr = 33,76 dB).

Figura 74 - Resultado do método balanceado (bpp = 0,4044). Da esquerda para a direita: input, pré-DnCnn (psnr = 31,83 dB), output (psnr = 32,40 dB).

Figura 75 - Resultado do método de melhoramento do JPEG (JPEGnn) (bpp = 0,4245). Da esquerda para a direita: input, JPEG (psnr = 34,39 dB), JPEGnn (psnr = 35,50 dB).

5.4.3. Estudo por imagem com Uma Descrição

Esta secção apresenta o estudo da utilização de métodos de reconstrução de imagem para codificação de imagem MDC onde podemos ter apenas uma descrição. Os resultados são referentes a codificação balanceada.

Os resultados apresentados entre a Figura 81 e a Figura 83 comparam a codificação MDC balanceada (Bal) com JPEG (SDCJPEG), com a codificação JPEG usando o método de melhoria de imagem (JPEGnn), assim como a reconstrução usando a interpolação bicúbica para a resolução original de uma descrição.

Observando os resultados usando uma descrição, consegue-se ver que os resultados são afetados pela resolução da imagem, visto que com imagens de resolução menor (i.e imagens 2 e 5) o melhoramento é um pouco instável. Supõe-se que como se tem menos informação qualquer erro vai se salientar no cálculo final.

Nota-se também um resultado um pouco diferente na Figura 79 (Barbara), em que os resultados são piores. Estima-se que devido ao facto de a imagem ser maioritariamente

composta por linhas ou estruturas repetitivas, sendo curioso que usando duas descrições essa mesma imagem consegue obter resultados mais coerentes com o resto do estudo. No entanto, pelo facto de retirar informação à imagem (i. e. uma descrição) essas estruturas repetitivas são mais difíceis de reconstruir do que estiverem com um ruído acentuado.

De um modo um modo geral, o método DnCNN aplicado a uma descrição apresenta uma melhoria face à interpolação bicubica, entre 1.5 a 4 dB, o que mostra a efetividade do método usado, face a métodos de interpolação tradicionais.

