



Departamento de Ciências e Tecnologias da Informação

Retocagem Digital

Pedro Miguel Ferreira Lourenço

Dissertação submetida como requisito parcial para obtenção do grau de
Mestre em Engenharia de Telecomunicações e Informática

Orientador:
Doutor Luís Eduardo de Pinho Ducla Soares, Professor Auxiliar,
ISCTE-IUL

Co-Orientador:
Doutor Paulo Jorge Lourenço Nunes, Professor Auxiliar,
ISCTE-IUL

Setembro, 2015

Agradecimentos

Este espaço é curto e não me permite agradecer a todas as pessoas como eu gostaria, que ao longo do meu Mestrado em Engenharia de Telecomunicações e Informática me ajudaram, cada um da sua forma para que eu pudesse atingir os meus objetivos.

Aos meus coordenadores de Dissertação Luís Soares e Paulo Nunes, agradeço pela disponibilidade demonstrada, paciência, profissionalismo e pela orientação que me levaram a finalização desta etapa. Todas as dicas e sugestões para o melhoramento deste projeto que me ofereceram desde o início e pela luta incessante para poder obter o melhor resultado possível.

Quero agradecer a toda a minha família:

Aos meus Pais, Filomena Lourenço e Paulo Lourenço, por todos os sacrifícios que tiveram de fazer ao longo das suas vidas para eu chegar até aqui, um enorme obrigado por todo o apoio, incentivo que me proporcionaram não só no decorrer deste projeto mas em todo o meu percurso académico. Por acreditarem sempre em mim e naquilo que faço e com toda a sabedoria com que me educaram.

Um especial agradecimento que dedico a minha namorada Inês Sabino, tenho muito orgulho na pessoa que amo, que incondicionalmente esteve do meu lado a apoiar, nunca me deixou desistir, quando perdia a motivação e me deixava ir a baixo, bastava esboçar um simples sorriso que conseguia derrubar todo o meu desânimo e voltar acreditar que era possível chegar ao final desta etapa. Foram muitos altos e baixos nesta última fase do Mestrado que levou 2 anos a concluir, mas em todos transmitiu sempre confiança e força.

À minha irmã e cunhado, Patrícia Lourenço e Ricardo Lourenço por estarem sempre disponíveis para me ajudar em qualquer que fosse a minha necessidade e por todo o apoio demonstrado. Também aos meus sobrinhos pelas horas de descontração e divertimento em lazer.

Aos meus sogros, Rosa Freitas e António Sabino a minha gratidão pelo apoio e compreensão ao longo desta fase, pela forma como me integraram na família e pela forma própria de me incentivar a concluir o Mestrado e de melhor em todos os aspetos.

À minha avó Lurdes que sempre com muito carinho me tratou e o orgulho, falava e demonstrava em ter um neto a concluir o curso. Com muita pena minha não poderá assistir ao final do meu Mestrado.

Dedico todo este trabalho à minha família. Confesso que por muitas vezes pensei que todos acreditavam mais que eu próprio em concluir esta Dissertação. Espero que agora depois de finalizar esta etapa, possa, compensar e retribuir todo o apoio, carinho e dedicação que todos me oferecem.

Retocagem Digital

Agradeço, também a todos os docentes que lecionaram os seus conhecimentos e que me permitiu fazer este trabalho e adquirir competências teóricas importantes que serão usados na minha vida profissional.

Não vou enumerar os meus amigos pois todos me ajudaram de alguma forma, por isso, a todos eles agradeço a ajuda, apoio e por todos os momentos de entusiasmo e descontração partilhados.

"Cada um que passa na nossa vida, passa sozinho, pois cada pessoa é única e nenhuma substitui outra. Cada um que passa na nossa vida, passa sozinho, mas não vai só nem nos deixa sós. Leva um pouco de nós mesmos, deixa um pouco de si mesmo. Há os que levam muito, mas há os que não levam nada. Essa é a maior responsabilidade da nossa vida, e a prova de que duas almas não se encontram ao acaso."

(Antoine de Saint-Exupéry) [19]

Abstract

Digital image and video correction and editing have been proliferating. This kind of activities rely on techniques, usually know as Digital Inpainting or Digital Retouching, that have been increasingly growing in popularity as well as acceptance in the course of time. Digital Retouching techniques are relatively recent, becoming popular since the year 2000, and are nowadays widely used in several areas. In image and video correction, the main objective of this technique is essentially to correct "imperfections" that are detectable by the human eye, known as "naked eye". This "imperfections" correction may consist of image scratches or dust spots removal, up to the removal of larger objects such as logos, watermarks, people, etc. Nevertheless, the main goal of Digital Retouching is to make these changes imperceptible to the human viewer or as natural as possible.

This Dissertation considers different techniques that are used to correct the several problems. It was developed a Digital Retouching application to evaluate the performance of various techniques mentioned above through image quality metrics.

Keywords: digital retouching, image inpainting, image editing of the image, image correction

Resumo

A correção e edição de imagem e de vídeo têm vindo a proliferar. Neste tipo de atividades, utiliza-se normalmente uma técnica que tem tido uma aceitação cada vez mais crescente com o passar do tempo, a técnica do Retocagem Digital. O primeiro algoritmo matemático capaz de traduzir o trabalho humano em trabalho computacional, surgiu em 2000 através de Bertalmio e Sapiro. De notar é que, na correção de imagem e de vídeo, o principal foco desta técnica é essencialmente corrigir pequenos “erros”, os quais são detetáveis ao olho humano leia-se “olho nu” numa imagem. Alguns exemplos podem ser enumerados, desde riscos, ou pontos de poeira, até a remoção de outros objetos como os logotipos, marcas de água (watermarks), pessoas, etc. Sendo assim, o objetivo principal tornar essas modificações impercetíveis na imagem.

Posto isto, consideraram-se assim diferentes técnicas que são usadas para corrigir os diferentes problemas supracitados. Estas técnicas podem englobar uma parte mais pequena da imagem na qual se tem de ter em conta apenas a dispersão de cores numa matriz bastante povoada e dispersa, ou na parte maior da imagem, na qual se tem em conta essencialmente quais são os pixéis vizinhos ou até mesmo as cores desses pixéis, de modo a manter a coerência no preenchimento de cada pixel, considerando uma dada área.

Palavras-chave: retocagem digital, edição de Imagem, correção de imagem

Índice

Agradecimentos.....	iii
Abstract.....	v
Resumo.....	vii
Índice.....	ix
Lista de Tabelas	xv
Lista de Abreviaturas	xvii
Capítulo 1	1
<i>Introdução</i>	1
1.1. <i>Contexto</i>	1
1.2. <i>Motivação</i>	3
1.3. <i>Objetivos</i>	4
1.4. <i>Organização da Dissertação</i>	4
Capítulo 2	5
<i>Algoritmos de Retocagem Digital</i>	5
2.1 <i>Retocagem Digital: Revisão Bibliográfica</i>	5
2.1.1 <i>Técnica de Bertalmio e Sapiro</i>	5
2.1.2 <i>Técnica de Chan e Shen</i>	6
2.1.3 <i>Técnica de Criminisi</i>	7
2.1.4 <i>Técnica de Lorenzi, Melgani e Mercier</i>	9
2.2 <i>Análise da Qualidade de Imagem</i>	10
2.2.1 <i>Métricas No-Reference</i>	11
2.2.2 <i>Métricas Full-Reference</i>	11
Capítulo 3	16
<i>Aplicação de Retocagem Digital</i>	16
3.1 <i>Arquitetura da aplicação</i>	16
3.2 <i>Módulo de Seleção e Introdução de Distorção</i>	17
3.3 <i>Módulo de Retocagem da Imagem</i>	21
3.3.1 <i>Algoritmo FoE Model</i>	21
3.3.2 <i>Algoritmo Exemplar-based Retouching</i>	23
3.3.3 <i>Algoritmo Total Variation Image Retouching</i>	24

3.3.4	<i>Algoritmo Coherence Sensitive Hashing</i>	25
3.4	<i>Módulo de Avaliação da Qualidade da Imagem Final</i>	27
3.4.1	<i>Técnicas No-Reference</i>	27
3.4.2	<i>Técnicas Full-Reference</i>	27
Capítulo 4	30
<i>Resultados e Discussão</i>		30
4.1	<i>Condições de Teste</i>	30
4.2	<i>Resultados</i>	32
4.2.1	<i>Teste 1: Remoção de um Derrame no Olho</i>	32
4.2.2	<i>Teste 2: Remoção de uma Pessoa da Imagem</i>	37
4.2.3	<i>Teste 3: Remoção de Texto Sobreposto numa Imagem</i>	40
4.2.4	<i>Teste 4: Aplicação e Remoção de Ruído Salt & Pepper (2% e 4%)</i>	43
4.2.5	<i>Teste 6: Aplicação e Remoção de Riscos numa Imagem</i>	49
4.2.6	<i>Teste 7: Aplicação e Remoção de Texto numa Imagem</i>	52
Capítulo 5	60
<i>Observações Finais</i>		60
5.1	<i>Conclusões</i>	60
5.2	<i>Futuros Melhoramentos</i>	60
<i>Bibliografia</i>		62

Lista de Figuras

- Figura 1.1 Retocagem de uma imagem corrompida por riscos: a) Imagem corrompida; b) Imagem restaurada
- Figura 1.2 Remoção dos óculos: a) Sujeito com óculos b) Seleção dos óculos c) Sujeito sem óculos
- Figura 1.3 Correção de imagem envelhecida: a) Foto antiga corroída pelo tempo b) Foto restaurada
- Figura 2.1 Remoção de texto em imagens através de Retocagem: a) Imagem com texto; b) Imagem reconstruída
- Figura 2.2 Direção de propagação em relação à normal da fronteira da área selecionada
- Figura 2.3 Exemplo de Retocagem usando a técnica de Bertalmio e Sapiro: a) Antes b) Depois
- Figura 2.4 Exemplo de Retocagem usando a técnica de Chan e Shen: a) Imagem com áreas a retocar marcadas a vermelho b) Imagem reconstruída
- Figura 2.5 Remoção de um objeto da imagem: a) Imagem original b) seleção do objeto c) Imagem reconstruída sem o objeto
- Figura 2.6 Relação entre variáveis
- Figura 2.7 Exemplo da utilização da estratégia de multi-resolução para Retocagem Digital
- Figura 2.8 Exemplo da métrica SSIM aplicada a uma imagem
- Figura 2.9 Exemplo de uma comparação entre as métricas MSE e PSNR
- Figura 3.1 Diagrama que representa o funcionamento da aplicação de Retocagem Digital desenvolvida
- Figura 3.2 Interface Gráfica da Aplicação
- Figura 3.3 Definição de uma área de seleção: a) Pontos de delimitação da área de seleção (6 pontos) b) Máscara c) Parte da Imagem selecionada
- Figura 3.4 Definição de uma área de seleção: a) Pontos de delimitação da área de seleção (12 pontos) b) Máscara c) Parte da Imagem selecionada
- Figura 3.5 Exemplos de seleções efetuadas a partir de máscaras existentes
- Figura 3.6 Aplicação de ruído *salt & pepper*: a) Imagem com ruído a 2% b) Máscara do ruído
- Figura 3.7 Aplicação de ruído *salt & pepper*: a) Imagem com ruído a 10% b) Máscara do ruído
- Figura 3.8 Aplicação de riscos: a) Imagem com riscos b) Máscara dos riscos
- Figura 3.9 Aplicação de texto: a), c) Imagens com o texto aplicado; b), d) Máscaras com o texto
- Figura 3.10 Exemplo conversão de RGB para YCbCr
- Figura 3.11 Exemplo imagem no espaço de cores YCbCr utilizado em vídeo e fotografias
- Figura 3.12 Exemplo imagem sem o sujeito no espaço de cores YCbCr final
- Figura 3.13 Exemplo de imagem final em formato RGB restaurada pelo algoritmo FoE Model

- Figura 3.14 Remoção de objetos numa imagem usando o algoritmo de “Exemplar-based Retouching”: a) Imagem Original, b) Máscara de seleção, c) Imagem Restaurada
- Figura 3.15 Exemplo borda extraída através da máscara
- Figura 3.16 Exemplo da variação do fator de confiança ao longo da área selecionada
- Figura 3.17 Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching” (Lambda = $1 \cdot \exp(4)$) (Lambda = $1 \cdot \exp(4)$)
- Figura 3.18 Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching” (Lambda = $1 \cdot \exp(1)$)
- Figura 3.19 Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching” (Lambda = $1 \cdot \exp(2)$)
- Figura 3.20 Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching” (Lambda = $1 \cdot \exp(3)$)
- Figura 3.21 Exemplo de máscara com a informação da primeira componente de cores usada no Algoritmo “Coherence Sensitive Hashing”
- Figura 3.22 Redimensionamento da Imagem Restaurada (4 iterações espaçadas para se notar a diferença de tamanho)
- Figura 3.23 Resultado final após a conversão para RGB usando o no Algoritmo “Coherence Sensitive Hashing”
- Figura 3.24 Exemplo de Retocagem de uma imagem e respetivas métricas de qualidade a) Imagem Original b) Máscara c) Imagem Reconstruída d) Valores de qualidade obtidos
- Figura 4.1 Pasta das imagens utilizadas na aplicação
- Figura 4.2 Pasta das máscaras utilizadas na aplicação
- Figura 4.3 Seis imagens de teste utilizadas (*cow*, *new_original*, *Orange*, *lena*, *olho*, *ilha*)
- Figura 4.4 Máscaras utilizadas para os testes a) e b)
- Figura 4.5 Teste 1 - Remoção de um Derrame no Olho a) Imagem Original b) Máscara
- Figura 4.6 Teste 1 - Remoção de um Derrame no Olho (Algoritmo FoE Model)
- Figura 4.7 Gráficos das Métricas de Qualidade para FoE Model a) BRISQUE b) BIQI
- Figura 4.8 Teste 1 - Remoção de um Derrame no Olho (Algoritmo Exemplar-Based Retouching)
- Figura 4.9 Gráficos das Métricas de Qualidade para Exemplar-Based Retouching a) BRISQUE b) BIQI
- Figura 4.10 Teste 1 - Remoção de um Derrame no Olho (Algoritmo Total Variation Image Retouching)
- Figura 4.11 Gráficos das Métricas de Qualidade para Total Variation Image Retouching a) BRISQUE b) BIQI
- Figura 4.12 Teste 1 - Remoção de um Derrame no Olho (Algoritmo Coherence Sensitive Hashing)

- Figura 4.13 Gráficos das Métricas de Qualidade para Coherence Sensitive Hashing Algorithm a) BRISQUE b) BIQI
- Figura 4.14 Teste 2 - Remoção de uma Pessoa da Imagem a) Imagem Original b) Máscara de seleção
- Figura 4.15 Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo FoE Model)
- Figura 4.16 Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Exemplar-Based Retouching)
- Figura 4.17 Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Total Variation Image Retouching)
- Figura 4.18 Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)
- Figura 4.19 Teste 3 - Remoção de Texto Sobreposto numa Imagem a) Imagem Original b) Máscara
- Figura 4.20 Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo FoE Model)
- Figura 4.21 Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Exemplar-Based Retouching)
- Figura 4.22 Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Total Variation Image Retouching)
- Figura 4.23 Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)
- Figura 4.24 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%) a) Imagem Original b) Máscara
- Figura 4.25 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo FoE Mode)
- Figura 4.26 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Exemplar-Based Retouching)
- Figura 4.27 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Total Variation Image Retouching)
- Figura 4.28 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Coherence Sensitive Hashing Algorithm)
- Figura 4.29 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) a) Imagem Original b) Máscara
- Figura 4.30 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo FoE Model)
- Figura 4.31 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Exemplar-Based Retouching)
- Figura 4.32 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Total Variation Image Retouching)
- Figura 4.33 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Coherence Sensitive Hashing Algorithm)

- Figura 4.34 Teste 6 - Aplicação e Remoção de Riscos numa Imagem a) Imagem Original b) Máscara
- Figura 4.35 Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo FoE Model)
- Figura 4.36 Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Exemplar-Based Retouching)
- Figura 4.37 Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Total Variation Image Retouching)
- Figura 4.38 Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)
- Figura 4.39 Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 25] a) Imagem Original b) Máscara
- Figura 4.40 Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 25] (Algoritmo FoE Model)
- Figura 4.41 Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 25] (Algoritmo Exemplar-Based Retouching)
- Figura 4.42 Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 25] (Algoritmo Total Variation Image Retouching)
- Figura 4.43 Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 25] (Algoritmo Coherence Sensitive Hashing Algorithm)
- Figura 4.44 Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] a) Imagem Original b) Máscara
- Figura 4.45 Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] (Algoritmo FoE Model)
- Figura 4.46 Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] (Algoritmo Exemplar-Based Retouching)
- Figura 4.47 Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] (Algoritmo Total Variation Image Retouching)
- Figura 4.48 Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] (Algoritmo Coherence Sensitive Hashing Algorithm)

Lista de Tabelas

Tabela 1 Teste 1 - Remoção de um Derrame no Olho

Tabela 2 Teste 2 - Remoção de uma Pessoa da Imagem

Tabela 3 Teste 3 - Remoção de Texto Sobreposto numa Imagem

Tabela 4 Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)

Tabela 5 Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%)

Tabela 6 Teste 6 - Aplicação e Remoção de Riscos numa Imagem

Tabela 7 Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25]

Tabela 8 Teste 8 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 40]

Lista de Abreviaturas

BIQI	Blind Image Quality Index
BRISQUE	Blind Reference less Image Spatial Quality Evaluator
dB	Decibel
EDP	Equações Diferenciais Parciais
FoE	Fields of Experts
FR	Full-Reference
MSE	Mean Squared Error
NR	No-Reference
Pixel	Picture Element
PSNR	Peak Signal-to-Noise Ratio
RR	Reduced-Reference
SNR	Signal-to-Noise Ratio
SSIM	Structural Similarity Index
TV	Total Variation

Capítulo 1

Introdução

Neste capítulo é apresentado de uma forma resumida o conceito de Retocagem Digital, descrevendo-se o contexto da sua utilização e os motivos e objetivos para a sua abordagem. Nos capítulos seguintes apresenta-se uma revisão de várias técnicas relevantes nesta área propostas na literatura e é descrita e analisada pormenorizadamente a aplicação de Retocagem Digital desenvolvida no contexto desta Dissertação.

1.1. Contexto

As técnicas de Retocagem já existentes desde o século passado, onde era feita exclusivamente por artistas maioritariamente pintores, os quais restauravam falhas ou defeitos que existissem nos seus quadros. Nos dias que correm e tendo em consideração que nos encontramos na era das imagens digitais, as técnicas supracitadas são essencialmente realizadas através de computadores. Com tal evolução, tornou-se possível restaurar qualquer tipo de imagem. Alguns defeitos que surgem na maior parte das imagens são por exemplo riscos, ponto de poeira (fotos envelhecidas), marcas de água, efeito olhos vermelhos (*red-eyes*), ou até mesmo a falta de uma pequena área na imagem. Exemplo da correção de riscos numa imagem (Figura 1.1).



Figura 1.1: Retocagem de uma imagem corrompida por riscos: a) Imagem corrompida; b) Imagem restaurada

Assim sendo, a Retocagem foi-se tornando cada vez mais numa área de extrema importância para o desenvolvimento do Processamento Digital de Imagem. Um marco importante nesta área é o trabalho de Bertalmio e Sapiro [1], que propuseram em 2000 o método de restauro de imagem baseado em Equações Diferenciais Parciais (EDP). No seu trabalho, Bertalmio e Sapiro fizeram uso do algoritmo EDP para replicar no domínio das imagens digitais as técnicas básicas já anteriormente usadas por artistas experientes na área. Este algoritmo baseia-se na ideia de que depois de seleccionadas as áreas a serem restauradas, estas sejam preenchidas automaticamente, fazendo uso de informação que as rodeia, ou seja, adjacente. Para tal, usam-se as linhas *isofoto*, i.e., linhas de contorno que contêm o mesmo nível de luminosidade situadas nos limites das regiões. É ainda importante referir que, em contraste com aproximações anteriores, nesta técnica não é necessário especificar a proveniência da informação adicional (ou nova).

