DM

Pesquisa Multimodal de Imagens em Dispositivos Móveis

DISSERTAÇÃO DE MESTRADO

José Ricardo de Abreu Carvalho
MESTRADO EM ENGENHARIA INFORMÁTICA

UNIVERSIDADE da MADEIRA
A Nossa Universidade
www.uma.pt

outubro | 2021

Pesquisa Multimodal de Imagens em Dispositivos Móveis

DISSERTAÇÃO DE MESTRADO

José Ricardo de Abreu Carvalho
MESTRADO EM ENGENHARIA INFORMÁTICA

ORIENTAÇÃO
Pedro Filipe Pereira Campos

COORIENTAÇÃO
Diogo Nuno Crespo Ribeiro Cabral

UNIVERSIDADE da MADEIRA

Pesquisa Multimodal de Imagens em Dispositivos Móveis

José Ricardo de Abreu Carvalho

Constituição do júri de provas públicas:

Presidente:

- Karolina Baras, (Professora Auxiliar da Universidade da Madeira)

Arguente:

- Filipe Magno de Gouveia Quintal, (Professor Auxiliar Convidado da Universidade da Madeira)

Vogal:

- Diogo Nuno Crespo Ribeiro Cabral, (Investigador Auxiliar e Professor Auxiliar Convidado do ITI/LARSyS, IST, Universidade de Lisboa)

Dezembro 2021
Funchal – Portugal

Resumo

Apesar das evoluções no campo de Reverse Image Search, com algoritmos cada vez mais robustos e eficazes, continua a haver interesse para que as técnicas de pesquisa possam ser aprimoradas, melhorando a experiência do utilizador na procura das imagens que tem em mente.

O objetivo principal deste trabalho foi desenvolver uma aplicação para dispositivos móveis (smartphones) que permitisse ao utilizador encontrar imagens através de inputs multimodais. Assim, esta dissertação, para além de propor pesquisas por diversos modos (palavras-chave, desenho, e imagens da câmara ou existentes no dispositivo), propõe que o utilizador consiga criar uma imagem por si só através de desenho, ou editar/alterar uma imagem existente, tendo feedback no momento aquando de cada alteração/interação. Ao longo da experiência de pesquisa, o utilizador consegue usar as imagens encontradas (que achar relevantes) e ir aprimorando a pesquisa através dessa edição, indo de encontro ao que pensa encontrar.

A implementação desta proposta teve como base a Cloud Vision API da Google responsável pela obtenção dos resultados através do input de imagem, a Google Custom Search API para a obtenção de imagens através do input por texto, e a framework ATsketchkit que permitia a criação de desenho, para o sistema iOS da Apple.

Foram realizados testes com um conjunto de utilizadores com diversos níveis de experiência em pesquisa de imagens e na habilidade de desenho, permitindo aferir a preferência nos diferentes métodos de input, a satisfação na obtenção dos resultados, bem como da usabilidade do protótipo.

Palavras-chave: Pesquisa Multimodal, Reverse Image Search, Visão Computacional, Content-based Image Retrieval

Abstract

Despite the evolution in the field of reverse image search, with algorithms becoming more robust and effective, there still interest for improving search techniques, improving the user experience when searching for the images the user has in mind.

The main goal of this work was to develop an application for mobile devices (smartphones) that would allow the user to find images through multimodal inputs. Thus, this dissertation, in addition to propose the search for images in different ways (keywords, drawing/sketching, and camera or device images), proposes that the user can create an image by himself through drawing, editing / changing an existing image, having feedback at the time of each change / interaction. Throughout the search experience, the user can use the images found (which it finds relevant) and improve the search through its edition, going against what it thinks to find.

The implementation of this proposal was based on a Google Cloud Vision API responsible for obtaining the results, and the ATsketchkit framework that allowed the creation of drawings, for Apple's iOS system.

Tests were carried out with a set of users with different levels of experience in image research and different drawing ability, allowing to assess preference in different input methods, satisfaction with the images retrieved, as well as the usability of the prototype.

Keywords: Multimodal Search, Reverse Image Search, Computer Vision, Content-based Image Retrieval

Agradecimentos

Em primeiro lugar gostaria de agradecer à minha família pois foram muito importantes para que conseguisse alcançar este marco académico.

Uma menção de agradecimento também aos meus amigos mais próximos e também ao orientador, o Dr. Diogo Cabral, por toda ajuda e orientação para o desenvolvimento deste trabalho.