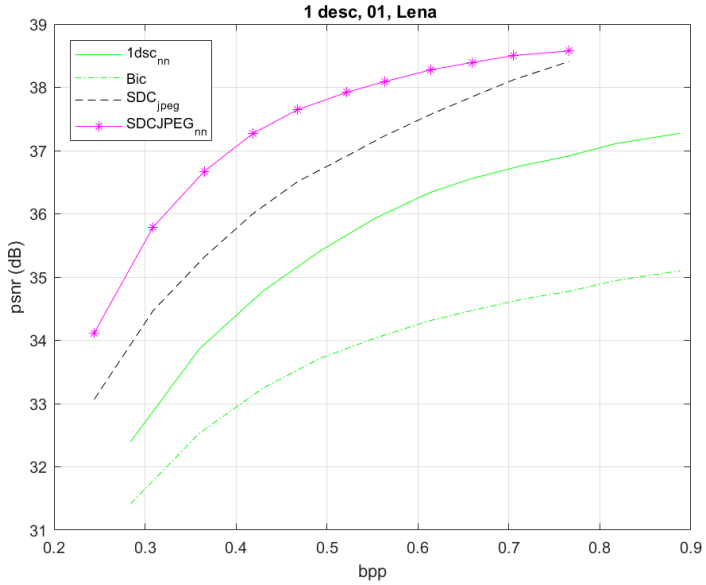


Figura 76 - Resutados Dnn em MDC, com 1 descrição, imagem 1

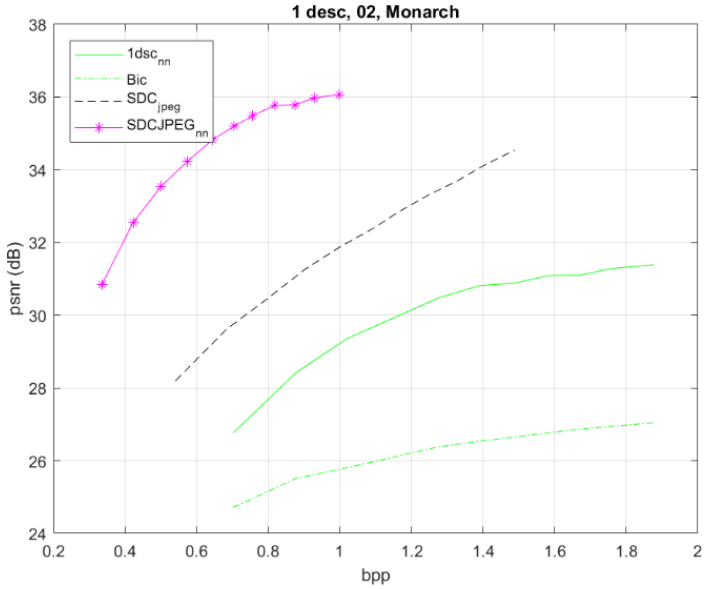


Figura 77 - Resutados Dnn em MDC, com 1 descrição, imagem 2

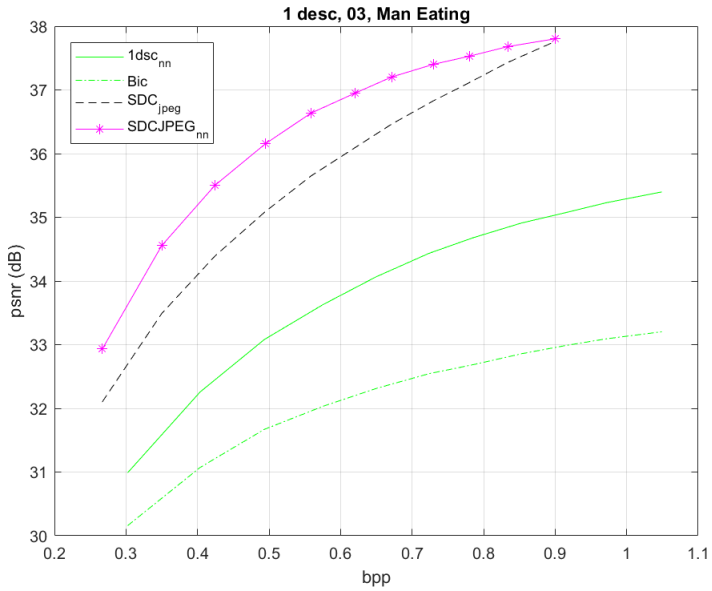


Figura 78 - Resutados Dnn em MDC, com 1 descrição, imagem 3

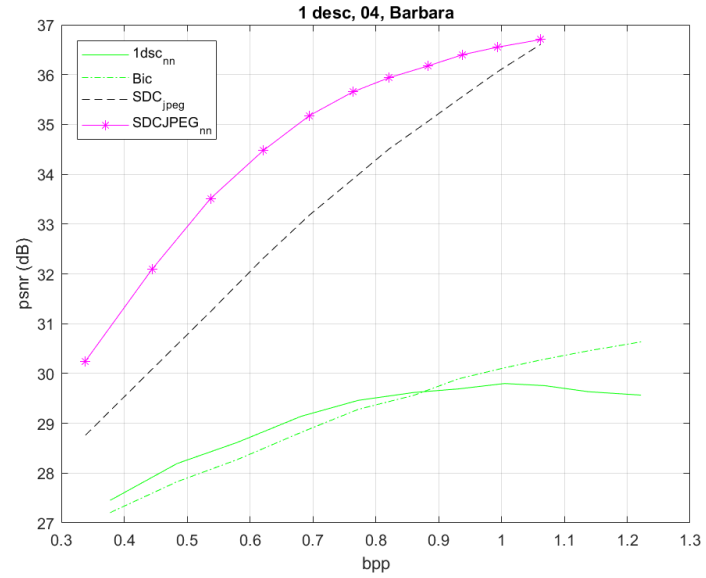


Figura 79 - Resutados Dnn em MDC, com 1 descrição, imagem 4

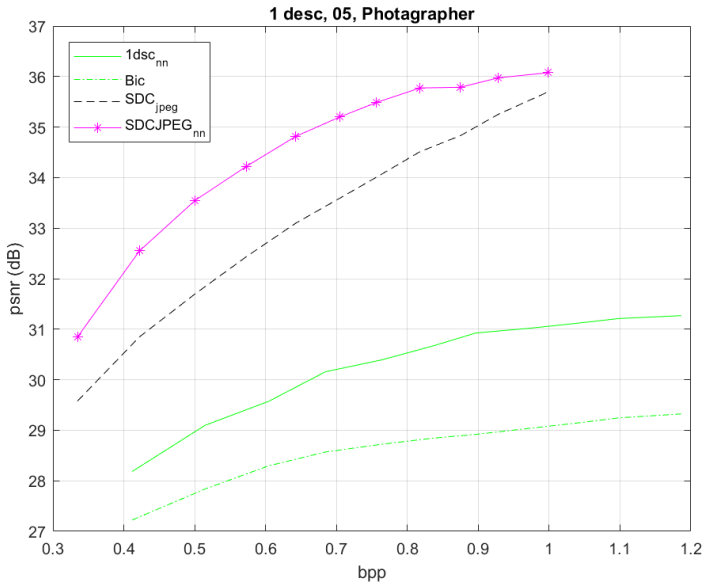


Figura 80 - Resutados Dnn em MDC, com 1 descrição, imagem 5

Nas Figura 81 à Figura 85, estão apresentados resultados subjetivos dos métodos utilizados para a imagem 1(Lena) e para a imagem 3 (Men Eating). É evidente a melhoria da qualidade introduzida pelo método de redução de ruído.

Figura 81 - Resultado do método usando 1 descrição (bpp = 0,04296. Da esquerda para a direita: input, pré-dncnn (psnr = 32,24 dB), output (psnr = 34,70 dB).

Figura 82 - Resultado do método de melhoramento do JPEG (JPEGnn) (bpp = 0,4185). Da esquerda para a direita: input, JPEG (psnr = 36,00 dB), JPEGnn (psnr = 37,27 dB).

Figura 83 - Resultado do método usando 1 descrição (bpp = 0,4038. Da esquerda para a direita: input, pré-dncnn (psnr = 31,07 dB), output (psnr = 32,36 dB).

Figura 84 - Resultado do método de melhoramento do JPEG (JPEGnn) (bpp = 0,4245). Da esquerda para a direita: input, JPEG (psnr = 34,39 dB), JPEGnn (psnr = 35,50 dB).

5.4.4. Visão Global