Desta forma, a Retocagem torna-se numa ferramenta bastante útil para corrigir imagens mantendo a consistência da maior parte da imagem original com a imagem restaurada, exceto a área que foi submetida a restauração, pois visualmente a área corrigida será, na maior parte dos casos, indetetável. A área que se pretende corrigir, na aplicação de Retocagem desenvolvida nesta dissertação, é feita através de uma seleção manual pelo utilizador para que depois seja aplicado o método de Retocagem, mas também é possível a seleção ser feita de forma automática pela aplicação. Essa área pode conter um objeto que se pretenda retirar (e.g., os óculos no sujeito da Figura 1.2), como pode ter um risco ou outro problema/defeito qualquer (e.g., imagem envelhecida da Figura 1.3) e será essa área que se pretende corrigir.



Figura 1.2: Remoção dos óculos: a) Sujeito com óculos b) Seleção dos óculos c) Sujeito sem óculos



Figura 1.3: Correção de imagem envelhecida: a) Foto antiga corroída pelo tempo b) Foto restaurada

Para o preenchimento das áreas selecionadas é tomada em conta a zona em redor, porque é através dos pixéis vizinhos que vai ser feito o “cálculo” e seguidamente a propagação dos dados a preencher.

1.2. Motivação

Existem muitas situações em que as imagens digitais ou vídeos têm elementos indesejados em si. Estes podem ser riscos ou pontos de poeira no caso de fotografias antigas que foram Digitalizadas, mas também pode ser objetos indesejados (e.g., óculos, carros, etc.) numa fotografia Digital que estavam presentes no momento da captura da imagem. Outros exemplos, incluem a data na foto, marcas de água colocados por vários sites, e até pessoas não desejadas numa foto ou vídeo.

A Retocagem pode definir-se então como o processo de reconstruir partes perdidas ou deterioradas das imagens e vídeos, bem como a substituição de partes indesejadas. Por exemplo, no mundo dos museus, no caso de uma pintura valiosa, esta tarefa de restauração seria efetuada por um perito de obras de arte. No mundo Digital, a Retocagem é feita através da aplicação de algoritmos sofisticados para substituir partes perdidas ou corrompidas, por exemplo: 1) uma imagem com informação em falta que pretendemos recuperar, por exemplo, uma imagem que contém riscos; 2) outro caso possível é a existência de algum objeto que pretendemos retirar da imagem, por exemplo, uma imagem onde estejam duas pessoas e pretendemos remover uma delas.

Existem muitas aplicações para estas técnicas. Na fotografia e no cinema, estas técnicas são utilizadas para restauração de filmes (e.g., remoção de fissuras em fotografias ou arranhões e manchas de poeira no cinema). São também utilizadas para a remoção do efeito do olho vermelho, a data que aparece nas fotografias e a remoção de outros objetos para fins criativos.

Além disso, estas técnicas também podem ser usadas para substituir os blocos perdidos na transmissão de imagens codificadas, por exemplo, numa transmissão de vídeo.

1.3. Objetivos

Os objetivos desta dissertação são:

- Efetuar uma revisão bibliográfica das principais técnicas de Retocagem Digital;
- Implementar uma aplicação de Retocagem Digital com uma interface gráfica de fácil utilização a ser desenvolvida em MATLAB, de modo a permitir a aplicação de várias técnicas de Retocagem a imagens em diferentes formatos;
- Avaliar o desempenho de várias técnicas de Retocagem Digital de modo interativo, de modo que seja possível ao utilizador da aplicação comparar os resultados obtidos com as diferentes técnicas implementadas e averiguar qual a melhor para uma determinada situação.

1.4. Organização da Dissertação

A Dissertação está dividida em cinco capítulos:

- No Capítulo 1, que consiste na introdução à Dissertação, são apresentados o contexto, a motivação e os objetivos do trabalho efetuado.
- No Capítulo 2 é apresentada a revisão do estado de arte; está subdividido em três secções onde se descrevem de uma forma breve os conceitos básicos de Retocagem Digital e as principais técnicas propostas na literatura. Neste capítulo apresenta-se ainda uma revisão das métricas usadas para avaliação de qualidade das imagens “retocadas”.
- No Capítulo 3 é apresentada a aplicação de Retocagem Digital desenvolvida no contexto desta Dissertação; está subdividido em quatro subsecções, onde se descreve a estrutura funcional da aplicação, bem como o seu funcionamento, os algoritmos de distorção de imagem, os algoritmos de Retocagem Digital implementados e as métricas de qualidade de imagem usadas.
- No Capítulo 4 é efetuada a análise e discussão dos estando dividido em duas secções; numa primeira onde são definidas as condições experimentais e o conjunto de testes efetuados, e numa segunda, onde são analisados os resultados obtidos.
- Finalmente, no Capítulo 5 são apresentadas as conclusões finais desta Dissertação e também algumas sugestões para trabalho futuro.

Capitulo 2

Algoritmos de Retocagem Digital

A Retocagem é composta por diversas técnicas utilizadas em imagens e vídeo. Este tipo de técnicas em imagens são utilizadas para a remoção de objetos ou até mesmo para restauração de imagens antigas, como referido no Capítulo 1. No vídeo, temos o caso do cinema onde é utilizado na remoção de algo que apareça na gravação de cenas.

2.1 Retocagem Digital: Revisão Bibliográfica

2.1.1 Técnica de Bertalmio e Sapiro

No final do Século XX, com a disseminação dos dispositivos digitais atingiu-se uma nova era na criação, captura e manipulação da imagem.

Em 2000, surgiu o primeiro algoritmo de Retocagem para imagens digitais por Bertalmio e Sapiro [1], introduzindo o método de restauro de imagem que se baseia em Equações Diferenciais Parciais, onde o utilizador em primeiro lugar terá de selecionar a área que pretende corrigir e em seguida será aplicado o algoritmo de recuperação ou remoção de objetos (ver exemplo da Figura 2.1).



Figura 2.1: Remoção de texto em imagens através de Retocagem: a) Imagem com texto; b) Imagem reconstruída

Este algoritmo consiste na propagação da informação numa área em Retocagem, onde $I^n(i, j)$ representa o valor de cada pixel com coordenadas (i, j) da área selecionada Ω , no instante de tempo t e a atualização da informação de cada pixel é dada por $I_t^n(i, j)$ a uma taxa Δt (taxa essa que representa a velocidade da atualização da imagem). Este algoritmo pode ser descrito pela equação (1):

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_t^n(i, j), \forall (i, j) \in \Omega \quad (1)$$

A equação (1) representa o conceito usado por Bertalmio [4] na resolução do problema do Retocagem Digital. Para cada pixel vai existir uma melhoria, $I^{n+1}(i, j)$, adicionando à informação atual do pixel, $I^n(i, j)$, no instante t a informação atualizada, $I_t^n(i, j)$, a uma taxa Δt a cada passo da iteração.

Este algoritmo é aplicável a qualquer imagem a cores ou em níveis de cinzento. Propaga a informação através das extremidades da área selecionada e mantém o ângulo de incidência sobre a região selecionada. Evitando, assim, que exista perda de ângulos, em áreas pequenas, garantindo a direção da propagação do algoritmo, corrigindo-a e não se tornando visível ao olho humano qualquer tipo de irregularidade. Essa informação é propagada desde as extremidades até ao centro da área, até ficar totalmente coberta, tendo em conta sempre os pixels vizinhos (Figura 2.2).

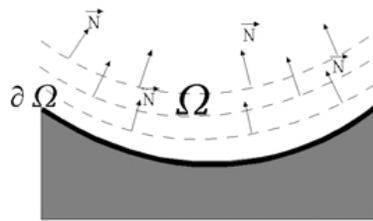


Figura 2.2: Direção de propagação em relação à normal da fronteira da área selecionada

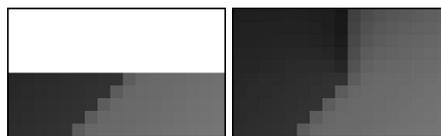


Figura 2.3: Exemplo de Retocagem usando a técnica de Bertalmio e Sapiro: a) Antes b) Depois

2.1.2 Técnica de Chan e Shen

Em 2001, Chan e Shen [2] propuseram uma nova abordagem para Retocagem Digital baseada no conceito de Variação Total (TV). O algoritmo proposto é aplicado em pequenas áreas da imagem tendo em conta todas as informações da imagem e também as áreas que de alguma forma foram perdidas. Este algoritmo não procura áreas que estejam perdidas ou pixels vazios da imagem, mas sim

em reconstruir a imagem toda. Sendo um algoritmo iterativo que procura preencher as áreas vazias com as informações dos pixels mais próximos e de medidas de propagação da informação.

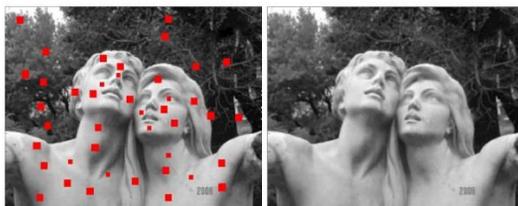


Figura 2.4: Exemplo de Retocagem usando a técnica de Chan e Shen: a) Imagem com áreas a retocar marcadas a vermelho b) Imagem reconstruída

2.1.3 *Técnica de Criminisi*

Em 2004, Criminisi [3][3] propôs uma técnica baseada em uma região de Retocagem, isto é, uma região onde vai ser aplicada a técnica de forma a reconstruí-la da melhor forma. Onde se usa a geometria e as texturas reais presentes nas imagens para reconstruir as áreas pretendidas (Figura 2.5).



Figura 2.5: Remoção de um objeto da imagem: a) Imagem original b) seleção do objeto c) Imagem reconstruída sem o objeto

Os algoritmos anteriores por não usarem as texturas das imagens digitais, não conseguem reconstruir grandes áreas, porque ficarão sempre visíveis alguns artefactos ou zonas da imagem desfocadas. Esses artefactos não estão tão presentes na técnica de Criminisi pois nesta técnica a estrutura da imagem é dividida em duas partes: de um lado ficamos com a informação dos pixels da imagem e do outro a textura da imagem.

Esta reconstrução é possível graças à propagação da informação dos pixels de uma área da imagem original que representa melhor a área que se pretende recuperar (Figura 2.6).

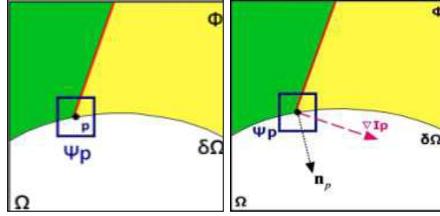


Figura 2.6: Relação entre variáveis

A técnica de Criminisi pode ser descrita resumidamente através dos seguintes passos:

1. Cálculo da Prioridade computacional, $P(p)$, de cada pixel, onde é necessário calcular o preenchimento ao pé do rebordo da área selecionada, equação (2):

$$P(p) = C(p)D(p) \quad (2)$$

Em seguida, o termo de confiança $C(p)$, equação (3):

$$C(p) = \frac{\sum_{q \in \psi_p \cap (I - \Omega)} C(q)}{|\psi_p|} \quad (3)$$

Em que $C(p)$ representa o nível de confiança da informação no pixel p , que diminui à medida que se aproxima do centro da área selecionada. A propagação é assim feita através dos pixels com maior nível de confiança que vai diminuindo até ao centro. Temos então:

- $C(p) = 1$ quando $p \in I - \Omega$
- $C(p) = 0$ quando $p \in \Omega$

Onde $|\psi_p|$ é a área tendo em conta ψ_p que é um bloco de pixels centrado em p , I é o domínio da imagem.

O termo de consistência de dados, $D(p)$, é calculado através da equação (4):

$$D(p) = |\nabla I_p \cdot \vec{n}_p| \propto \quad (4)$$

(isto é, quanto maior for a consistência dos dados entre o pixel p e a forma do rebordo da área, mais importante será o pixel, tendo maior prioridade para ser preenchido).

Onde \propto é o factor normalizado (por exemplo, $\propto = 255$ é um valor típico para imagens em nível de cinzento), ∇I_p é o isofoto (direção e intensidade, linhas de contorno que contêm o mesmo nível de luminosidade situadas nos limites das áreas) por ponto p , \vec{n}_p é o vetor unitário normal ao rebordo da área $\partial\Omega$ no ponto p e \cdot expressa o vector ortogonal.

2. A propagação da textura e estrutura da informação. Onde é feita a pesquisa na área de origem para se encontrar qual o melhor bloco que mais semelhanças tem com ψ_p :

$$\Psi_{p'} = \arg \max_{p \in \partial \Omega^t} P(p) \quad (5)$$

Onde o caminho $\Psi_{p'}$ maximiza a prioridade computacional $P(p)$. Sabendo o ponto central da área é possível manter a mesma regra de prioridades para cada bloco.

3. Atualização dos valores de confiança. Assim que o pixel for preenchido com o novo valor de Ψ_p , a confiança $C(p)$ tem de ser atualizada e a área delimitada por $\Psi_{p'}$:

$$C(q) = C(p'), \forall q \in \Psi_{p'} \cap \Omega \quad (6)$$

Então $P(p)$ vai ser recalculado até que o processo de preenchimento da área seja completado.

2.1.4 Técnica de Lorenzi, Melgani e Mercier

Em 2011, Lorenzi [12], e Lorenzi, Melgani e Mercier [7] utilizaram técnicas já existentes (e.g., a técnica de Criminisi, referida anteriormente), para criar uma melhor estratégia de reconstrução de imagens. A nova estratégia assenta no processamento multi-resolução da imagem, que inicialmente começa com o método de Criminisi [3] para preencher as áreas em falta. Numa segunda iteração deste processo vai sofrer uma alteração, isto é, em vez de ter em conta a imagem original, vai analisar os dados da iteração anterior. Para não haver perda de dados de todas as iterações e não se aplicar apenas a substituição de áreas em falta, é aplicado um cálculo que consiste na soma ponderada para obter o novo contorno. Este processo é repetido até se obter um valor menor que o predefinido W (sendo W o tamanho da imagem, W_i e W_F que representa valores iniciais e finais do tamanho da imagem), obtidos através destes três passos:

1. Corrigir os valores iniciais e finais do tamanho da imagem W_i e W_F respetivamente, com $i = 1$ e $I_i = I$.
2. Aplicar a técnica de Criminisi e calcular Ψ_{pn} :

$$\Psi_{pn} = \Psi_{qn}, \text{ se } i = 1$$

$$\Psi_{pn} = \alpha \Psi_{qn} + (1 - \alpha) \Psi_{pn-1}, \text{ caso contrario; } i = i + 1$$

, onde a imagem de saída será I_i com o conjunto $W_i = W_{i-1}/2$.

3. Se $W_i \geq W_F$ volta para o ponto anterior (2), caso contrário o algoritmo pára.

Onde α é o factor que regula a mudança progressiva na transformação da imagem. Dado pela equação (7):

$$\Psi_{pn} = \alpha \Psi_{qn} + \alpha (1-\alpha) \Psi_{qn-1} + \dots + \alpha (1-\alpha)^{n-2} \Psi_{q2} + (1-\alpha)^{n-1} \Psi_{q1} \quad (7)$$

De outra forma esta é uma estratégia que aplica vários níveis de iteração, repetindo o processo até obter o melhor resultado possível, isto é, quando $W_i \geq W_F$.

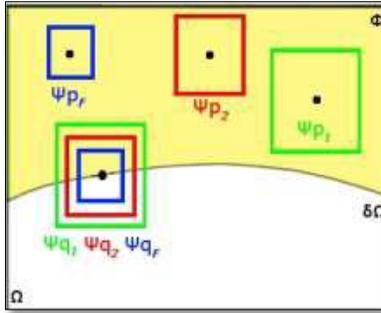


Figura 2.7: Exemplo da utilização da estratégia de multi-resolução para Retocagem Digital

2.2 Análise da Qualidade de Imagem

Em termos de análise da qualidade de imagem existem, essencialmente, três abordagens fundamentais, tendo em conta informação da imagem original que é usada:

- **“No-Reference” (NR)** - As métricas NR são métricas que no seu cálculo utilizam somente a informação da imagem final, não utilizando qualquer referência sobre a imagem original (a avaliação da qualidade de imagem é feita através de modelos de distorção genéricos que são baseados em estatísticas que funcionam no domínio espacial), sendo este cálculo limitado nas suas previsões finais sobre a qualidade da imagem.
- **“Full-Reference” (FR)** - As métricas FR tem referência da imagem original antes de esta sofrer qualquer tipo de distorção, compressão, etc., logo são calculadas tendo em conta a imagem original e final. Uma desvantagem é que normalmente o uso da imagem original requer mais processamento computacional.
- **“Reduced-Reference” (RR)** - As métricas RR têm conhecimento da informação parcial da imagem original para tentar comparar com a qualidade da imagem final, distorcida.

2.2.1 Métricas No-Reference

- **Blind Reference less Image Spatial Quality Evaluator (BRISQUE)[14]** – Esta métrica usa os valores de coeficiente de luminância para quantificar as possíveis perdas de “realidade” da imagem. As outras características utilizadas no cálculo derivam da luminância. O BRISQUE tem uma baixa complexidade computacional, tornando-se num dos melhores métodos para utilizar.
- **Blind Image Quality Index (BIQI)[16]** – Sendo uma métrica NR só utiliza a informação da imagem distorcida da qual divide a informação em pequenos blocos de pixéis.

$$BIQI = \sum_{i=1}^m p_i q_i \quad (8)$$

Onde p_i é a probabilidade de cada distorção na imagem e q_i é o índice de qualidade correspondente a essa distorção. O resultado varia entre 0 e 100; 0 representa o melhor resultado de qualidade e 100 representa o pior resultado de qualidade.

2.2.2 Métricas Full-Reference

- **Mean Squared Error (MSE)[17]** – Nesta métrica é calculado um valor quadrático entre a imagem original e a imagem final. Esse cálculo é obtido para cada pixel, somando as diferenças ao quadrado de todos os pixéis (pixel por pixel) e dividindo pelo número total de pixéis. Quanto menor for o valor de MSE, mais próximo vai estar a imagem distorcida da imagem inicial;

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \quad (9)$$

I é a informação matricial da imagem original e K a imagem final, m e n representam as dimensões da imagem.

- **Structural Similarity Index (SSIM)[20][21]** – No cálculo desta métrica são usadas duas imagens, a imagem original, sem ruído ou qualquer distorção para ser utilizada como referência na comparação com a imagem final reconstruída, tratando-se de um método FR. Este método foi criado com o objetivo de melhorar a medida de qualidade das imagens, em comparação com o MSE e posteriormente o PSNR (que advém do MSE), pois estes provaram não ser muito bons em comparação com o olho humano. Neste método é feita uma abordagem que estima os erros perceptíveis, isto é, deteta as mudanças estruturais e visualmente perceptíveis entre as imagens.

Onde, μ_x , μ_y , σ_x e σ_y são as variáveis locais que representam a derivações e covariância entre as imagens x e y (original e final), as constantes $C1$ e $C2$ (são valores pequenos) servem para evitar que o denominador seja 0.

Retocagem Digital

$$SSIM_{(x,y)} = \frac{(2\mu_x\mu_y + C_1) * (\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1) * (\sigma_x^2 + \sigma_y^2 + C_2)} \quad (10)$$

O resultado desta métrica varia entre 0 e 1, quanto mais próximo estiver do valor 1, mais próxima a imagem final vai estar da imagem original.