Índice

Índice de Figuras

Índice de Tabelas

Lista de Acrónimos e Siglas

API	*Application programming interface*
CBIR	*Content-based Image Retrieval*
CBVIR	*Content-based Visual Information Retrieval*
CNN	*Convolutional Neural Network*
CSI	*Creativity Support Index*
IA	Inteligência Artificial
IDE	*Integrated Development Environment*
IR	*Image Retrieval*
JSON	JavaScript *object notation*
OCR	*Optical Character Recognition*
QBIC	*Query By Image Content*
ML	*Machine Learning*
MVC	*Model View Controller*
RIL	Reverse Image Lookup
URL	*Uniform Resource Locator*
RIS	*Reverse Image Search*
TBIR	*Text-based Image Retrieval*
SBIR	*Semantic Based Image Retrieval / Sketch-Based Image Retrieval*
SUS	*System Usability Scale*

1. Introdução

Cada vez mais a informação é visual, e os equipamentos são acumuladores de dados, nomeadamente fotos e imagens, por isso a interação com estes dados deve ser aprimorada. Apesar de ser positiva a quantidade de dados acessíveis (quase ilimitada), isto provoca também com que seja mais difícil encontrar a melhor informação.

À medida que as imagens são produzidas, é necessário encontrar formas eficientes de as descobrir. Os métodos de pesquisa têm vido a ser melhorados de forma a tentar solucionar o problema de encontrar a informação "certa".

As pesquisas na web começaram por ser baseadas em texto (texto-to-text), no entanto, com o evoluir da tecnologia e respetivo impacto na sociedade, tornou-se importante não só devolver informação (em texto) relacionada com a pesquisa, mas também imagens (texto-to-image). Hoje em dia o acesso a todo o tipo de informação, textual e visual, é muito mais fácil e imediato (Chatfield et al. 2015) .

O primeiro motor de busca a lançar um recurso que permitia inserir um termo de texto e retornar imagens foi o Altavista no final dos anos 90's. No entanto estas pesquisas apenas se baseavam em títulos e descrições das imagens, ou texto correspondente a acompanhá-las nas páginas web, não se baseavam na capacidade de obtenção de informações através do processamento das imagens existentes nas bases de dados[1].

Percebeu-se que seria melhor aproveitar a informação retirada das imagens de forma a conseguir efetuar comparações entre elas. No entanto, apenas em 2008, passados aproximadamente 10 anos do lançamento da funcionalidade de pesquisa de imagens através do input de texto, é que surge a ferramenta "TinEye"[1], a primeira a conseguir a funcionalidade de comparar imagens, onde o utilizador, em vez de descrever a imagem através de texto, conseguia encontrar imagens semelhantes e a sua origem através de uma imagem de input.

Existem cerca de 6,3 biliões de smartphones[2] em todo o mundo dotados de câmara e capacitados para usar a internet. Os smartphones têm características particulares como

[1] Syte, https://www.syte.ai/blog/visual-ai/brief-history-image-search, Data do último acesso: 26/09/2020

[2] Statista, https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide, Data doúltimo acesso: 14/08/2020

ecrãs de pequenas dimensões e interações multimodais através, por exemplo de ecrãs tácteis (Russell-Rose e Tate 2013). Se algumas destas características podem promover novas interações também contêm os seus desafios, como o é caso da dificuldade de escrita em teclados virtuais(Varcholik, LaViola, e Hughes 2012). Alguma investigação tem sido feita com o objetivo ultrapassar estes desafios como na ExplorationWall (Klouche et al. 2015), uma interface de pesquisa exploratória em artigos científicos para ecrãs tácteis. No entanto, no que diz respeito a pesquisa de imagens continua a existir um foco em interfaces de para sistemas de interação tradicionais, i.e., com rato e teclado (Cruz, Cabral, e Campos 2021).

Existem então neste momento oportunidades para utilizar cada vez mais a tecnologia de pesquisa inversa de imagens, chamada de "Reverse Image Search" (RIS) ou "Reverse Image Lookup" (RIL), que no fundo é uma pesquisa por imagens tendo como input (query) também uma outra imagem (Thompson e Reilly 2017). Houve alguma demora no avanço desta funcionalidade, provavelmente devido às limitações tecnológicas da época. No entanto nestes últimos anos os grandes progressos nos equipamentos (computadores, smartphones e tablets), como as melhorias nas comunicações, permitiram um avanço gigante no acesso e partilha de conteúdos. Isto é, os equipamentos passaram a ser munidos com espaço de armazenamento maior e câmaras a custos mais acessíveis, permitindo que a recolha de conteúdos multimédia fosse mais abrangente, rápida e fácil, e as comunicações passaram a ser capazes de transmitir a uma velocidade maior a informação, tornando possível enviar e receber ficheiros de forma quase imediata (Yasmin, Mohsin, e Sharif 2014).

1.1. Objetivo

O objetivo deste trabalho será o de estudar a pesquisa de imagens em dispositivos móveis.