A Figura 85 – Resultados da visão global sobre os métodos usados.Figura 85 apresenta uma visão global de todos os métodos usados. Juntando todos métodos, aplicando ao set de imagens para avaliação de desempenho com ambas as descrições. Note-se que nos resultados de descodificação de uma descrição, o débito considerado é o resultante de apenas essa descrição. Deste modo, é possível comparar com os resultados do JPEG e JPEGnn, considerando que a descrição é codificada com menor resolução espacial, sendo que na descodificação, esta resolução é re-estabelecida.

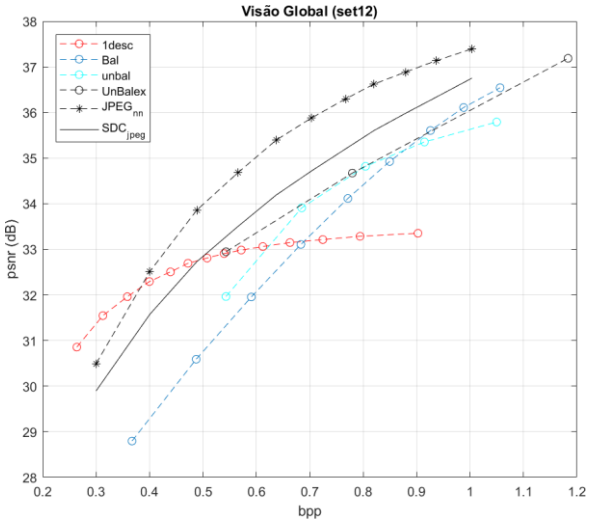


Figura 85 – Resultados da visão global sobre os métodos usados.

Comparando os resultados usando uma ou duas descrições, estes, vão depender da qualidade da imagem. Assim, em níveis mais baixos da qualidade de imagem a descodificação com uma descrição, consegue superar até o melhoramento de JPEG estagnando por volta 0.5 bpp.

Voltando a comparar os métodos Balanceado e Não Balanceado, consegue-se observar melhor que, dependendo da situação, pode haver benefícios em codificar imagens com qualidades diferentes e processá-las numa CNN, onde para menores níveis de qualidade o método Não balanceado consegue superar o método Balanceado.

Estes resultados podem ser incorporados em esquemas de codificação para transmissão com erros, onde em função da taxa de erros, se pode escolher qual a melhor forma de codificação de modo a otimizar a qualidade de descodificação em cada situação.