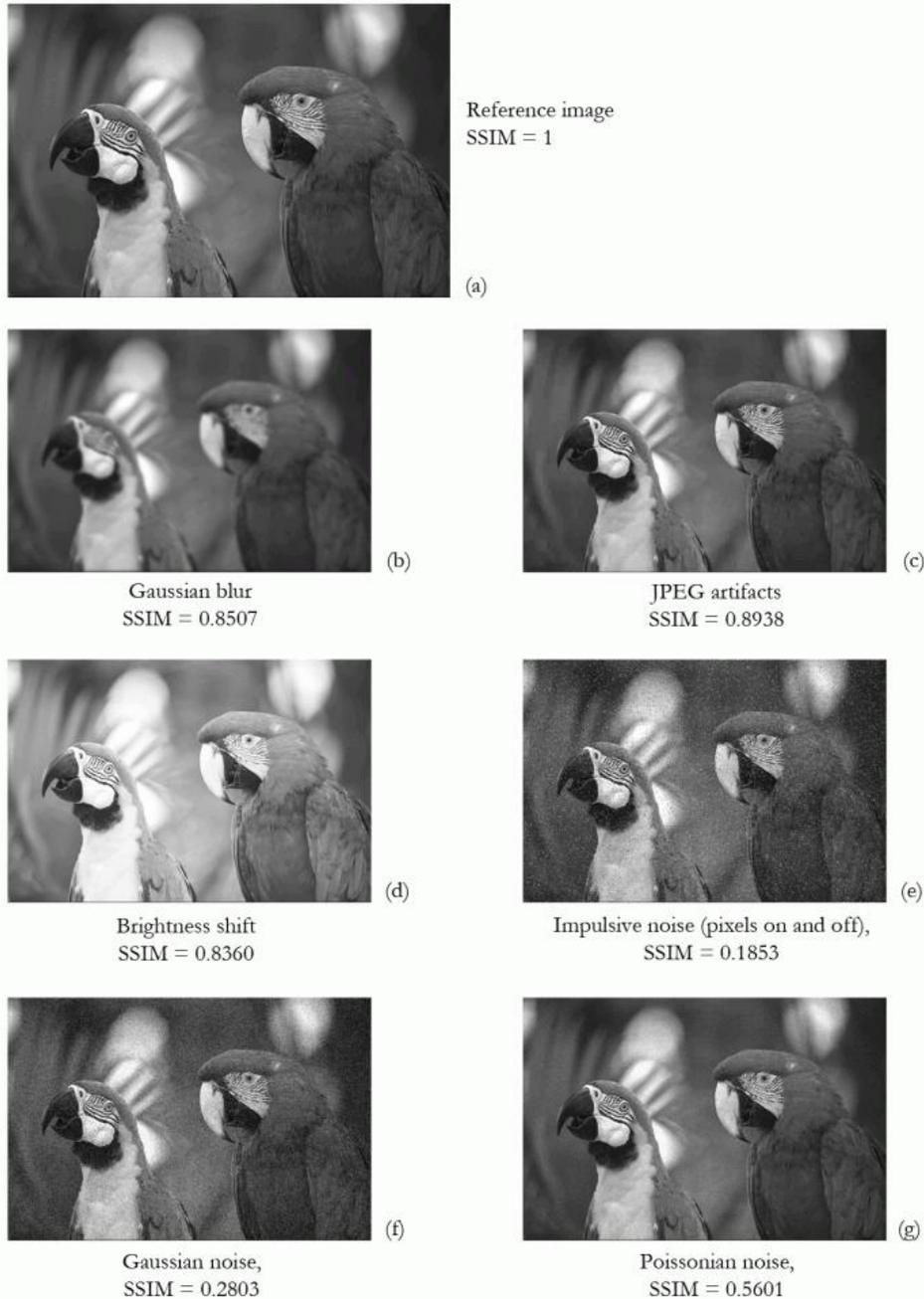


Figura 2.8: Exemplo da métrica SSIM aplicada a uma imagem [17]

Como é possível ver no exemplo da Figura 2.8, o valor de SSIM está coerente em relação aos vários resultados obtidos, a imagem distorcida que mais se aproximou da imagem original foi a (c), onde se obtém o maior valor de SSIM (mais próximo de valor 1).

- **Peak Signal-to-Noise Ratio (PSNR)[17]** – É uma métrica que obtém bons resultados e não requer de grande capacidade computacional. Quanto mais elevado for o valor de PSNR (dB) de uma imagem, mais próximo vai estar a imagem final da imagem original. Utilizando a medida MSE;

$$PSNR = 10 * \log_{10} \left(\frac{R^2}{MSE} \right) \quad (11)$$

R representa o máximo valor possível no pixel, sendo a escala de 0 a 255, R = 255.

Retocagem Digital

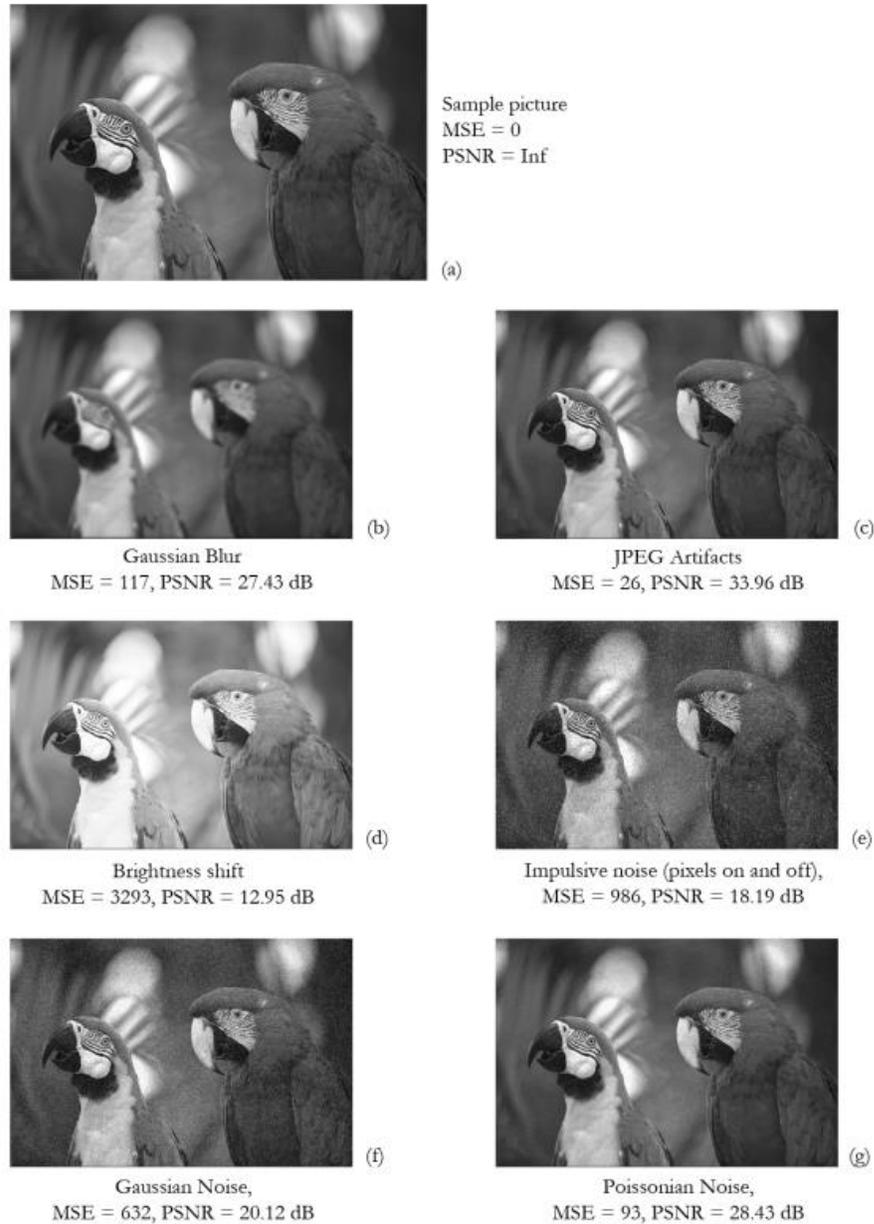


Figura 2.9: Exemplo de uma comparação entre as métricas MSE e PSNR [17]

Como é possível ver no exemplo da Figura 2.9, MSE e PSNR não estão coerentes em relação aos vários resultados; a imagem distorcida que mais se aproximou da imagem original foi a (c), onde se obtém o menor valor para MSE e o maior valor em dB para PSNR, mas o pior resultado visual é a imagem (f) e não a imagem (d) que apresentar os piores resultados para as métricas de MSE e PSNR, isso é aconteceu devido à imagem ter um contraste diferente.

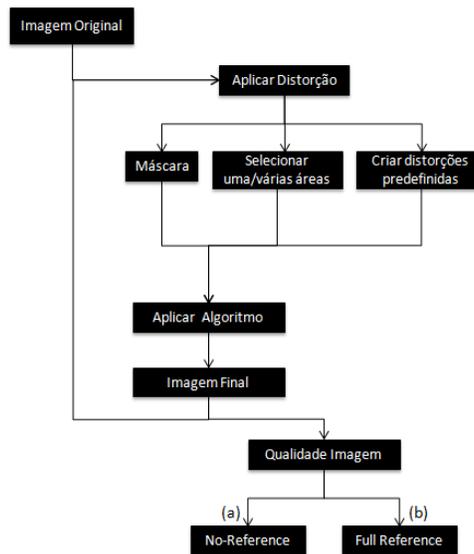
Capítulo 3

Aplicação de Retocagem Digital

Neste capítulo apresenta-se a arquitetura da aplicação desenvolvida, bem como as técnicas de Retocagem Digital utilizadas para o tratamento de imagem nela incluídas e referidas anteriormente no Capítulo 2, referentes a Bertalmio e Sapiro, Shen, Criminisi e Lorenzi.

3.1 Arquitetura da aplicação

Este capítulo vai ser dividido em três partes de acordo com o diagrama UML (Unified Modeling Language) referido anteriormente: Secção 3.2. – “Módulo de Seleção e Introdução de Distorção”, Secção 3.3. – “Módulo de Retocagem da Imagem”; e Secção 3.4. – “Módulo de Avaliação da Qualidade da Imagem Final”.



(a) a sua execução só depende da imagem final, (b) a execução depende de ter uma imagem original sem distorções em conjunto com a imagem final recuperada.

Figura 3.1: Diagrama que representa o funcionamento da aplicação de Retocagem Digital desenvolvida

Na Figura 3.1. apresenta-se o modelo de funcionamento da aplicação de Retocagem Digital desenvolvida nesta Dissertação onde posteriormente podem ser acrescentadas mais funções e técnicas de Retocagem para “enriquecer” esta aplicação.

Após o carregamento da imagem na aplicação são disponibilizados diversos métodos para distorção da imagem, sendo possível selecionar uma região da imagem à qual se quer efetuar o restauro.

Uma opção possível é “estragar” a imagem (inserir defeitos) e posteriormente tentar reconstruir essa área. Para tal, aplica-se um dos quatro algoritmos de Retocagem Digital disponíveis na aplicação, que vão ser descritos mais à frente, obtendo-se assim a imagem final, que será a imagem restaurada (que pode simplesmente ser reconstruída ou corrigida de algum defeito). A outra opção possível é carregar uma imagem já com defeitos.

Nesta aplicação, podemos também obter uma medida da qualidade da imagem obtida através dos algoritmos de qualidade de imagem implementadas (Full-Reference e No-Reference descritas no Capítulo 2).

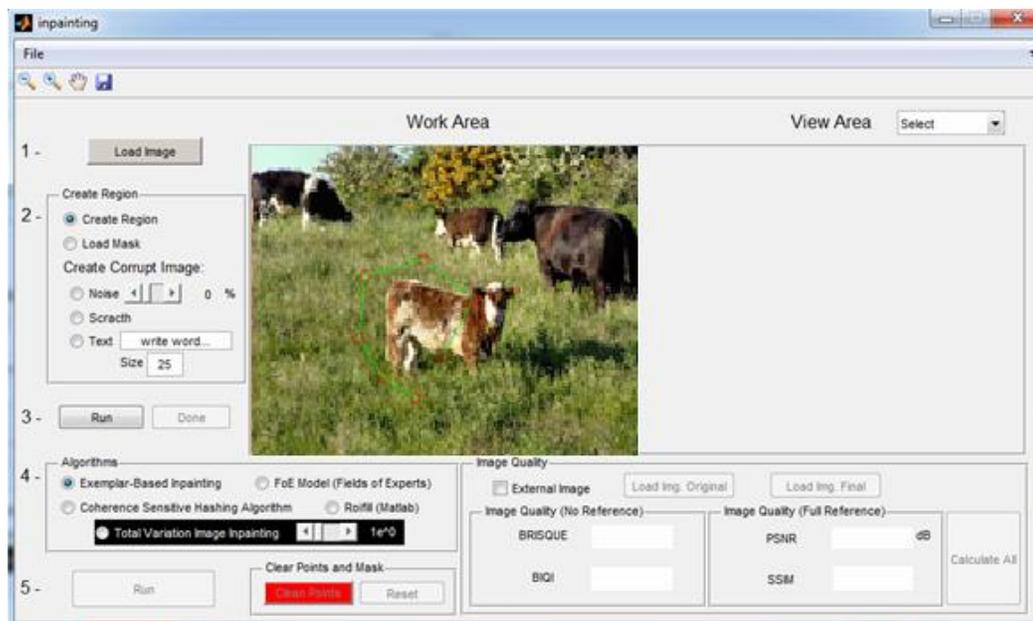


Figura 3.2: Interface Gráfica da Aplicação de Retocagem Digital desenvolvida

3.2 Módulo de Seleção e Introdução de Distorção

Nesta secção descreve-se cada um dos vários métodos de seleção/distorção de imagem, isto é, onde é possível selecionar a área da imagem que se pretende corrigir. Para tal, o utilizador criar uma máscara através da opção dos pontos de seleção ou pode usar uma máscara existente. Na parte da

distorção, é possível aplicar três tipos de distorção: i) ruído *salt & pepper*, ii) sobrepor linhas verticais e horizontais à imagem (formando um padrão de quadrados), iii) sobreposição de texto à imagem.

Todas as máscaras criadas são guardadas numa pasta “mascara” para possíveis utilizações futuras.

Em primeiro lugar vai ser descrito o método de seleção de uma área através de pontos, que requer no mínimo 3 pontos para poder formar uma região de seleção e não tem limite máximo de pontos (Figuras 3.3. e 3.4).

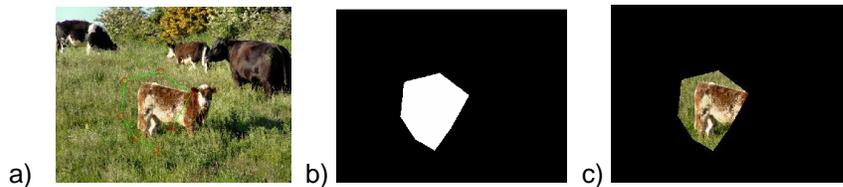


Figura 3.3: Definição de uma área de seleção: a) Pontos de delimitação da área de seleção (6 pontos) b) Máscara c) Parte da Imagem selecionada

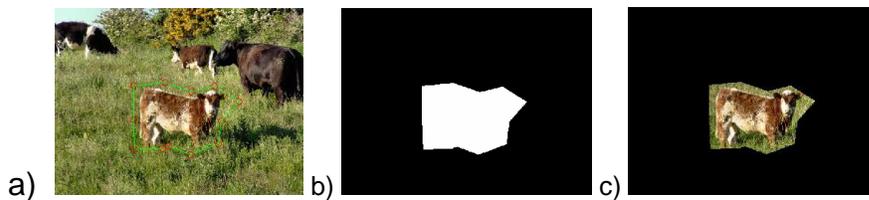


Figura 3.4: Definição de uma área de seleção: a) Pontos de delimitação da área de seleção (12 pontos) b) Máscara c) Parte da Imagem selecionada

Em segundo lugar existe a opção de selecionar uma máscara já existente, que pode conter delimitações das áreas a corrigir mais bem definidas (Figura 3.5).

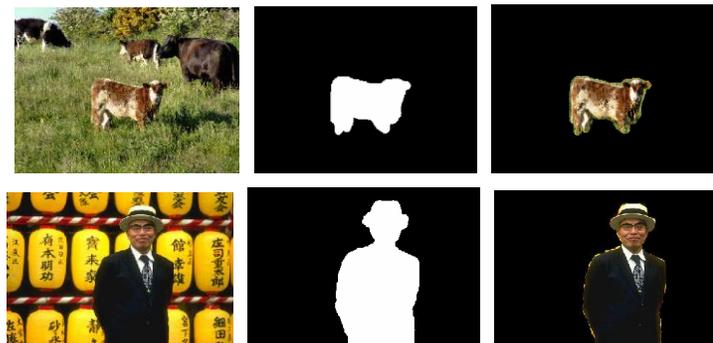


Figura 3.5: Exemplos de áreas de seleção definidas a partir de máscaras existentes

Na parte de aplicação de distorção à imagem original é possível, como referido acima, introduzir ruído, riscos e texto.

A introdução de ruído é efetuada recorrendo à função *imnoise()* do MATLAB, que “aplica” ruído do tipo *salt & pepper* (i.e., tipo grãos de sal espalhados em cima de uma imagem). Neste caso o utilizador seleciona a imagem à qual pretende aplicar o ruído e a percentagem de ruído que quer aplicar. Para melhores resultados esta percentagem deve variar entre 0% e 10%; no entanto pode ir até 100%. Para percentagens elevadas de ruído deixa de ser possível ter uma área aceitável da imagem inicial para poder ser restaurada com sucesso. Este parâmetro (dado como uma percentagem) indica a percentagem de pixéis que vão sofrer alteração e, portanto, uma percentagem de 100% corresponde a ter todos os pixéis afetados; 0% corresponde a não ter ruído nenhum. Os artefactos criados na imagem através deste método, são calculados da seguinte fórmula: $p(\%) \times \text{número_pixeis_imagem}$, onde $p(\%)$ é a percentagem definida pelo utilizador e $\text{número_pixeis_imagem}$ o número de pixéis da imagem. Sendo assim gerado o ruído final e aplicado de forma aleatória. Ao mesmo tempo que é gerada a imagem com ruído é também gerada a máscara desse mesmo ruído e guardada na pasta das máscaras.

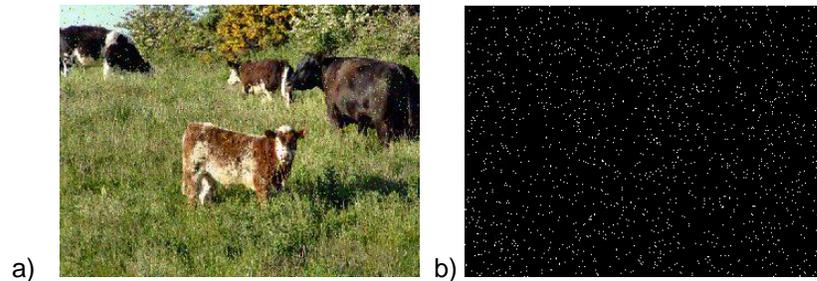


Figura 3.6: Aplicação de ruído *salt & pepper*: a) Imagem com ruído a 2% b) Máscara do ruído

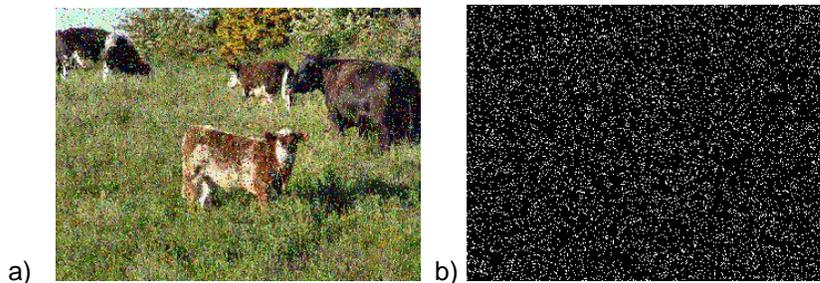


Figura 3.7: Aplicação de ruído *salt & pepper*: a) Imagem com ruído a 10% b) Máscara do ruído

Para a aplicação de riscos apenas é feita uma sobreposição de imagens, isto é, uma soma entre os pixéis da imagem original e os pixéis em branco presentes na máscara.