Terá o propósito de desenvolver uma solução que permita estudar os diferentes tipos de input, e tentar melhorar o os resultados obtidos através das pesquisas por imagens em dispositivos móveis. A interação do utilizador deverá proporcionar um melhor acesso aos resultados, bem como conseguir ajudá-lo a encontrar a informação que tem em mente. Para além da pesquisa por palavra-chave, o utilizador poderá criar (desenhar usando o dedo) conteúdos ou editar uma imagem já existente, pesquisando por conteúdo relacionado e obtendo esses resultados a cada interação com o sistema. Será importante perceber como os utilizadores interagem com os sistemas de "Reverse Image Search", mas também como estes sistemas interagem com o utilizador.

1.2. Contribuições

As principais contribuições deste trabalho são:

- O desenvolvimento de uma aplicação móvel para pesquisa multimodal de imagens
- O desenvolvimento de 3 interfaces de pesquisa de imagens: texto, desenho, fotografia/imagem
- O desenvolvimento e avaliação de uma *feature* que permite efetuar pesquisas inversas de imagens, a cada momento da interação.
- A avaliação das diferentes modalidades do protótipo com 12 utilizadores.

1.3. Estrutura do Documento

Este relatório contém 5 capítulos, com a seguinte estrutura:

- No primeiro capítulo é feita uma introdução do trabalho apresentando a motivação e objetivo;
- No segundo capítulo é apresentado o estado da arte, onde foi feita uma pesquisa de informação sobre a evolução dos temas relevantes para este trabalho.
- No terceiro capítulo é feita uma descrição do protótipo, a arquitetura de sistema, interface e interação.
- No quarto capítulo é descrito o processo de avaliação do protótipo e são demonstrados os resultados obtidos.
- No quinto capítulo é apresentada uma conclusão e são mencionadas algumas considerações futuras.

"De que vale a informação, se não a conseguimos encontrar?"

Y. Alp Aslandogan, C. Thier, C. Yu, Y. Zou, N. Rishe, "Using Semantic Contents and WordNet" in Image Retrieval", *Proceedings of SIGIR 97, Philadelphia, USA*, 1997, pp. 286-295.

2. Estado da arte

Neste capítulo é feito um enquadramento dos conceitos relevantes para o trabalho e evolução da tecnologia no que diz respeito a "Image Retrieval" e a "Reverse Image Search", nomeadamente sobre formas de pesquisar, tecnologias usadas, algoritmos, Redes Neurais e interfaces para dispositivos móveis.

2.1. Técnicas de Pesquisa por Imagem ("Image Retrieval")

Antes de extrair informações sobre as imagens, deve-se perguntar que tipo de informação é conveniente retirar para os humanos, e se essa informação será útil no processo de pesquisa.

Ao olhar para uma imagem, um ser humano consegue identificar objetos, pessoas lugares, ou até mesmo perceber o que se está a passar de acordo com o que acontecia no momento em que a foto foi tirada. Ainda temos que considerar que a perceção da imagem pode ser influenciada pelo conhecimento, experiências, ou cultura do utilizador. Logo, é possível que dois indivíduos tenham diferentes perceções da mesma imagem. Isto irá resultar que estes indivíduos façam diferentes "queries" apesar de quererem a mesma imagem (Bagyammal e Parameswaran, 2015). Um sistema de Image Retrieval ideal deve ser flexível o suficiente para ter estas situações em conta.

As pesquisas de imagens na internet necessitam de técnicas eficientes e efetivas devido ao aumento exponencial das imagens digitais. A pesquisa de imagens é considerada uma área de pesquisa muito extensa.

As técnicas de "Image Retrieval" começaram a ser exploradas desde os anos 1970's para as áreas de Gestão de Base de Dados e "Computer Vision" (Y. Riu et. al, 1999). Esta técnica de pesquisas de imagem pode ser dividida em três categorias: "Text-based Image Retrieval" (TBIR), "Content-based Image Retrieval" (CBIR), e "Semantic-based image retrieval" (SBIR).

Para o **TBIR** eram adicionadas às imagens anotações, descrições, palavras-chave. Para além do conteúdo, também eram adicionadas informações como o nome da imagem, o formato, o tamanho, ou as dimensões. Inicialmente estas informações eram adicionadas manualmente, o que não era prático para um grande número de imagens. Outro inconveniente era a riqueza e detalhe das imagens, o que provoca com que informações introduzidas por utilizadores diferentes, pudessem estas também ser

diferentes para conteúdos similares. Outro problema era a dependência do idioma (Tamura et al. 1984).