6. CONCLUSÃO

Neste projeto foi proposto um estudo sobre o uso de CNN com codificação de imagens com múltiplas descrições para método de pós-processamento em complemento dos codificadores convencionais para a melhoria da qualidade da imagem.

Numa primeira fase foi avaliada uma rede pré-treinada aplicada em melhoria de imagem codificada em MDC, utilizando técnicas de restauro de imagem assim como de super-resolução. Os resultados obtidos apresentaram melhorias relativamente a métodos de interpolação tradicionais.

Usando um novo método de codificação MDC, foi treinada uma rede para redução de ruído, de modo a melhorar a qualidade de descodificação, para diferentes fatores de qualidade. O método desenvolvido mostrou efetividade na melhoria das imagens descodificadas, tanto para MDC balanceado e não balanceado, quando se considera a descodificação de cada descrição de forma individual assim como a decodificação das descrições em conjunto. Os resultados obtidos vão em conta ao esperado, e inclusivamente em débitos mais baixos, atingem-se melhores resultados que a codificação SDC.

6.1. Trabalho futuro

Os resultados do presente estudo têm espaço para melhorar, através da utilização de técnicas estado da arte para melhoria de imagem que, entretanto, foram desenvolvidas. Assim um possível trabalho futuro consiste no uso destas técnicas de pós-processamento ao esquema de codificação com múltiplas descrições desenvolvido.

Por outro lado, o treino das DNN pode incorporar modelos de perdas de pacotes de imagem em transmissão multicanal, e assim avaliar em que medida se pode melhorar a qualidade das imagens descodificadas.

Finalmente, estas técnicas podem ser usadas em aplicações em tempo real de arquiteturas de transmissão de imagem para redes sem fios com muitos nós e altamente dinâmicas, que são usadas atualmente no contexto das cidades inteligentes.

BIBLIOGRAFIA

[1] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang and S. Wang, "Image and Video Compression With Neural Networks: A Review," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,* vol. 30, no. 6, JUNE 2020.

[2] F. Zhang, D. Ma, C. Feng and D. R. Bull, Video Compression with CNN-based Post Processing, 2021.

[3] S. Zhu, Z. He, X. Meng, J. Zhou, Y. Guo and B. Zeng, "A New Polyphase Down-Sampling-Based Multiple Description Image Coding," *IEEE TRANSACTIONS ON IMAGE PROCESSING,* vol. 29, 2020.

[4] S. Golabi, M. S. Helfroush and H. Danyali, "Reversible robust data hiding based on waveletfilters modification," *Multimedia Tools and Applications,* 2019.

[5] Z. Zhaoxiang, A. Iwasaki, G. Xu and J. Song, "Cloud detection on small satellites based on lightweight U-net and image compression," *Journal of Applied Remote Sensing,* 2019.

[6] P. Correia, P. Assuncao and V. Silva, "Multiple Description of Coded Video for Path Diversity Adaptation," *IEEE Transactions on Multimedia ,* vol. 14, no. 3, 2012.

[7] F. Zhu, A Review of Deep Learning Based Image Super-resolution Techniques, Xihua University, 2022.

[8] F. Zhu, A Review of Deep Learning Based Image, 2022.

[9] O. Elharrouss, N. Almaadeed, S. Al-Maadeed and Y. Akbari, "Image inpainting: A review," *CoRR,* vol. abs/1909.06399, no. --, pp. --, 2019.

[10] Zhang, "IRCNN," [Online]. Available: https://github.com/cszn/IRCNN.

[11] C. Tiana, L. Feic, W. Zhengd, Y. Xua, W. Zuof and C.-W. Ling, "Deep Learning on Image Denoising: An overview," *arXiv,* Vols. -, no. -.

[12] K. Zhang, W. Zuo, S. Gu and L. Zhang, "Learning Deep CNN Denoiser Prior for Image Restoration," 2017. [Online]. Available: https://github.com/cszn/IRCNN.

[13] [Online]. Available: https://www.nbshare.io/notebook/751082217/Activation-Functions-In-Python/.

[14] [Online]. Available: https://nms.kcl.ac.uk/osvaldo.simeone/ml4eng/Chapter_1_KCL.pdf.

[15] [Online]. Available: https://modelzoo.co/model/loss-landscape.