Retocagem Digital

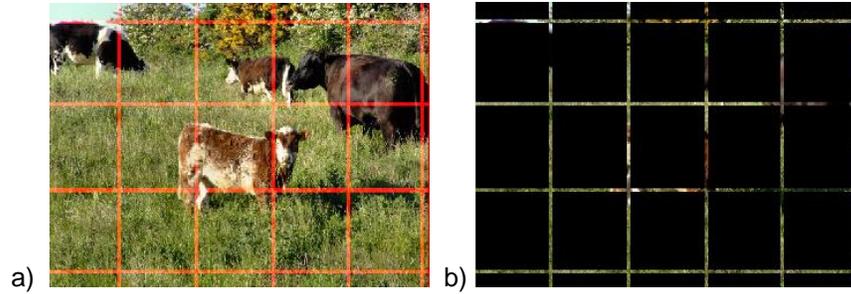


Figura 3.8: Aplicação de riscos: a) Imagem com riscos b) Máscara dos riscos

O terceiro método de distorção é a sobreposição de texto à imagem original. Isto é feito através da geração de uma imagem auxiliar no MATLAB onde se cria o texto a inserir (e.g., `image_aux = vision.TextInserter(imagem, cor do texto, tamanho do texto, localização)`). Em seguida o texto (variável “`image_aux`”) é “aplicada” na imagem original através da função `step(image_aux, imagem original)`, que retorna a imagem modificada, como é possível ver na Figura 3.9 alínea a).

O texto definido pelo utilizador na interface gráfica é assim aplicado em n posições diferentes (nos testes efetuados no contexto desta Dissertação $n = 5$). Estas posições são geradas aleatoriamente através da função `rand()`, tendo sempre em atenção a dimensão da imagem. No entanto, é de notar que a palavra nem sempre aparece o número de vezes especificado; por vezes o pixel inicial onde vai começar a ser “aplicado” o texto é um pixel das extremidades da imagem e isso implica que o texto ficará fora da imagem.

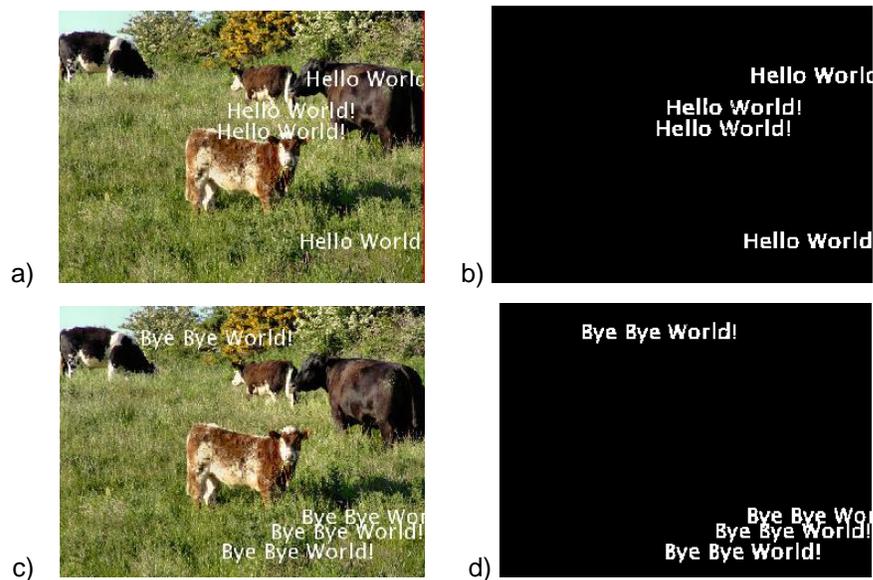


Figura 3.9: Aplicação de texto: a), c) Imagens com o texto aplicado; b), d) Máscaras com o texto

3.3 Módulo de Retocagem da Imagem

Como foi referido anteriormente, a aplicação de Retocagem Digital desenvolvida nesta Dissertação possui quatro algoritmos (Bertalmio, Criminisi, Lorenzi e Shen) dos quais foi reutilizado excertos de código e adaptado na aplicação) para aplicação de Retocagem Digital, que podem ser utilizados para remover objetos de uma imagem ou até mesmo corrigir uma região defeituosa.

3.3.1 Algoritmo FoE Model

A implementação do algoritmo FoE Model (Fields of Experts) integrado nesta aplicação é da autoria de Stefan Roch do Departamento das Ciências Computacionais da Universidade de Brown [23].

Em primeiro lugar, a imagem original é convertida para um espaço de cores YCbCr (a imagem inicial pode estar no formato jpg e bmp). Com esta conversão, a imagem de entrada passa a estar numa matriz de dimensão $M \times N \times 3$ que contém Y – luminância, Cb e Cr – crominâncias baseadas nas diferenças de azul e vermelho, onde a intensidade da luz é corrigido baseada nas componentes RGB.

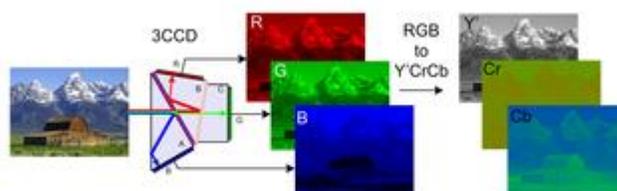


Figura 3.10: Exemplo conversão de RGB para YCbCr[28]



Figura 3.11: Exemplo imagem no espaço de cores YCbCr utilizado em vídeo e fotografias

Retocagem Digital

Em seguida é aplicado o filtro de gradiente sobre a imagem para melhorar. É aplicado em cerca de 2500 iterações (valor predefinido para obter o melhor resultado, isto é, a solução que obtém o melhor compromisso entre qualidade e velocidade de processamento) de maneira a tornar a convergência para a imagem final mais rápida.

Por último são feitas pequenas reparações na imagem e são limpos pequenos artefactos que possam ter permanecido na imagem, após a aplicação do filtro de gradiente. Essa limpeza é feita através da comparação de blocos de pixéis. Este passo é aplicado em cerca de 250 iterações (valor predefinido), Figura 3.12.



Figura 3.12: Exemplo imagem sem o sujeito no espaço de cores YCbCr final

Por último, volta-se a converter a imagem YCbCr para o formato RGB no qual é perceptível o resultado pretendido visualmente. Este processo é feito na gama de cores YCbCr por ser melhor para o tratamento de imagem em blocos e o desempenho computacional ser melhor.



Figura 3.13: Exemplo de imagem final em formato RGB restaurada pelo algoritmo FoE Model

3.3.2 Algoritmo Exemplar-based Retouching

A implementação do algoritmo “Exemplar-based Retouching” proposto por Criminisi [18] integrado nesta aplicação foi desenvolvida por Sooraj Bhat [24]. A imagem é reconstruída através de cópias de partes da imagem que se encontram a delimitar a região a ser preenchida, mantendo os detalhes da imagem, conseguindo um resultado final muito perto de uma imagem em que não é perceptível a olho nu na região corrigida.

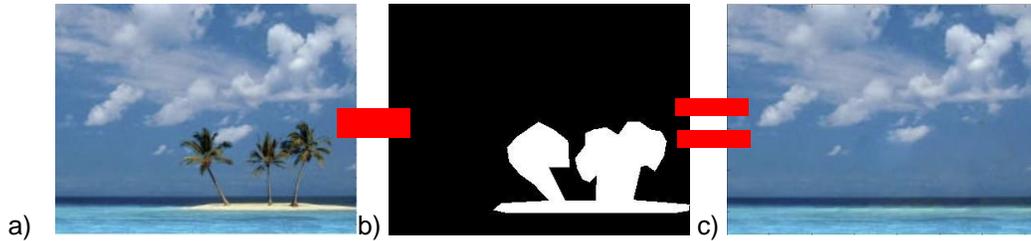


Figura 3.14: Remoção de objetos numa imagem usando o algoritmo de “Exemplar-based Retouching”: a) Imagem Original, b) Máscara de seleção, c) Imagem Restaurada

Da máscara é extraída a borda da área de Retocagem, sendo assim possível saber quais os pixels da imagem que se pretende preencher, Figura 3.15.



Figura 3.15: Exemplo borda extraída através da máscara

É aplicado um fator de confiança à informação do pixel. Esse fator vai diminuindo à medida que se afasta da borda delimitadora da região e se aproxima do centro da região, como é possível observar na Figura 3.16, através dos níveis de cinzentos, que são mais claros (maior confiança) na borda e vão ficando mais escuros (menor confiança) no centro.

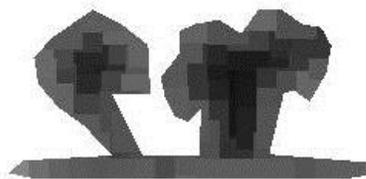


Figura 3.16: Exemplo da variação do fator de confiança ao longo da área selecionada

O método aplicado foi referido no Capítulo 2, a reconstrução é possível sabendo a propagação da informação dos pixels de uma área da imagem original que representa melhor a área que se pretende recuperar.

3.3.3 Algoritmo Total Variation Image Retouching

Implementação do algoritmo “Total Variation Image Retouching” proposto por Shen, foi desenvolvida por Pascal Gereuer [24].

Neste algoritmo começa-se por verificar as variáveis iniciais recebidas, como a imagem original, a máscara de seleção o valor do parâmetro lambda, etc. O parâmetro lambda serve para controlar a propagação do ruído fora da zona de Retocagem selecionada, na aplicação o lambda pode variar entre $1 \cdot \exp(0)$ e $1 \cdot \exp(10)$ que são valores aceitáveis para obtenção de resultados e respetivas comparações. O utilizador tem uma opção na aplicação para variar este valor.

É gerada uma estrutura que vai conter toda a informação (nessa estrutura existe um parâmetro que especifica o modelo de ruído, por omissão é considerado ruído *Gaussiano* e Imagem original), antes de ser aplicada a reconstrução da imagem. Após esta fase é efetuada a restauração final.



Figura 3.17: Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching”
(Lambda = $1 \cdot \exp(4)$)

Retocagem Digital



Figura 3.18: Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching”
(Lambda = $1 \cdot \exp(1)$)



Figura 3.19: Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching”
(Lambda = $1 \cdot \exp(2)$)



Figura 3.20: Exemplo do ciclo de reconstrução do algoritmo “Total Variation Image Retouching”
(Lambda = $1 \cdot \exp(3)$)

Como é possível ver nos exemplos anteriores presentes da Figura 3.17 até Figura 3.20 com a variação do lambda a força de desfocagem da área que não está selecionada para Retocagem vai ser menor, quanto maior for o valor de lambda. A partir do lambda igual a 1000 ($1 \cdot \exp(3)$), para valores acima, não se nota grande diferença a olho nu.

3.3.4 Algoritmo Coherence Sensitive Hashing

A implementação do algoritmo “Coherence Sensitive Hashing Algorithm” baseada na fórmula de Lorenzi que é proveniente da técnica de Criminisi. Esta função em MATLAB foi desenvolvida por Simon Korman e Shai Avidan [25].

Neste algoritmo, em primeiro lugar, é extraída da máscara de seleção a primeira componente, pois esta imagem é RGB.

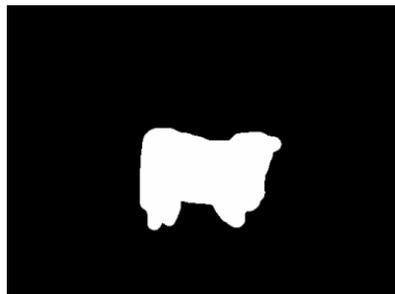


Figura 3.21: Exemplo de máscara com a informação da primeira componente de cores usada no Algoritmo “Coherence Sensitive Hashing”

Simultaneamente, é determinada a escala inicial para o início do processo de reconstrução, sendo depois aplicado um redimensionamento da imagem original e da máscara, para ambas ficarem com a mesma dimensão. É aplicado um ciclo de iterações (neste caso 30 iterações e a cada iteração é calculado o bloco vizinho mais próximo entre as imagens original e a máscara).

Por último é feita a junção dos blocos de pixéis obtidos anteriormente para criar a nova imagem, passando pela conversão para o formato RGB e redimensionamento original.



Figura 3.22: Redimensionamento da Imagem Restaurada (4 iterações espaçadas para se notar a diferença de tamanho)



Figura 3.23: Resultado final após a conversão para RGB usando o no Algoritmo “Coherence Sensitive Hashing”

3.4 Módulo de Avaliação da Qualidade da Imagem Final

Como foi referido anteriormente no Capítulo 2, as métricas existentes para avaliação da qualidade de imagem, podem ser divididas em três tipos: i) “No-Reference”(NR), ii) “Full Reference”(FR) e iii) “Reduced-Reference”(RR) [14].

- As métricas NR são as métricas em que é apenas utilizada a imagem final obtida no processo de Retocagem. Como não se usa qualquer informação sobre a imagem original, estas métricas apresentam algumas limitações na previsão da qualidade da imagem final.
- As métricas FR usam como referência a imagem original que é introduzida na aplicação e também a imagem final.
- As métricas RR usam apenas um conhecimento parcial da imagem original para tentar estimar a qualidade da imagem final distorcida (neste caso não foi implementada nenhuma métrica).

3.4.1 Técnicas No-Reference

- **Blind Reference less Image Spatial Quality Evaluator (BRISQUE)[14]** – Para o cálculo desta métrica foi utilizada uma aplicação desenvolvida por Anish Mitta [27]. Esta métrica usa os valores de coeficiente de luminância para quantificar as possíveis perdas de “realidade” da imagem. As outras características utilizadas no cálculo derivam da luminância.
- **Blind Image Quality Index (BIQI)[16]** – Para o cálculo desta métrica foi utilizada uma aplicação desenvolvida por Anush Krishna Moorthy [27]. Sendo uma métrica NR só utiliza a informação da imagem distorcida da qual divide a informação em pequenos blocos de pixéis.

Estas duas métricas (No-Reference) variam entre 0 e 100, sendo que os valores mais próximos de 0 correspondem a melhor qualidade da imagem final obtida.

3.4.2 Técnicas Full-Reference

- **Mean Squared Error (MSE)[17]** – Nesta métrica é calculada uma média quadrática entre a imagem original e a imagem final. Esse cálculo é obtido para cada pixel, somando as diferenças ao quadrado de todos os pixéis (pixel por pixel) e dividindo pelo número total de pixéis. Quando o menor for o valor de MSE mais próximo vai estar a imagem distorcida da imagem inicial;

- **Structural Similarity Index (SSIM)[20][21]** – foi utilizada uma aplicação desenvolvida por Zhou Wang [20] quando maior for o valor de SSIM, isto é, o SSIM varia entre 0 e 1, quanto mais próximo estiver do valor 1, mais próxima a imagem final vai estar da imagem original.
- **Peak Signal-to-Noise Ratio (PSNR)[17]** – Quanto mais elevado for o valor de PSNR (dB) de uma imagem, mais próximo vai estar a imagem final da imagem original. Utilizando a medida MSE.

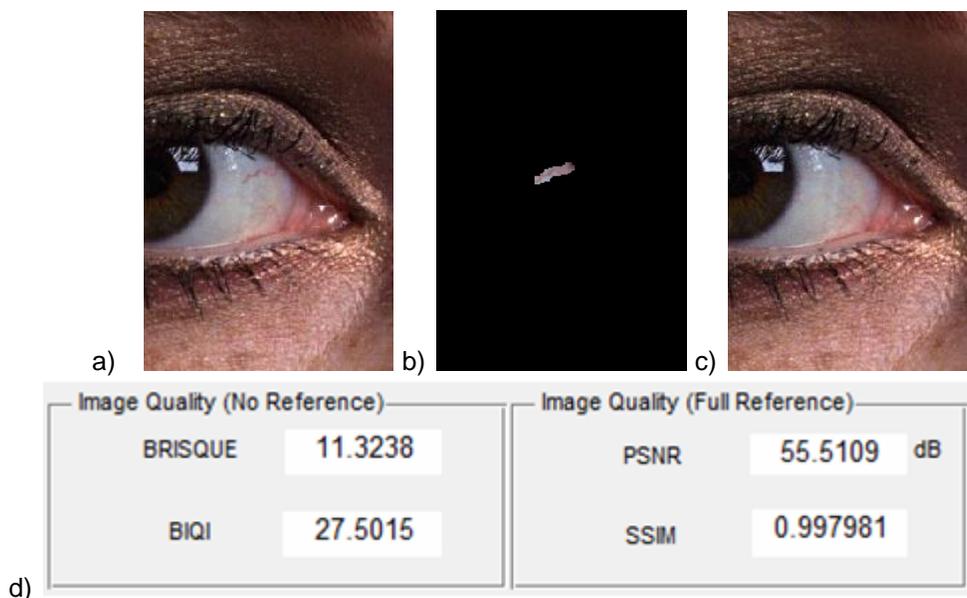


Figura 3.24: Exemplo de Retocagem de uma imagem e respetivas métricas de qualidade a) Imagem Original b) Máscara c) Imagem Reconstruída d) Valores de qualidade obtidos

Capítulo 4

Resultados e Discussão

Neste capítulo vão ser apresentados vários resultados de testes diferentes a fim de podermos analisar os níveis de restauro obtidos e qual o melhor algoritmo para restauração de imagens.

4.1 *Condições de Teste*

As imagens utilizadas nesta aplicação estão armazenadas numa pasta, como se ilustra na Figura 4.1.

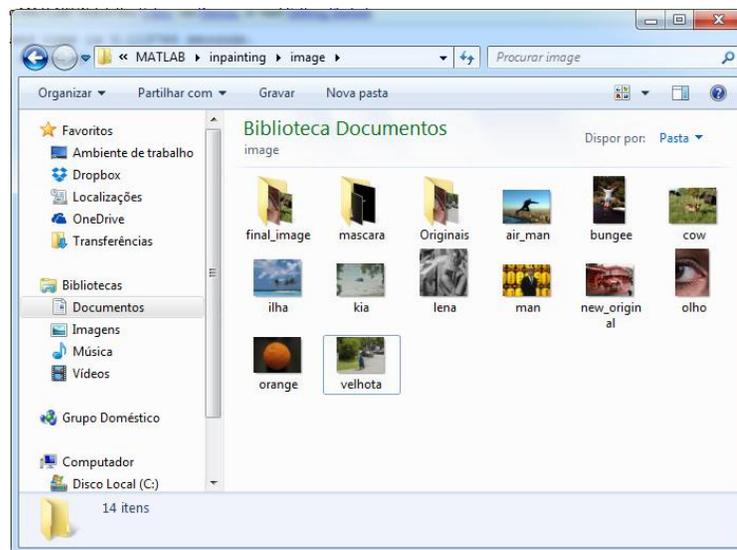


Figura 4.1: Pasta das imagens utilizadas na aplicação

A base de dados usada no contexto desta Dissertação é composta por oito imagens. As imagens mais utilizadas nos capítulos anteriores para as diversas demonstrações são as seguintes: *cow.png*, *man.png*, *bungee.png*, *olho.png* e a *new_original.png*.

As máscaras de seleção estão guardadas na pasta “mascara” e no caso das distorções as máscaras criadas são guardadas nesta mesma pasta.

Retocagem Digital

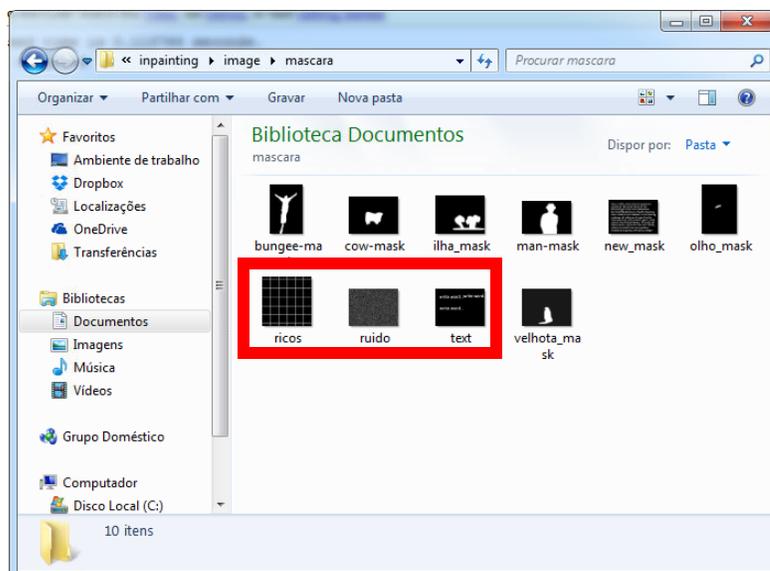


Figura 4.2: Pasta das máscaras utilizadas na aplicação

Neste caso, as imagens que estão dentro do retângulo vermelho são as máscaras que foram geradas pela aplicação e que assim poderão voltar a ser usadas posteriormente na aplicação sem ser necessário gerar uma nova distorção, mas sim aplicar através do botão *load mask*.

Na pasta Originais estão armazenadas as imagens que correspondem a imagens originais mas com o(s) objeto(s) removido(s) para se poderem usar as métricas FR, e.g., SSIM e PSNR. Por último, na pasta *final_image* são armazenadas as imagens restauradas.

Para obter resultados para análise vão ser utilizadas seis imagens apresentadas na Figura 4.3.

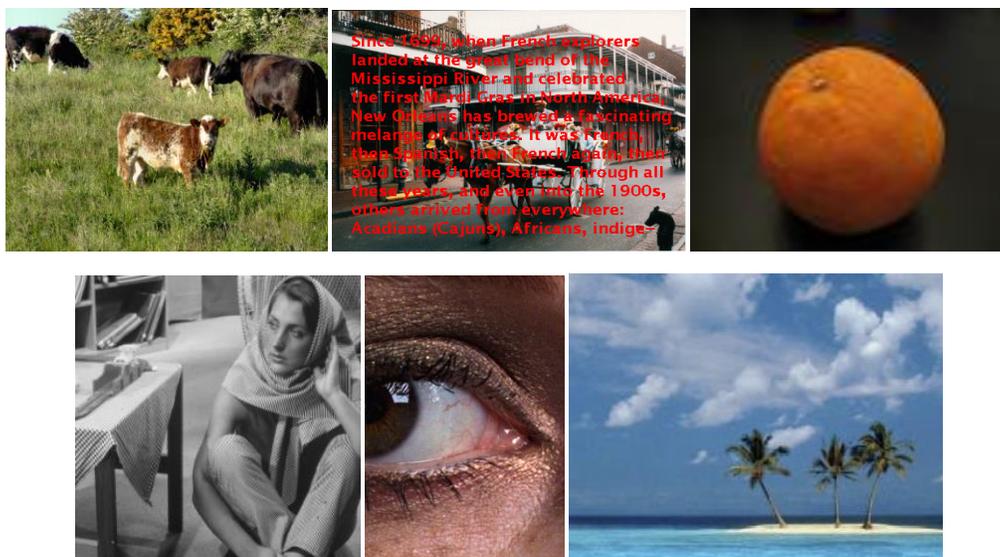


Figura 4.3: Seis imagens de teste utilizadas (cow, new_original, Orange, lena, olho, ilha)

Na Figura 4.4 apresentam-se as máscaras usadas no modo *load mask*: para as imagens da Figura 4.4 a) e Figura 4.4 b).

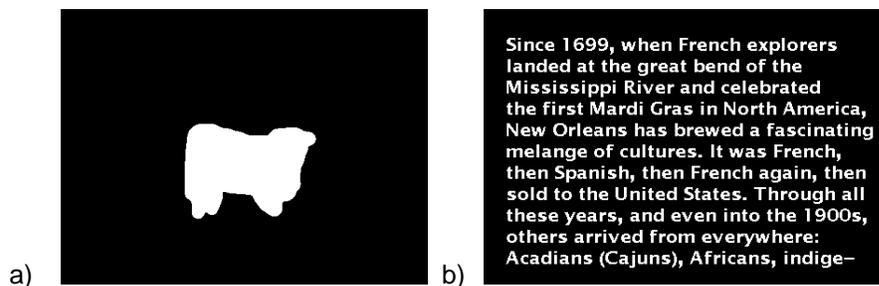


Figura 4.4: Máscaras utilizadas para os testes a) e b)

4.2 Resultados

4.2.1 Teste 1: Remoção de um Derrame no Olho

Remoção de um derrame no olho, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

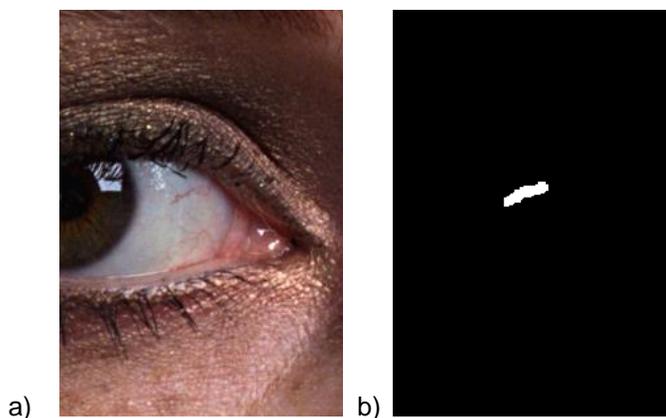


Figura 4.5: Teste 1 - Remoção de um Derrame no Olho a) Imagem Original b) Máscara

Algoritmo FoE Model:

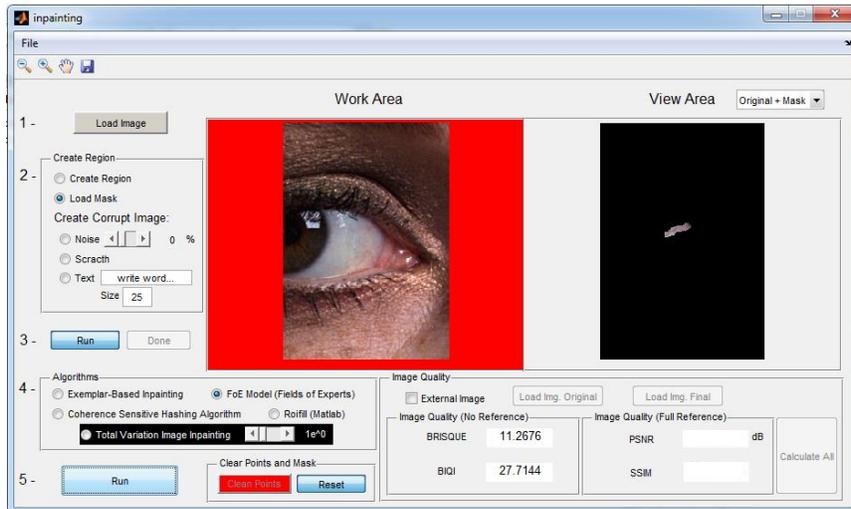


Figura 4.6: Teste 1 - Remoção de um Derrame no Olho (Algoritmo FoE Model)

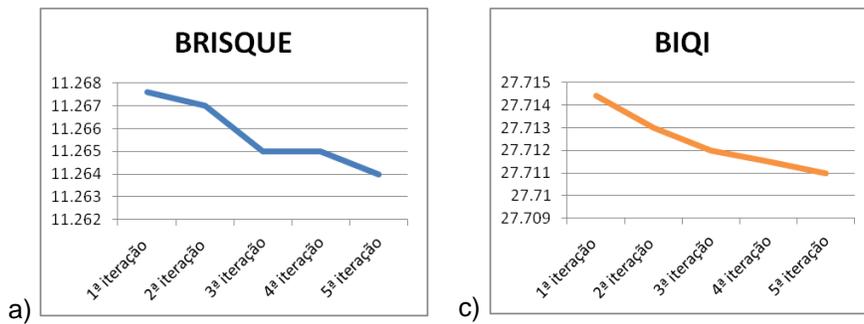


Figura 4.7: Gráficos das Métricas de Qualidade para FoE Model a) BRISQUE b) BIQI

Neste caso a progressão da qualidade da imagem é muito nítida, conseguindo obter-se o melhor nível de qualidade de imagem perto da quarta iteração em quase todas as métricas, tirando o caso de BRISQUE que graficamente teve um melhoramento mais perceptível na passagem da quarta para a quinta iteração. Para a quinta iteração obtém-se: BRISQUE = 11.26 e para BIQI = 27.71.

Algoritmo Exemplar-Based Retouching:

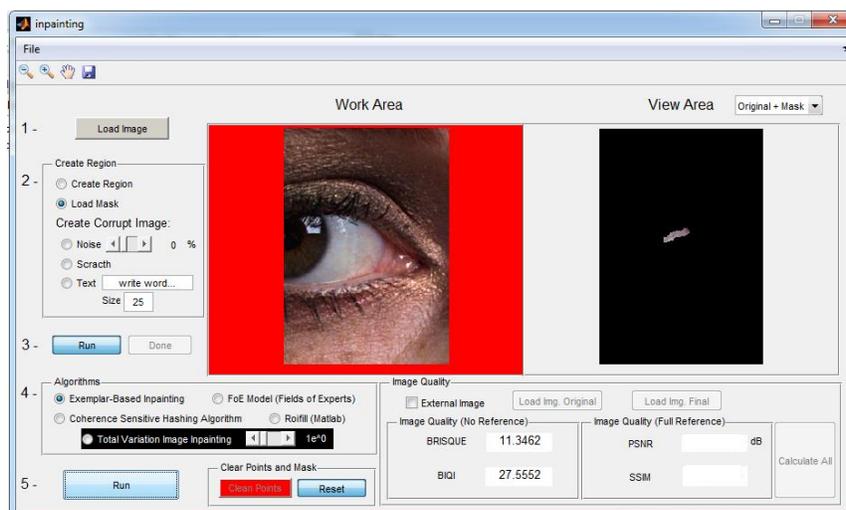


Figura 4.8: Teste 1 - Remoção de um Derrame no Olho (Algoritmo Exemplar-Based Retouching)

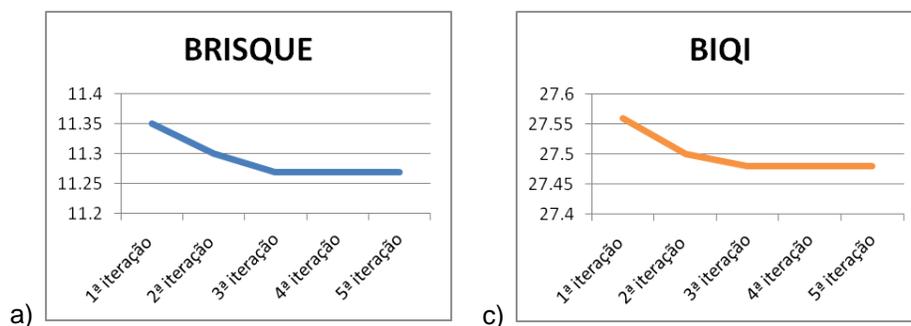


Figura 4.9: Gráficos das Métricas de Qualidade para Exemplar-Based Retouching a) BRISQUE b) BIQI

É possível concluir que para este algoritmo ao fim da terceira iteração o resultado já não se altera mais, obtendo assim o melhor valor. Para a quinta iteração obtém-se: BRISQUE = 11.27 e para BIQI = 27.48.

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

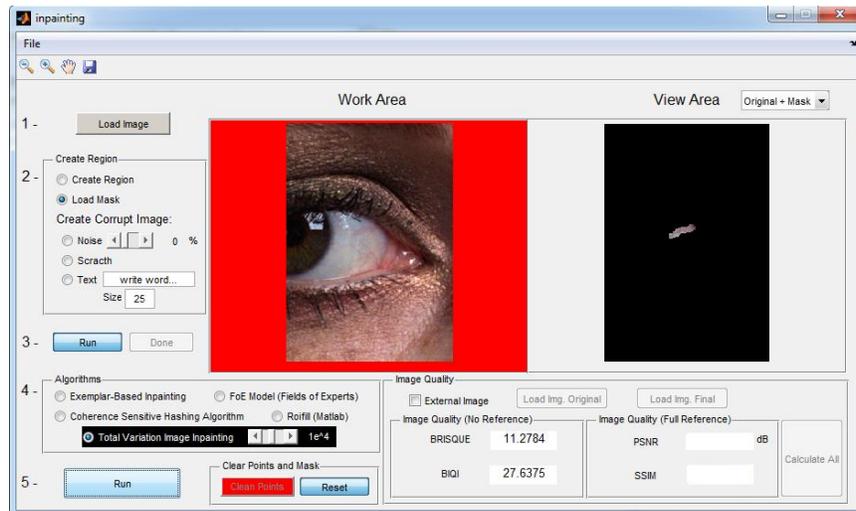


Figura 4.10: Teste 1 - Remoção de um Derrame no Olho (Algoritmo Total Variation Image Retouching)

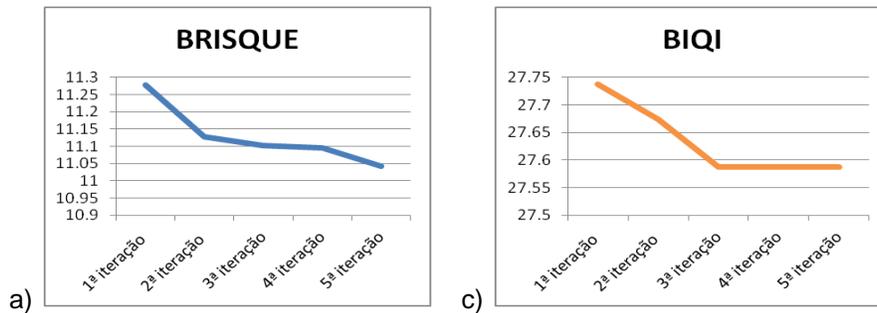


Figura 4.11: Gráficos das Métricas de Qualidade para Total Variation Image Retouching a) BRISQUE b) BIQI

Neste algoritmo ao fim da quinta iteração ainda não se obteve um valor de estagnação, a não ser na métrica BIQI que ao fim da terceira iteração obteve um valor que não veio a ser mais alterado até à quinta iteração. Existindo sempre um melhoramento nas medidas de qualidade a cada iteração. Para a quinta iteração obtém-se: BRISQUE = 11.04 e para BIQI = 27.58.

Algoritmo Coherence Sensitive Hashing Algorithm:

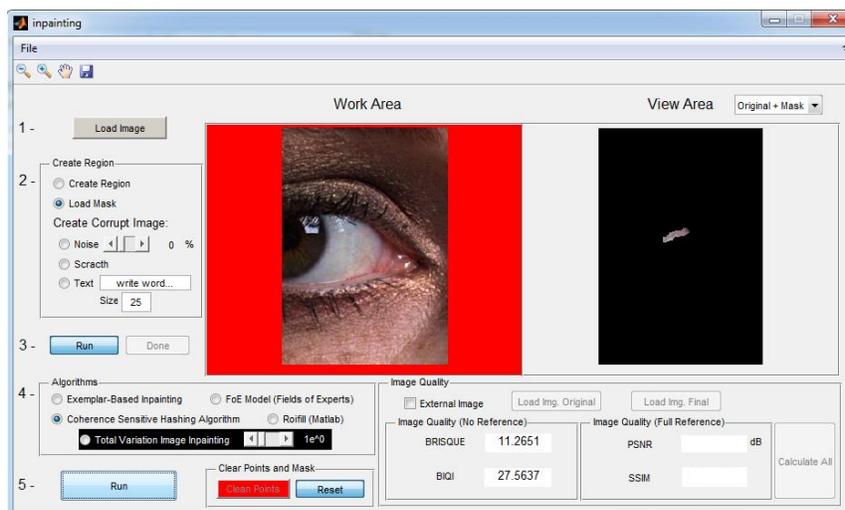


Figura 4.12: Teste 1 - Remoção de um Derrame no Olho (Algoritmo Coherence Sensitive Hashing Algorithm)

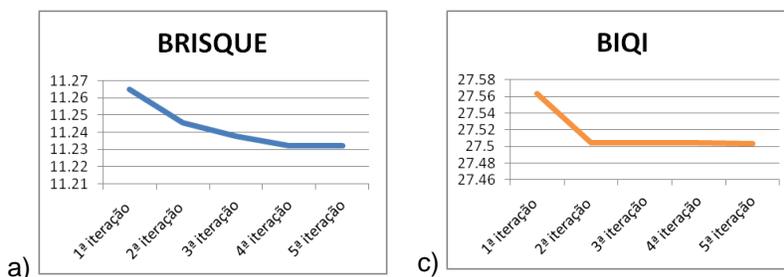


Figura 4.13: Gráficos das Métricas de Qualidade para Coherence Sensitive Hashing Algorithm a) BRISQUE b) BIQI

As métricas No-Reference só à quarta iteração é que se obtém um valor menor que se mantém para a quinta iteração. Para a quinta iteração obtém-se: BRISQUE = 11.23 e para BIQI = 27.50.

Tabela 1: Teste 1 - Remoção de um Derrame no Olho

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	-	-	-	-
	SSIM	-	-	-	-
No-Reference	BRISQUE	11.30	11.34	11.28	11.27
	BIQI	27.71	27.56	27.64	27.56

Neste primeiro teste a olho nu os piores algoritmos são: Total Variation Image Retouching e em Coherence Sensitive Hashing Algorithm onde se nota a zona restaurada. Nas medidas No-Reference os resultados são muito idênticos, mas quanto mais próximo de 0 melhor é a qualidade e assim sendo os algoritmos Total Variation Image Retouching e em Coherence Sensitive Hashing Algorithm obtêm um menor valor de BRISQUE.

Para este primeiro teste foram executadas 5 iterações para cada algoritmo, mantendo sempre a mesma máscara, no sentido de se poder ver a evolução e melhoria em cada iteração ao nível das métricas de qualidade das imagens que são obtidas em cada iteração.

Denomina-se por iteração, o método manual que consiste na execução do algoritmo, sobre o resultado obtido anteriormente, por diversas vezes consecutivas.