[16] Zisserman, A. Vedaldi and Andrew, “VGG Convolutional Neural Networks Practical,” [Online]. Available: https://www.robots.ox.ac.uk/~vgg/practicals/cnn/index.html. [Accessed 1 9 2022].

[17] [Online]. Available: https://www.youtube.com/watch?v=V2h3IOBDvrA.

[18] [Online]. Available: https://towardsdatascience.com/review-drn-dilated-residual-networks-image-classification-semantic-segmentation-d527e1a8fb5.

[19] [Online]. Available: https://www.youtube.com/watch?v=ZjM_XQa5s6s&ab_channel=deeplizard.

[20] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang and L. Shao, “Learning Enriched Features for Fast Image,” *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE,* vol. arXiv:2003.06792, 2022.

[21] A. Buades, B. Coll and J.-M. Morel, “A non-local algorithm for image denoising,” 2005.

[22] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang and L. Shao, Learning enriched features for real image restoration and enhancement, In ECCV, 2020.

[23] S. A. Barnes and Nick, Real image denoising with feature attention, ICCV, 2019..

[24] T. Brooks, B. Mildenhall, T. Xue, J. Chen, D. Sharlet and J. T. Barron, Unprocessing images for learned raw denoising, In CVPR, 2019.

[25] S. Guo, Z. Yan, K. Zhang, W. Zuo and L. Zhang, Toward convolutional blind denoising of real photographs, In CVPR, 2019.

[26] Y. Kim, J. W. Soh, G. Y. Park and N. I. Cho, Transfer learning from synthetic to real-noise denoising with adaptive instance normalization, In CVPR, 2020.

[27] E. H. Land, The retinex theory of color vision, 1977.

[28] Y. Zhang, J. Zhang and X. Guo, Kindling the darkness: A practical low-light image enhancer, 2019.

[29] L. Shen, Z. Yue, F. Feng, Q. Chen, S. Liu and J. Ma., Msr-net: Low-light image enhancement using deep convolutional network, 2017.

[30] A. I. a. R. Timofte, Ntire 2019 challenge on image enhancement: Methods and results, 2019.

[31] C. Dong, C. C. Loy, K. He and X. Tang, Image super-resolution using deep convolutional networks, TPAMI, 2015.

[32] C. Dong, C. C. Loy, K. He and X. Tang, Learning a deep convolutional network for image super resolution, In ECCV, 2014.

[33] J. Kim, J. K. Lee and K. M. Lee, Accurate image super-resolution using very deep convolutional networks, In ICCV, 2016.

[34] Y. Tai, J. Yang, X. Liu and C. Xu., Memnet: A persistent memory network for image restoration, In ICCV, 2017.

[35] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, In CVPR, 2016.

[36] Z. Yan, X. Li, M. Li, W. Zuo and S. Shan, Shift-net: Image inpainting via deep feature rearrangement, 2018.

[37] A. Hertz, S. Foge, R. Hanocka, R. Giryes and D. Cohen-Or, Blind visual motif removal from a single image, 2019.

[38] J. Zhao, Z. Chen, L. Zhang and X. Jin, Unsupervised learnable sinogram inpainting network (sin) for limited angle ct reconstruction, 2018.

[39] T. Nakamura, A. Zhu, K. Yanai and S. Uchida, Scene text eraser, 2017.

[40] O. Rukundo, S. E. Schmidt and O. T. V. Ramm, "Software Implementation of Optimized Bicubic Interpolated Scan Conversion in Echocardiography," *arXiv:2005.11269,* Vols. -, no. -, pp. -, 2020.

[41] Matconvnet. [Online]. Available: https://www.vlfeat.org/matconvnet/.

[42] K. Zhang, W. Zuo, Y. Chen and D. Meng, "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising," *IEEE Transactions on Image Processing,* vol. 26, no. 7, pp. 3142-3155, 2017.

[43] Zhang, "https://github.com/cszn/DnCNN," [Online].

[44] www.image-engineering.de. [Online]. Available: https://www.image-engineering.de/library/technotes/745-how-does-the-jpeg-compression-work.

[45] "JPEG XL Image Coding System," [Online]. Available: https://ds.jpeg.org/whitepapers/jpeg-xl-whitepaper.pdf.

[46] [Online]. Available: https://en.wikipedia.org/wiki/Huffman_coding.

[47] D. Marshall, 4 10 2001. [Online]. Available: https://users.cs.cf.ac.uk/Dave.Marshall/Multimedia/node231.html.