4.2.2 Teste 2: Remoção de uma Pessoa da Imagem

Remoção de uma pessoa na imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

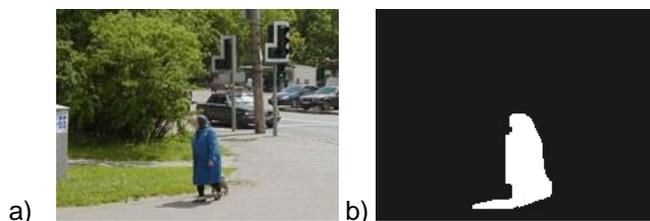


Figura 4.14: Teste 2 - Remoção de uma Pessoa da Imagem a) Imagem Original b) Máscara de seleção

Algoritmo FoE Model:

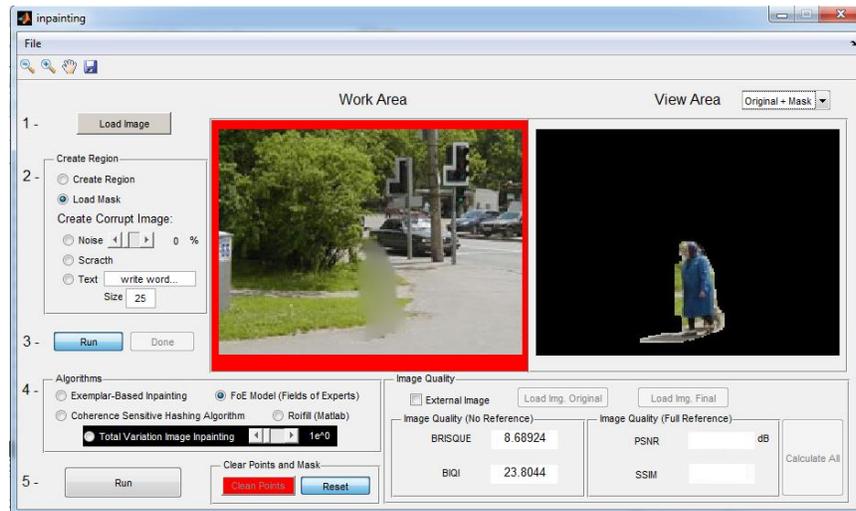


Figura 4.15: Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

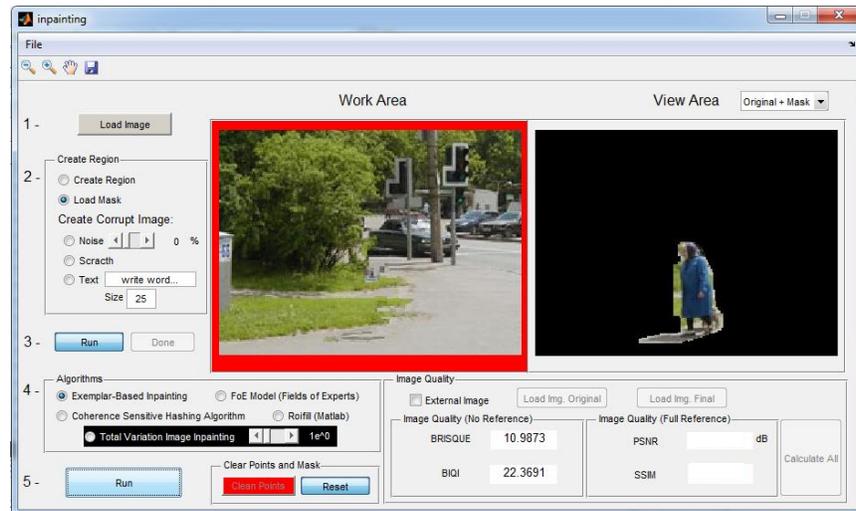


Figura 4.16: Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

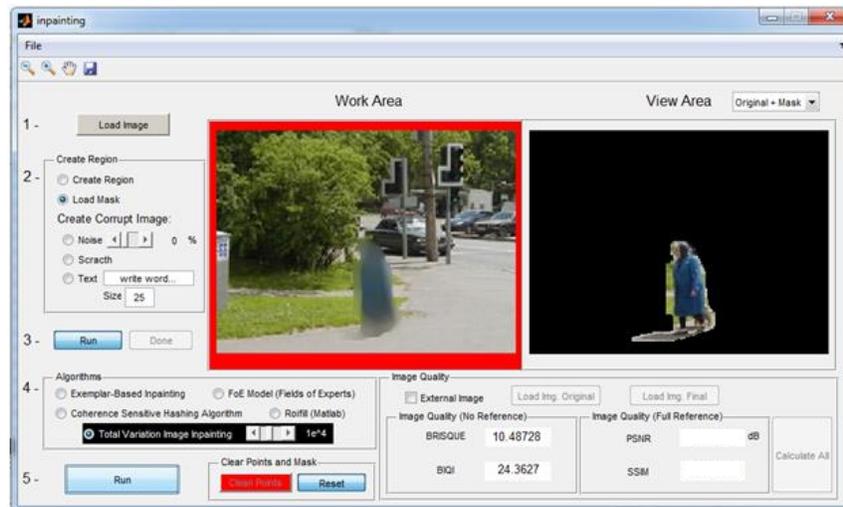


Figura 4.17: Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

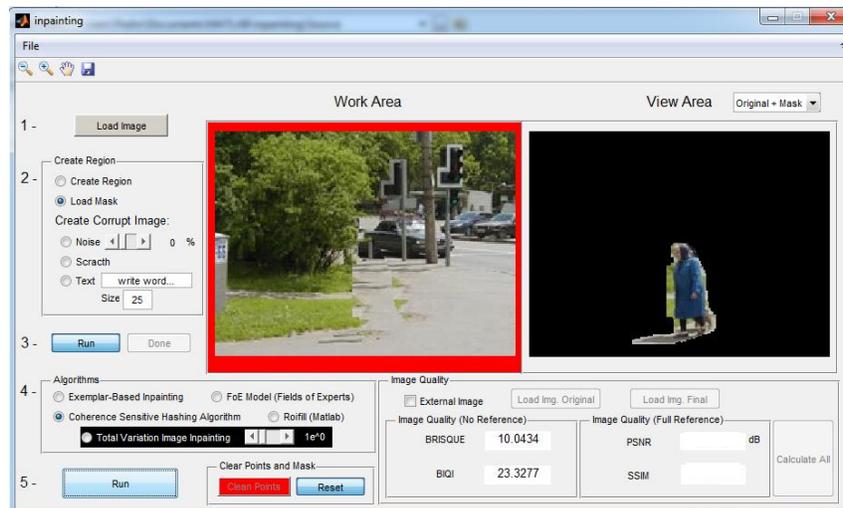


Figura 4.18: Teste 2 - Remoção de uma Pessoa da Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 2: Teste 2 - Remoção de uma Pessoa da Imagem

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	-	-	-	-
	SSIM	-	-	-	-
No-Reference	BRISQUE	8.69	10.99	10.49	10.04
	BIQI	23.80	22.37	24.36	23.33

As medidas NR o resultado final do algoritmo de FoE Model é melhor na medida de BRISQUE mas o de Exemplar-Based Retouching é melhor em termos de BIQI. Visto ser uma área considerável para restauro e os algoritmos não terem a mesma percepção que os nossos olhos os resultados são aceitáveis, pois fixou os melhores resultados nos algoritmos que realmente também obtiveram um melhor resultado.

4.2.3 Teste 3: Remoção de Texto Sobreposto numa Imagem

Remoção do texto da imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

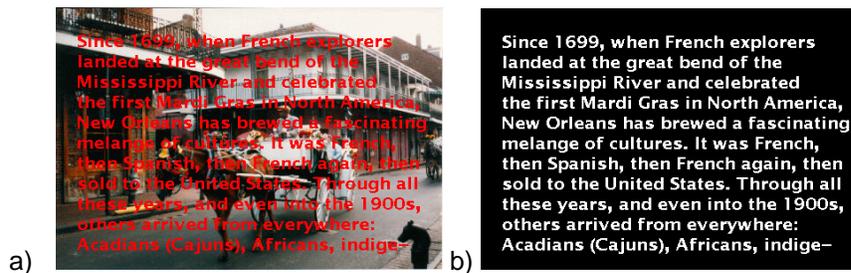


Figura 4.19: Teste 3 - Remoção de Texto Sobreposto numa Imagem a) Imagem Original b) Máscara

Algoritmo FoE Model:

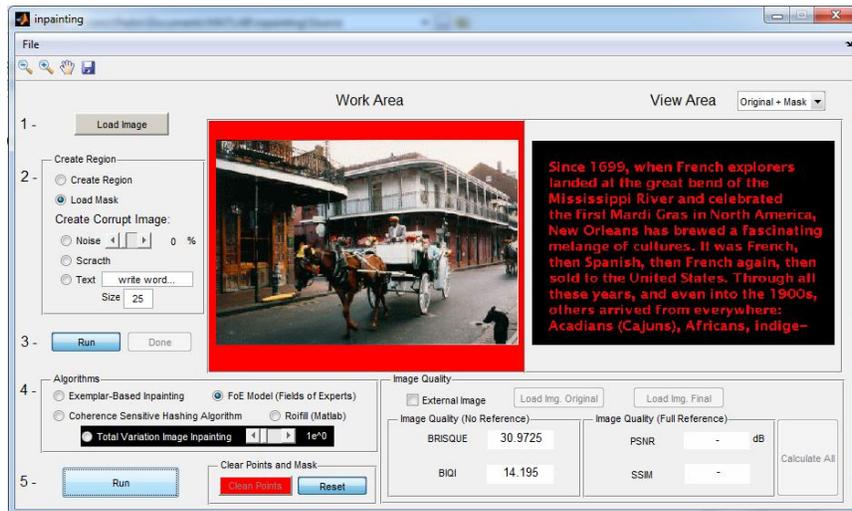


Figura 4.20: Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

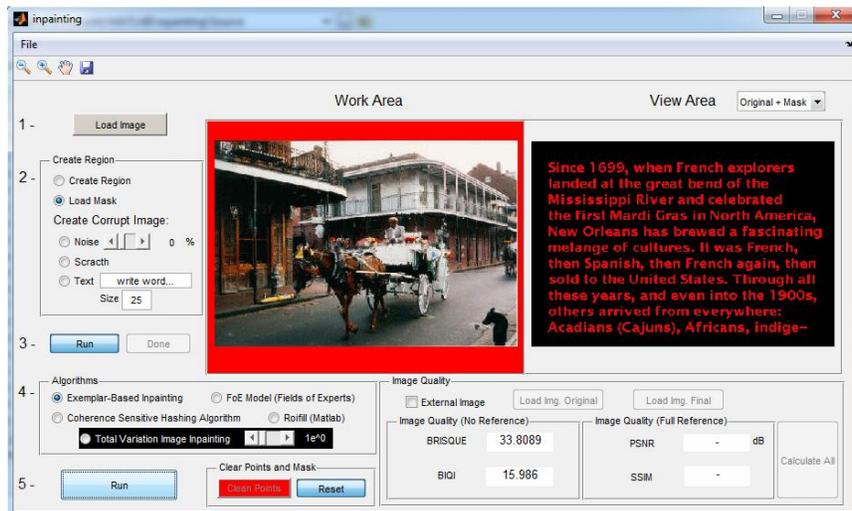


Figura 4.21: Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

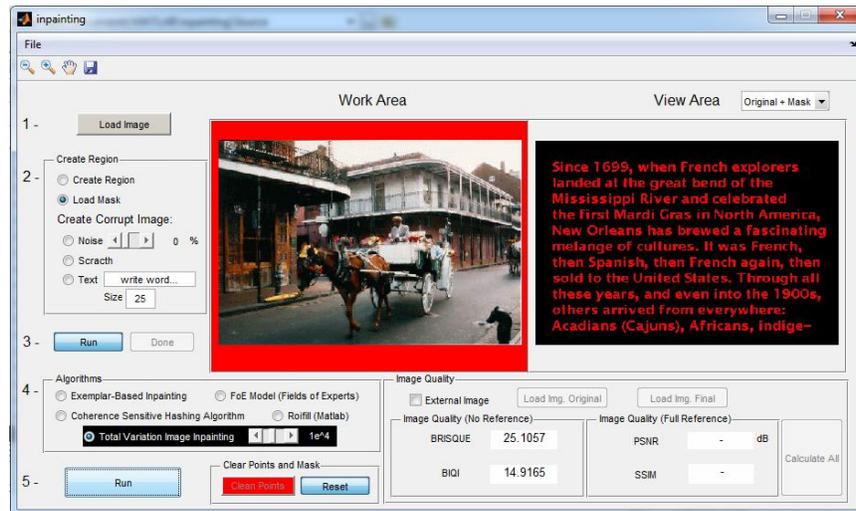


Figura 4.22: Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

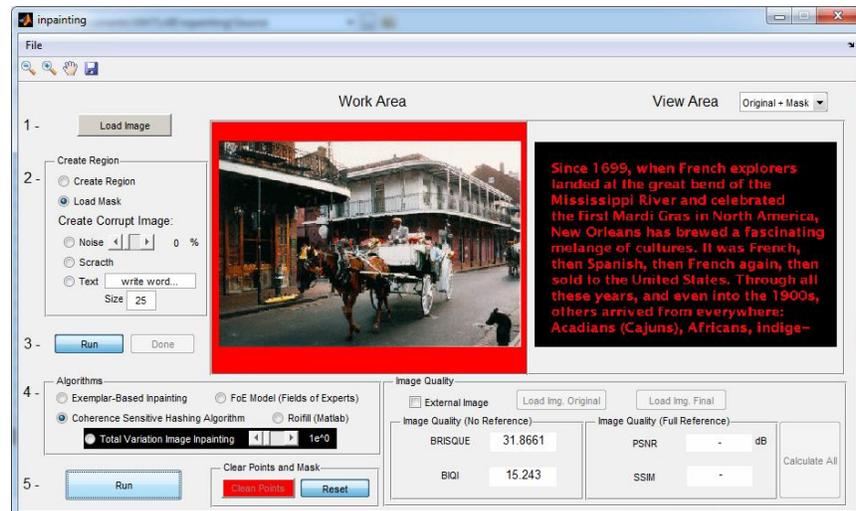


Figura 4.23: Teste 3 - Remoção de Texto Sobreposto numa Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 3: Teste 3 - Remoção de Texto Sobreposto numa Imagem

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	-	-	-	-
	SSIM	-	-	-	-
No-Reference	BRISQUE	30.97	33.81	25.11	31.87
	BIQI	14.20	16.00	14.92	15.24

Para este teste não existe a imagem inicial sem defeito e por isso não é possível obter o valor das métricas de qualidade FR.

Em termos visuais não é perceptível qual o melhor resultado, em termos métricos de uma forma geral o que obteve melhores resultado foi o algoritmo Total Variation Image Retouching. Apesar de o algoritmo FoE Model ter obtido um melhor resultado na métrica de BIQI mas um valor pior na métrica de BRISQUE o que não é muito quarente.

4.2.4 Teste 4: Aplicação e Remoção de Ruído Salt & Pepper (2% e 4%)

Remoção do ruído de 2% da imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

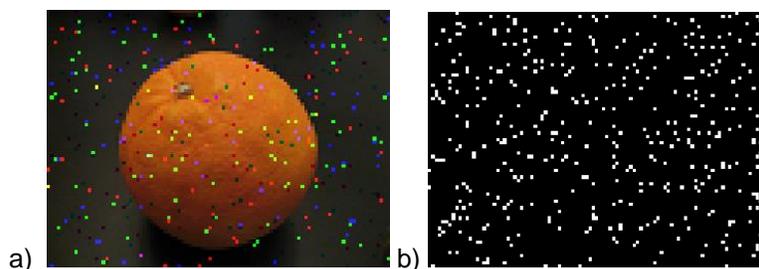


Figura 4.24: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%) a) Imagem Original b) Máscara

Algoritmo FoE Model:

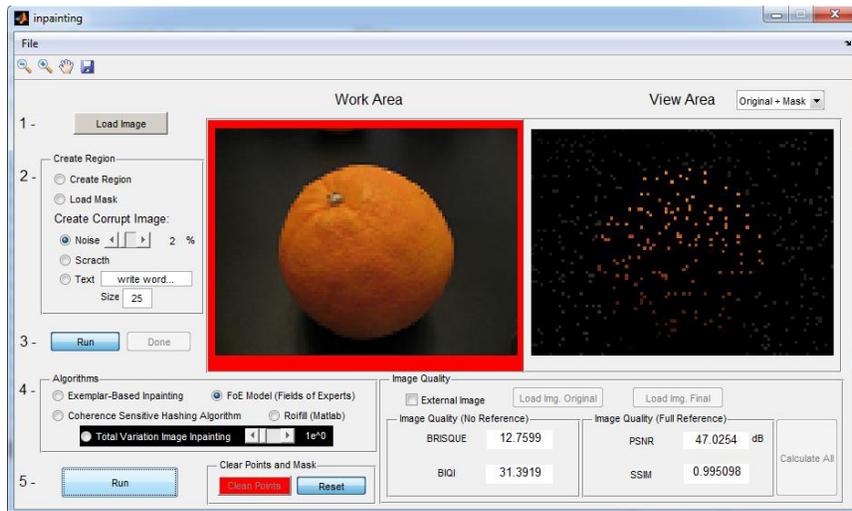


Figura 4.25: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo FoE Mode)

Algoritmo Exemplar-Based Retouching:

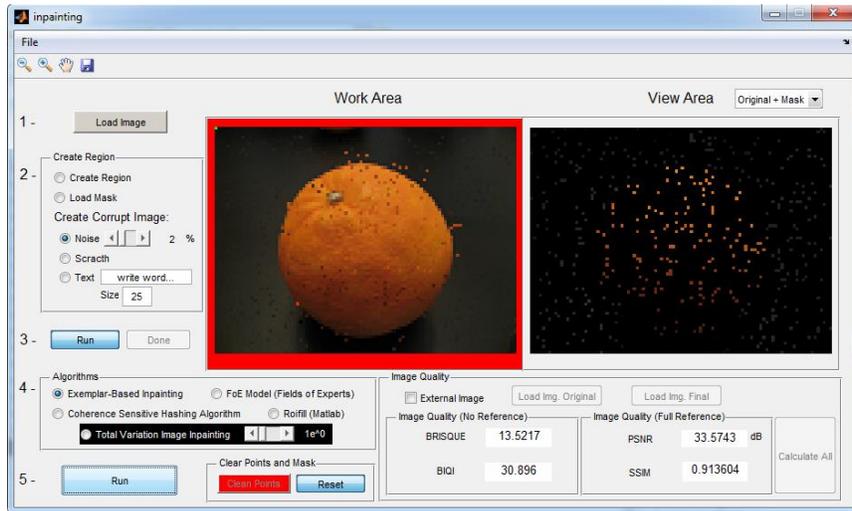


Figura 4.26: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

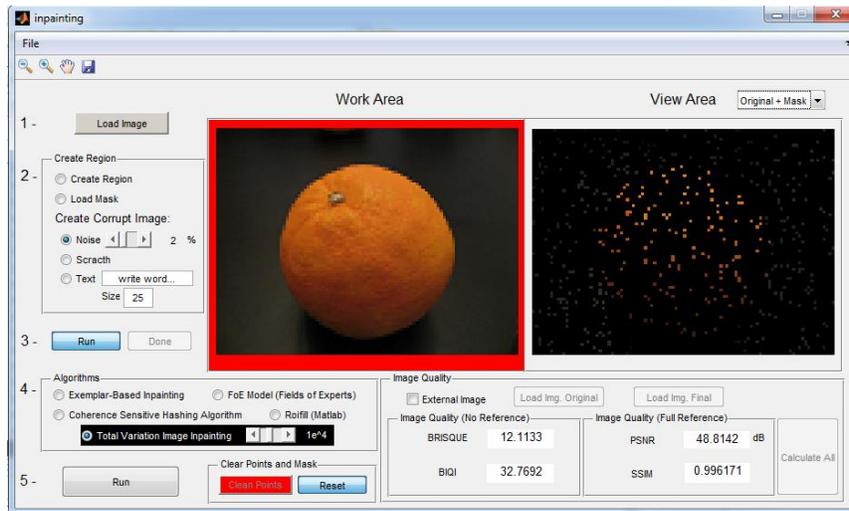


Figura 4.27: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

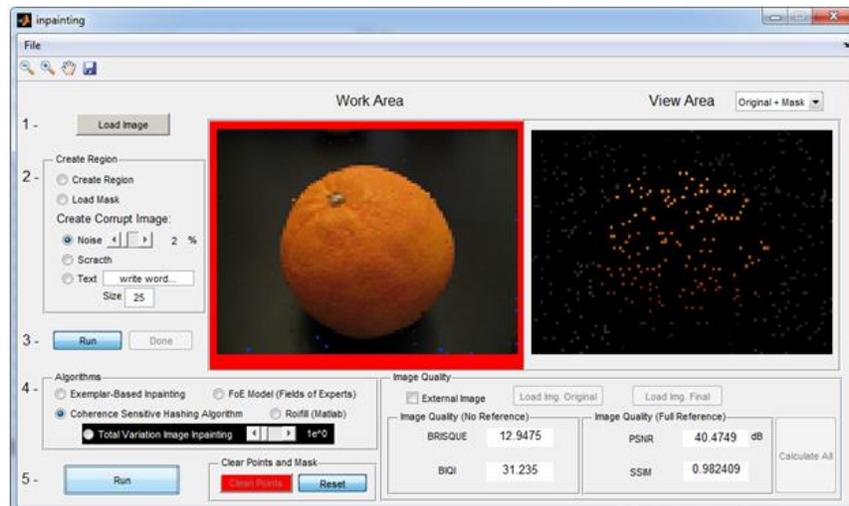


Figura 4.28: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)(Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 4: Teste 4 - Aplicação e Remoção de Ruído Salt & Pepper (2%)

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	47.03	33.57	48.81	40.48
	SSIM	0.9951	0.9136	0.9962	0.9824
No-Reference	BRISQUE	12.76	12.52	12.11	12.95
	BIQI	31.39	30.90	32.77	31.24

Na análise destes resultados, as métricas FR são coerentes, i.e., as medidas de PSNR e SSIM são coerentes com o resultado visual em que as melhores imagens são as dos algoritmos FoE Model e em Total Variation Image Retouching. Nas métricas NR mantém-se mais ao menos o mesmo critério, mas colocando a de Exemplar-Based Retouching igualmente boa em termos da métrica de BRISQUE. Para a métrica de BIQI o melhor resultado é o de Exemplar-Based Retouching, que não tem o menor sentido, pois visualmente é o pior resultado e até obtém o pior resultado também nas medidas FR.

Remoção do ruído de 4% da imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

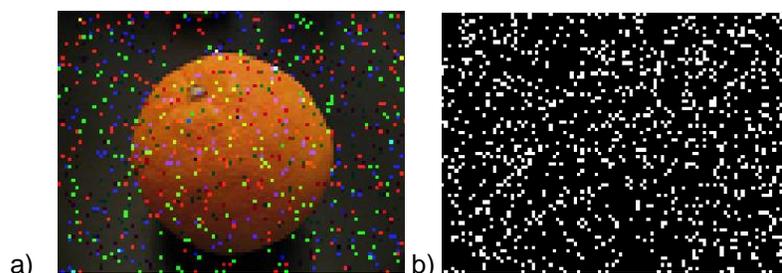


Figura 4.29: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) a) Imagem Original b) Máscara

Algoritmo FoE Model:

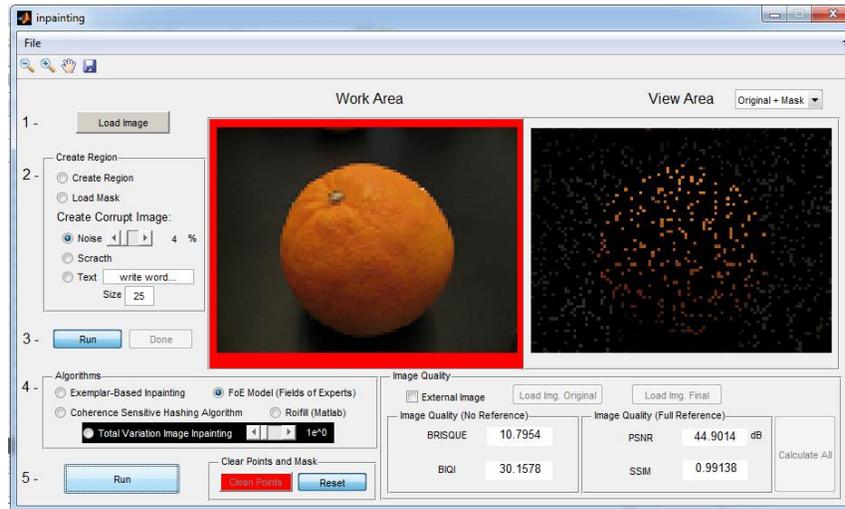


Figura 4.30: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

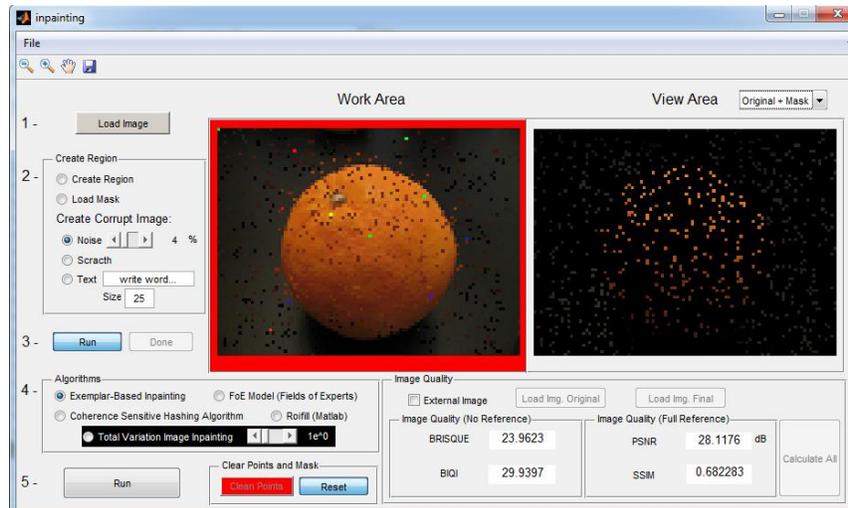


Figura 4.31: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

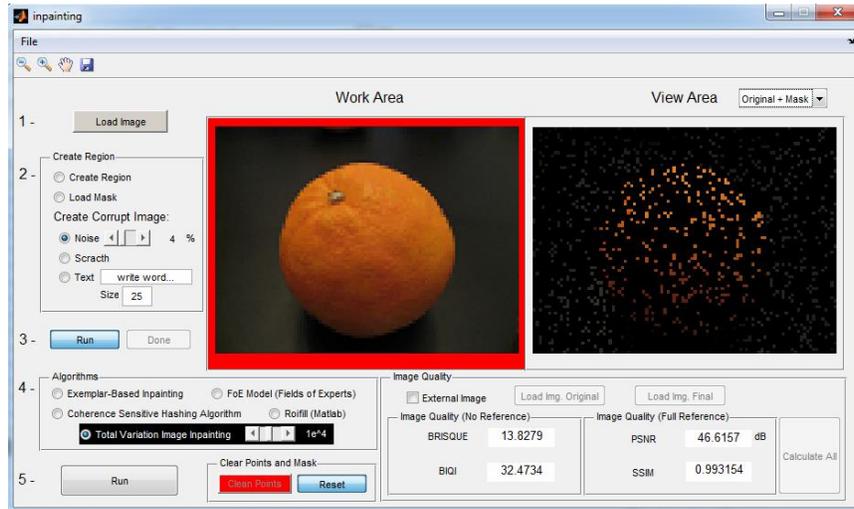


Figura 4.32: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

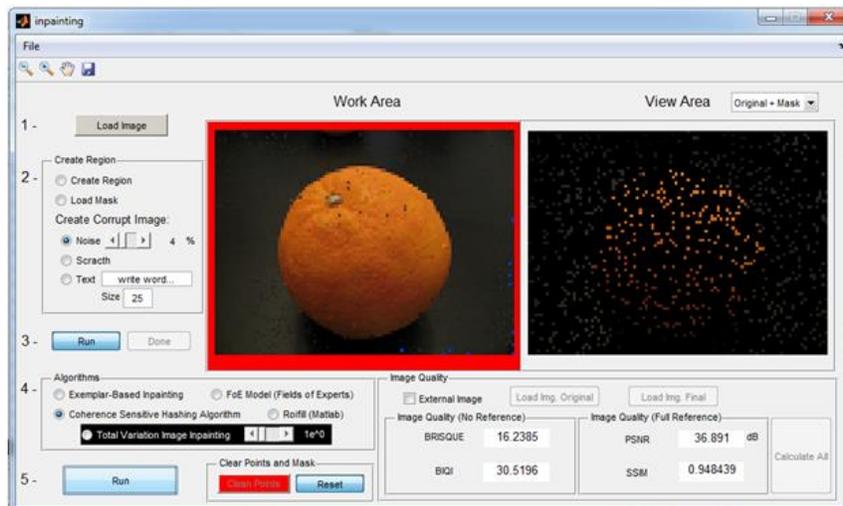


Figura 4.33: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%) (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 5: Teste 5 - Aplicação e Remoção de Ruído Salt & Pepper (4%)

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	44.90	28.12	46.62	36.89
	SSIM	0.9914	0.6823	0.9932	0.9484
No-Reference	BRISQUE	10.80	23.96	13.83	16.24
	BIQI	30.16	29.94	32.47	30.52

Neste teste que é idêntico ao anterior sofrendo apenas a alteração de uma variável inicial no cálculo da imagem, sendo essa alteração na geração do ruído passando de 2% para 4%. Neste sentido é possível verificar que os resultados obtidos são do mesmo género dos do teste anterior. As métricas FR são coerentes, i.e., as medidas de PSNR e SSIM são coerentes com o resultado visual em que as melhores imagens são as dos algoritmos FoE Model e em Total Variation Image Retouching. Nas medidas NR mantém-se mais ao menos o mesmo critério, mas colocando a de Exemplar-Based Retouching igualmente boa em termos da métrica de BIQI, sendo o melhor, que não tem o menor sentido, pois visualmente é o pior resultado e até obtém o pior resultado também nas medidas FR.

4.2.5 Teste 6: Aplicação e Remoção de Riscos numa Imagem

Remoção de riscos (horizontais e verticais) da imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

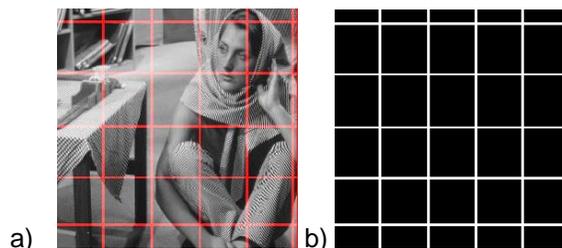


Figura 4.34: Teste 6 - Aplicação e Remoção de Riscos numa Imagem a) Imagem Original b) Máscara

Retocagem Digital

Algoritmo FoE Model:

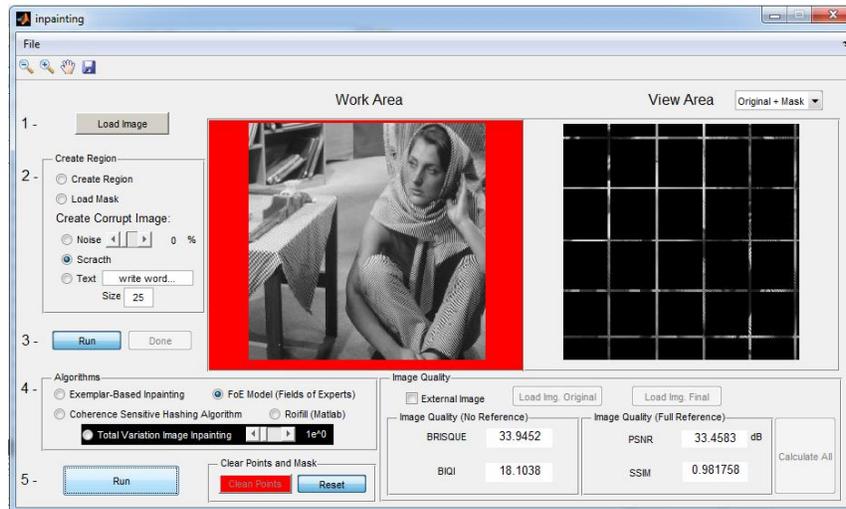


Figura 4.35: Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

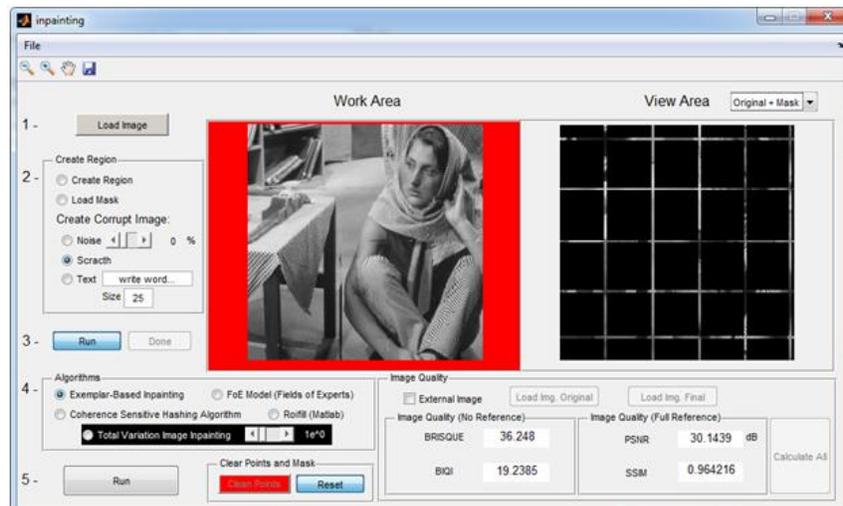


Figura 4.36: Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

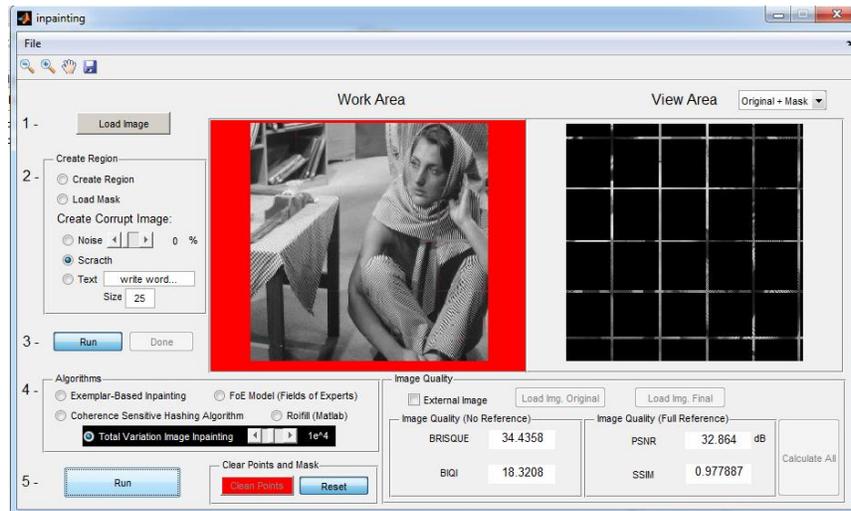


Figura 4.37: Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

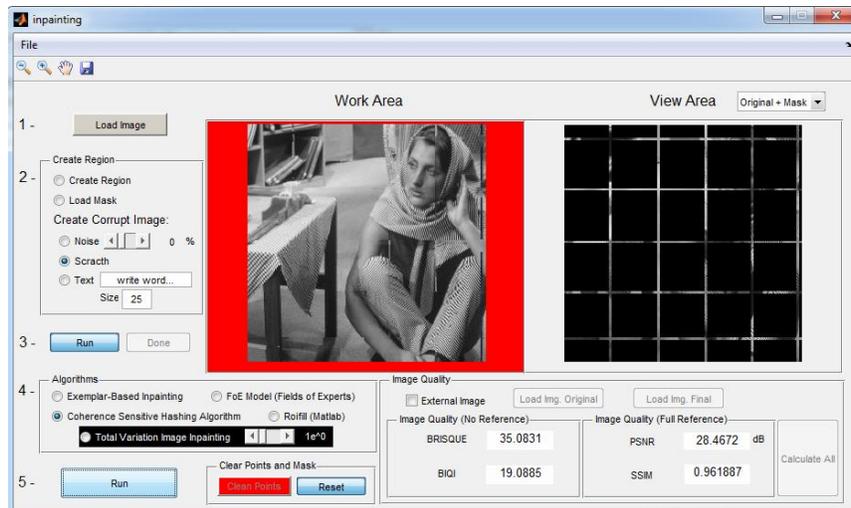


Figura 4.38: Teste 6 - Aplicação e Remoção de Riscos numa Imagem (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 6: Teste 6 - Aplicação e Remoção de Riscos numa Imagem

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	33.95	30.14	32.86	28.47
	SSIM	0.9818	0.9642	0.9779	0.9619
No-Reference	BRISQUE	33.46	36.25	34.44	35.08
	BIQI	18.10	19.24	18.32	19.09

Neste teste os melhores resultados visuais são os dos algoritmos FoE Model e Total Variation Image Retouching, em termos de métricas objetivas verifica-se o mesmo resultado tanto para as métricas FR como para as métricas NR.

4.2.6 Teste 7: Aplicação e Remoção de Texto numa Imagem

Remoção de texto (tamanho de letra 25) da imagem, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

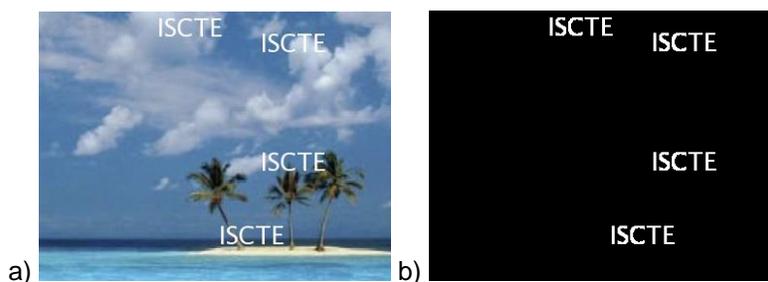


Figura 4.39: Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25] a) Imagem Original b) Máscara

Retocagem Digital

Algoritmo FoE Model:

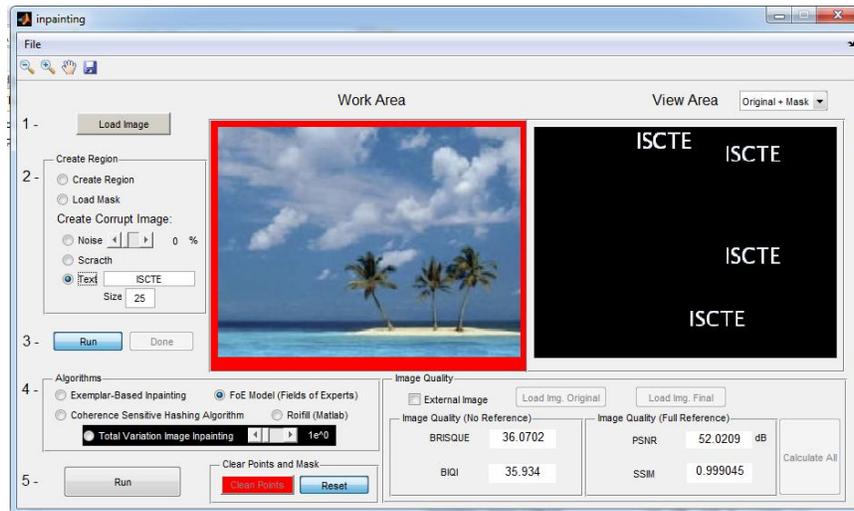


Figura 4.40: Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25] (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

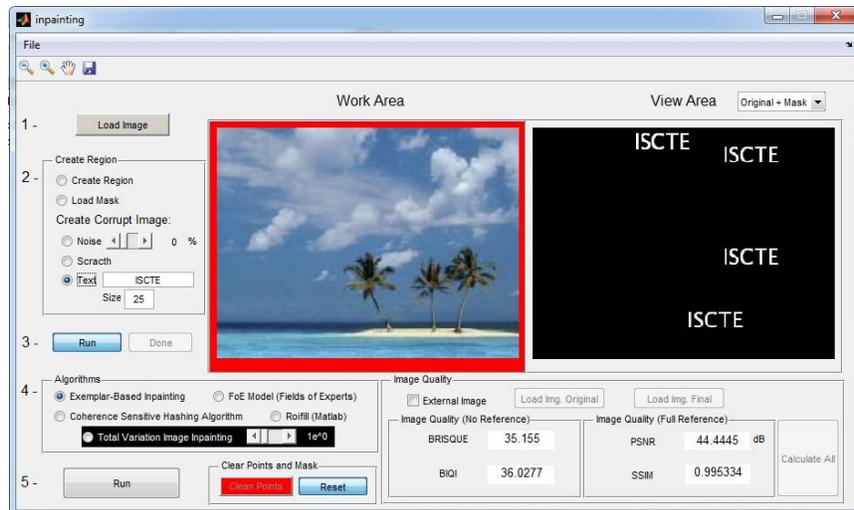


Figura 4.41: Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25] (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

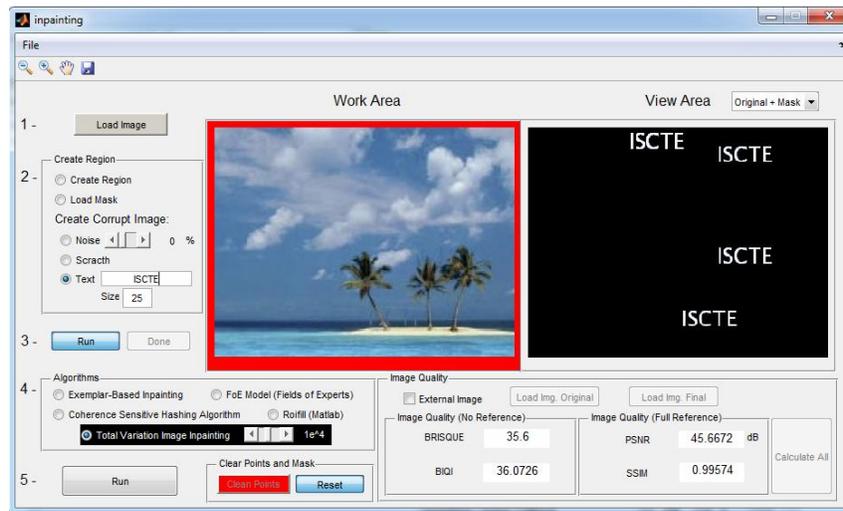


Figura 4.42: Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25] (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

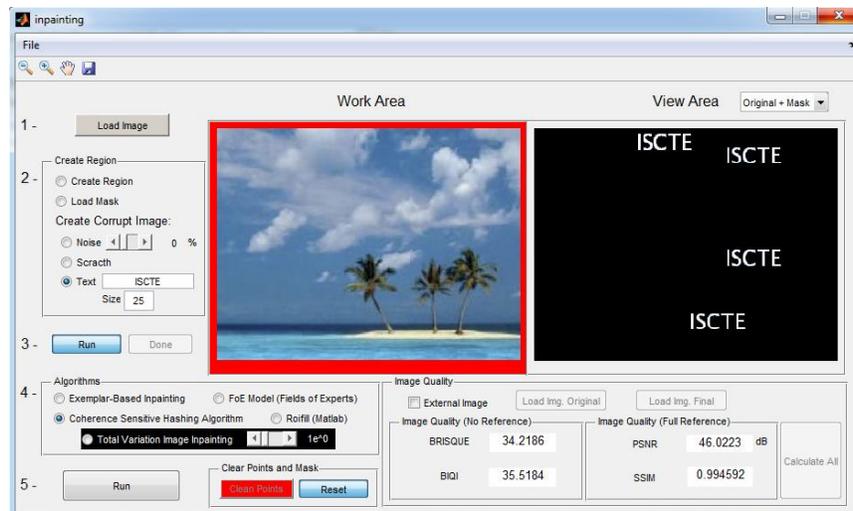


Figura 4.43: Teste 7 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 25] (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 7: Teste 7 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40]

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	52.02	44.44	45.67	46.02
	SSIM	0.9990	0.9953	0.9957	0.9946
No-Reference	BRISQUE	36.07	35.16	35.60	34.22
	BIQI	35.93	36.03	36.07	35.52

Neste sétimo teste, no qual é aplicada uma distorção do tipo texto, os resultados visuais são bons nos quatro algoritmos. Em termos de métricas, para as métricas FR o algoritmo de Bertalmio é o que obtém os melhores valores de PSNR e SSIM, com 52.02 dB e 0.9990 respetivamente. Nas Métricas NR os valores são muito idênticos entre algoritmos, mas ficando o algoritmo de Lozenzi com os melhores valores nas métricas de BRISQUE e BIQI, com 34.22 e 35.52 respetivamente.

Vamos aplicar o mesmo exemplo de remoção de texto, mas agora com o tamanho de letra 40, aplicando os 4 algoritmos e analisando por final a qualidade da imagem final obtida em cada um dos casos.

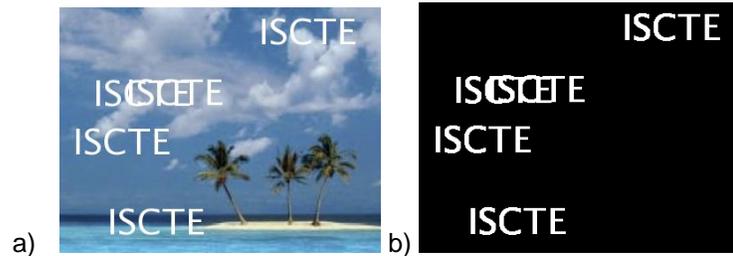


Figura 4.44: Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40] a) Imagem Original b) Máscara

Algoritmo FoE Model:

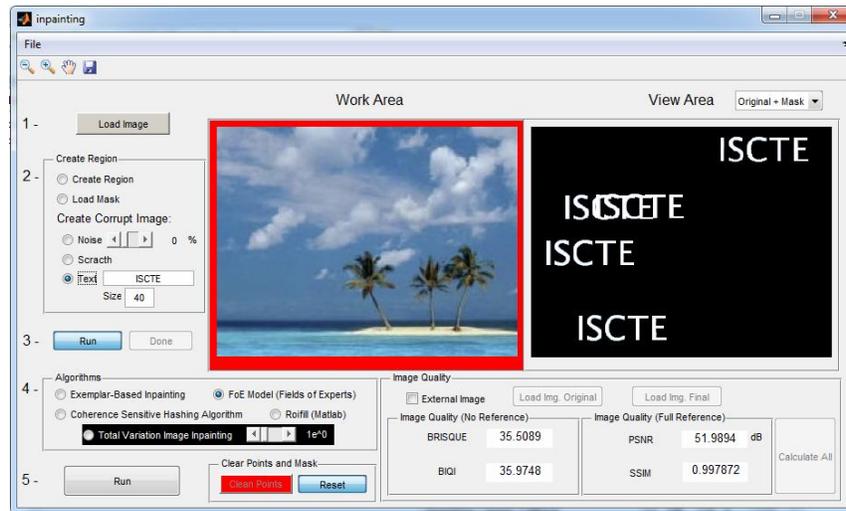


Figura 4.45: Teste 8 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 40] (Algoritmo FoE Model)

Algoritmo Exemplar-Based Retouching:

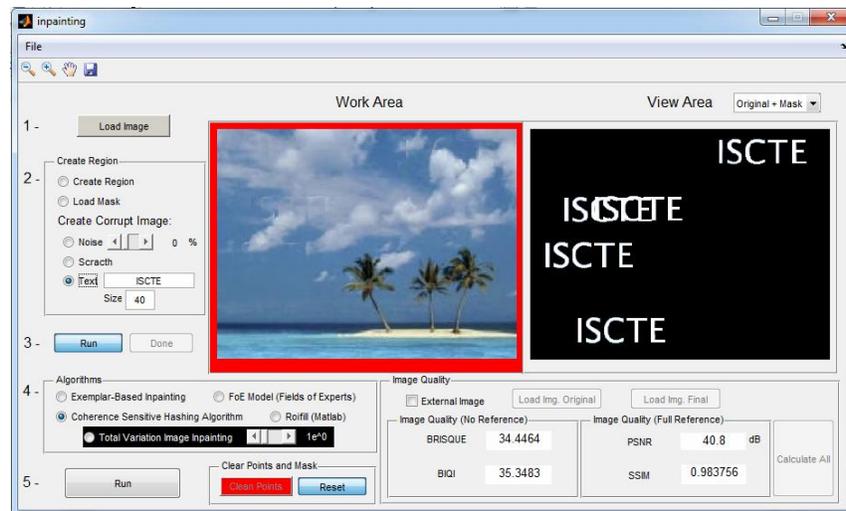


Figura 4.46: Teste 8 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 40] (Algoritmo Exemplar-Based Retouching)

Retocagem Digital

Algoritmo Total Variation Image Retouching $\rightarrow \lambda=1 \cdot \exp(4)$:

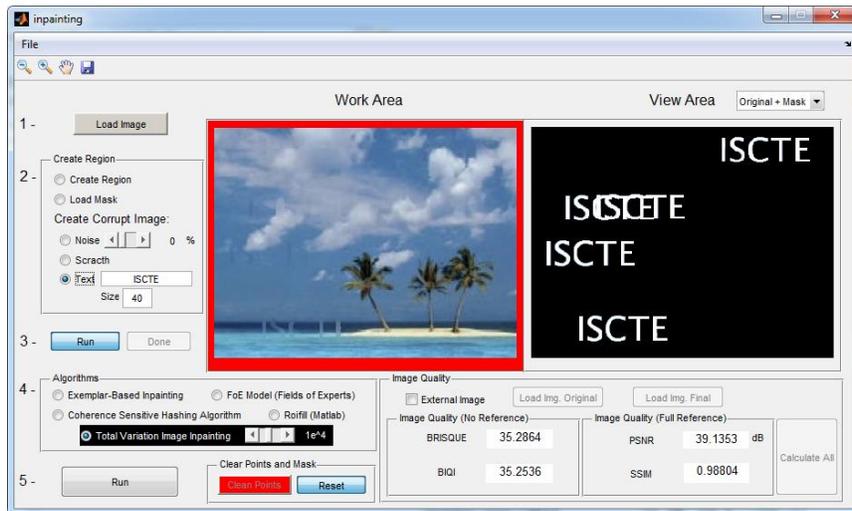


Figura 4.47: Teste 8 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 40] (Algoritmo Total Variation Image Retouching)

Algoritmo Coherence Sensitive Hashing Algorithm:

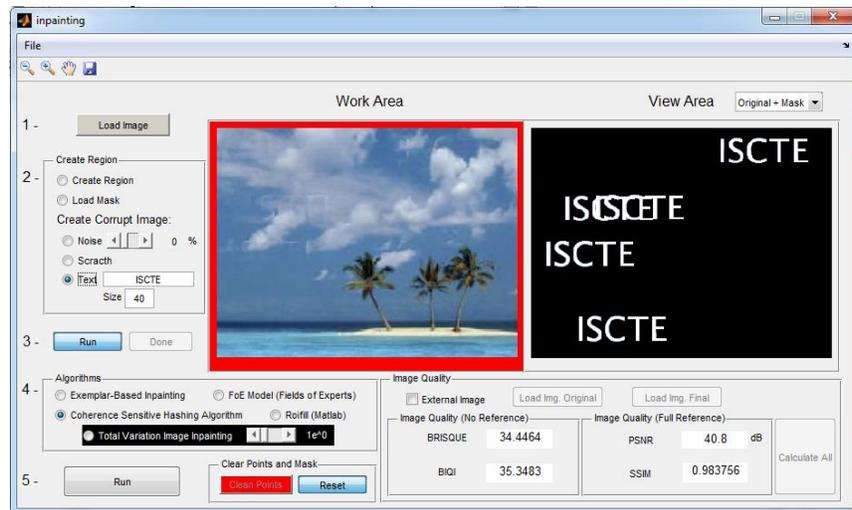


Figura 4.48: Teste 8 - Aplicação e Remoção de Texto numa Imagem ["ISCTE", Tamanho de Letra 40] (Algoritmo Coherence Sensitive Hashing Algorithm)

Tabela 8: Teste 8 - Aplicação e Remoção de Texto numa Imagem [“ISCTE”, Tamanho de Letra 40]

		FoE Model	Exemplar-Based Retouching	Total Variation Image Retouching	Coherence Sensitive Hashing Algorithm
Full-Reference	PSNR (dB)	51.99	42.72	39.14	40.8
	SSIM	0.9979	0.9895	0.9880	0.9838
No-Reference	BRISQUE	35.51	34.70	35.29	34.44
	BIQI	35.98	35.87	35.25	35.35

Neste último teste os resultados finais já diferem um pouco do teste anterior, pois o texto tem um tamanho superior e depois de aplicados os algoritmos em alguns dos resultados é possível ver pequenos defeitos nas imagens finais obtidas. Nas métricas tanto FR como NR obtém-se praticamente os mesmos resultados. Para as métricas FR o algoritmo de FoE Model é o que obtém os melhores valores de PSNR e SSIM. Nas métricas NR os valores são muito idênticos novamente, o algoritmo de Coherence Sensitive Hashing Algorithm continua com o melhor valor de BRISQUE, mas no entanto o melhor valor de BIQI está no algoritmo de Total Variation Image Retouching, não tendo uma diferença muito grande para a do algoritmo de Coherence Sensitive Hashing Algorithm.

Capítulo 5

Observações Finais

5.1 Conclusões

Ao longo desta Dissertação foi desenvolvida uma aplicação para Retocagem Digital (termo inglês: Inpainting), que integrou vários algoritmos propostos na literatura.

Nesta aplicação, apenas foram considerados quatro algoritmos descritos no Capítulo 2, onde se revê o estado de arte da Retocagem Digital. Para além dos algoritmos base, foram ainda criados métodos de utilização de máscaras sobre as imagens e métodos de criação de distorções de imagem de três tipos: ruído, riscos, e texto. Com objetivo de obter resultados finais diferentes e poder ter diversos resultados de qualidade de imagem para os quatro algoritmos a fim de poder haver comparações.

Os métodos de Retocagem Digital funcionam todos de forma diferente, pois são técnicas diferentes, num dos casos a técnica é capaz de fazer o preenchimento das áreas de reconstrução pixel a pixel, utilizando a informação dos pixéis vizinhos. Noutros casos existe a possibilidade de as técnicas fazerem o preenchimento através de blocos de pixéis, tornando os resultados por vezes piores. Os resultados experimentais obtidos em geral foram bons e permitiram demonstrar vários cenários. Ao mesmo tempo foi possível comprovar a qualidade do resultado final, mas em alguns casos a máquina não tem a mesma percepção que a visão humana, não indicando o melhor de todos como o primeiro.

Com este trabalho foi possível desenvolver o meu conhecimento sobre as técnicas de Retocagem Digital e todo o processamento envolvido na transformação das imagens, até obter a imagem final pretendida, dentro do possível.

5.2 Futuros Melhoramentos

Esta aplicação de Retocagem Digital pode sofrer ainda alguns melhoramentos, tanto na parte gráfica como ao nível da implementação dos vários algoritmos. Mas sobretudo está “aberto” ao acrescento de mais algoritmos de Retocagem Digital, métricas de qualidade de imagem, etc.

Métricas de qualidade essas que podem vir a tornar a análise do resultado mais fácil e mais preciso, pois existem mais dados de análise para tomar uma decisão final, quanto à qualidade da imagem obtida em relação a imagem original.

É possível obter resultados melhores, utilizando filtros para melhor a imagem (exemplo: filtro passa-baixo, filtro passa-alto, etc.) antes e possivelmente depois de aplicado o algoritmo de Retocagem Digital. O algoritmo que apresenta melhores resultados para restauros de áreas pequenas é o Fields of Experts Model enquanto para restaurar áreas maiores o melhor algoritmo é o Exemplar-Based Retouching.

Como referido anteriormente neste capítulo é possível acrescentar novos algoritmos, sejam eles mais antigos ou mais recentes, aproveitando sempre a interface desta aplicação para a utilização e passagem das imagens ao longo dos vários módulos que constituem a aplicação, sendo apenas necessário chamar a função desse novo algoritmo que for implementado e passando os parâmetros necessários. É possível acrescentar e utilizar técnicas baseadas na texturização, técnicas de decisão por prioridades em termos da escolha de pixels de preenchimento ou até mesmo de blocos de preenchimento, etc.

Outro possível melhoramento prende-se com a capacidade de processamento de alguns algoritmos, que pode sofrer otimizações na forma de tratamento, leitura e escrita dos pixels.

Bibliografia

- [1] M. Bertalmío, G. Sapiro, V. Caselles e C. Ballester., "Image Inpainting ", Proceedings of SIGGRAPH, New Orleans, LA, USA, Julho 2000. [paper]

- [2] T. F. Chan e J. Shen, "Mathematical Models for Local Nontexture Inpainting ", SIAM Journal of Applied Mathematics, vol. 62, no. 3, pp. 1019-1043, 2001. [paper]

- [3] A. Criminisi, P. Perez e K. Toyama, "Object Removal by Exemplar-Based Inpainting", Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. [paper]

- [4] M. Bertalmio, G. Sapiro, V. Caselles e C. Ballester, "Image Inpainting", University of Minnesota, Universitat Pompeu Fabra, 2000.[paper]

- [5] Shengfeng Li, Rujing Wang, Jin Xie e Yi Dong, "Exemplar Image Inpainting by Means of Curvature-Driven Method", International Conference on Computer Science and Electronics Engineering, 2012. [paper]

- [6] Sanjeev Kumar, Mainak Biswas, Serge J. Belongie e Truong Q. Nguyen, "Spatio-Temporal Texture Synthesis and Image Inpainting For Video Applications", University of California, San Diego, 2005 [paper]

- [7] L. Lorenzi, F. Melgani e G. Mercier, "Inpainting Strategies for Reconstruction of Missing Data in VHR Images", IEEE Geoscience and Remote Sensing Letters, Setembro 2011. [paper]

- [8] Jian Sun, Lu Yuan, Jiaya Jia e Heung-Yeung Shum, "Image Completion with Structure Propagation", Microsoft Research Asia, Tsinghua University, Chinese University of Hong Kong, 2005 [paper]

- [9] Lili Wang e Yi Wan, "Multiresolution Image Inpainting", Lanzhou University, China, 2012 [paper]

- [10] Mengxin Li, Ying Zhang, Xinghua Xia e Yang Cao, “Comparative Research on Image Inpainting Techniques”, Shenyang Jianzhu University, China, 2012 [paper]
- [11] Dang Thanh Trung, Chaker Larabi e Azeddine Beghdadi, “Multi-resolution Patch and Window-Based Priority for Digital Image Inpainting Problem”, Universite de Poitiers, França, 2012 [paper]
- [12] Lorenzi, “Development of an Innovative System for the Reconstruction of New Generation Satellite Images”, 2012 [paper]
- [13] Ke Gu, Guangtao Zhai, Xiaokang Yang e Wenjun Zhang, “An improved full-reference image quality metric based on structure compensation”, Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China, 2012 [paper]
- [14] Artigo: FR, NR e RR, link: <http://live.ece.utexas.edu/research/quality/intro.htm>, visualizado em Novembro de 2014 [online]
- [15] Artigo: BRISQUE, link: <http://live.ece.utexas.edu/research/quality/nrqa.htm>, visualizado em Novembro de 2014 [online]
- [16] Huixuan Tang, Neel Joshi e Ashish Kapoor, “Learning a Blind Measure of Perceptual Image Quality”, University of Toronto and Microsoft Research, 2012 [paper]
- [17] Artigo: MSE e PSNR, link: <http://emanuelecolucci.com/2011/04/image-and-video-quality-assessment-part-one-mse-psnr/>, visualizado em Novembro de 2014 [online]
- [18] Imagens: Exemplo Criminisi, link: <http://cs.brown.edu/courses/csci1950-g/results/final/eboswort/>, visualizado em Março de 2015 [online]
- [19] Texto de Agradecimento: Antoine de Saint-Exupéry, link: <http://peroladecultura.blogspot.pt/2010/06/110-anos-de-saint-exupery.html>, visualizado em Abril de 2015 [online]
- [20] Artigo: The SSIM Index for Image Quality Assessment, link: <http://www.cns.nyu.edu/~lcv/ssim/>, visualizado em Maio de 2015 [online]

- [21] Zhou Wang e Alan C. Bovik, "Mean Squared Error: Love It or Leave It?", IEEE Signal Processing Magazine, Janeiro 2009 [online]
- [22] Imagem: Exemplo SSIM, link: <http://emanuelecolucci.com/2011/06/image-and-video-quality-assessment-part-two-ssim-index/>, visualizado em Maio de 2015 [online]
- [23] Código: Stefan Roch, link: <http://www.gris.informatik.tu-darmstadt.de/~sroth/research/foe/downloads.html#test>, visualizado em Dezembro de 2014 [online]
- [24] Código: Sooraj Bhat, link: <http://cs.brown.edu/courses/csci1950-g/results/final/eboswort/>, visualizado em Janeiro de 2015 [online]
- [25] Código: Pascal Gereuer, link: <http://www.imm.dtu.dk/~pcha/mxTV/>, visualizado em Janeiro de 2015 [online]
- [26] Código: Simon Korman e Shai Avidan, link: <http://www.csee.wvu.edu/~xinl/demo/inpainting.html>, visualizado em Fevereiro de 2015 [online]
- [27] Códigos: Métricas de BRIQUE, BIQI, MSE e SSIM, link: <http://tdistler.com/iqa/algorithms.html>, visualizado em Abril de 2015 [online]
- [28] Imagem: Exemplo YCbCr, link: <https://en.wikipedia.org/wiki/YCbCr>, visualizado em Setembro de 2015 [online]