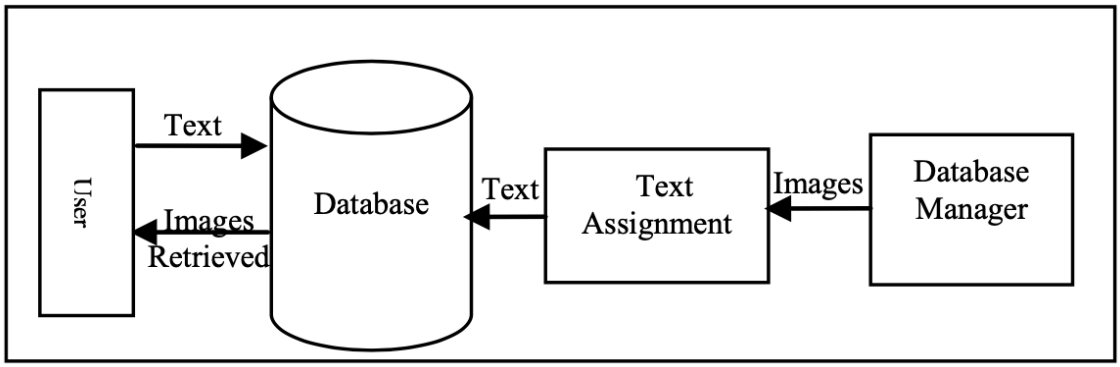

Figura 1- Sistema típico "Text Based Image Retrieval" (Alkhawlani et. Al 2015)

Os avanços na internet, e da produção digital nos campos das ciências, educação, medicina e indústria forçaram a evolução das técnicas, surgindo então o CBIR, também conhecido por QBIC (Query By Image Content) ou CBVIR (Content-based Visual Information Retrieval). Esta tecnologia surgiu já após os anos noventa, mas só depois do ano 2000 é que tem sido alvo de muita atenção, motivado pela necessidade de lidar de forma eficiente com a enorme quantidade de dados multimédia. Abrange áreas como segmentação de imagem e extração de recursos da imagem.

Como podemos veriricar na Figura 2, um sistema típico de CBIR é composto por duas etapas, sendo uma "offline" e outra "online". Na etapa offline é feita a extração das informações das imagens da base dados e armazenada numa base dados para esse fim. Na etapa "online" o utilizador efetua a query através do input da imagem, e as informações dessa imagem são extraídas. Os resultados são apresentados de acordo com a similaridade dos dados extraídos da imagem de input, com os dados da base de dados. Depois, os esquemas de indexação fazem com que os resultados sejam apresentados de forma a que a pesquisa seja o mais eficiente possível.

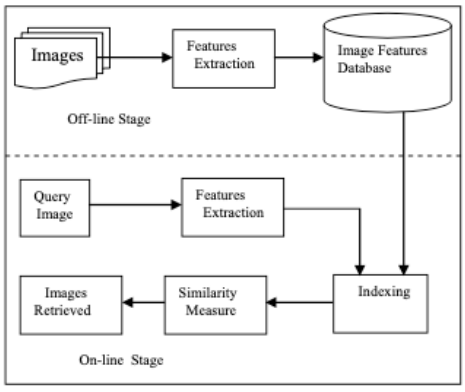

Figura 2 - Sistema típico de "Content Based Image Retrieval"(Alkhawlani et. Al 2015)

Uma vez que o sistema CBIR tem maior relevância com este projeto, deu-se um pouco mais de relevo ao mesmo.

2.1.1. Funcionamento do CBIR (Content-Based Image Retrieval)

Há um vasto número de campos que podem beneficiar desta tecnologia CBIR. Galerias de arte, projetos de engenharia, design de interiores, sistemas de informação geográfica, previsão do tempo, retalho, moda, direitos de autor, investigação criminal, análises médicas, reconhecimento facial, etc.

Os sistemas CBIR podem ser categorizados de acordo com a sua aplicação, mas também de acordo com (Banfi, 2000):

- Database Content – As imagens dessa base de dados são mais ou menos heterogéneas.

As bases de dados podem ser muito específicas, isto é, as imagens podem ser do mesmo tipo, retiradas nas mesmas condições (como por exemplo imagens de ressonâncias magnéticas através da mesma máquina); podem ser uma imagens numa base de dados com características específicas (imagens de selos num fundo negro, mas usando câmaras diferentes e condições de luz diferentes); bases de dados monotemáticas (as imagens respeitam um determinado tema ou assunto, como por exemplo imagens de flores); ou bases de dados heterogéneas onde existem todo o tipo de imagens sem respeitar qualquer critério.

- Query form - O sistema aceita querys de vários tipos como: palavras-chave, características (por exemplo a percentagem de uma determinada cor), ou uma imagem.
- Image Description – Quando uma imagem é adicionada à base de dados, são adicionados descritores (labels) à imagem. Aquando da fase da consulta, são usados esses descritores para julgar a semelhança entre a query e a imagem da base de dados. A informação associada a uma imagem pode ser de 3 tipos: semântica (descrição da imagem de acordo com o seu conteúdo: tema, ação apresentada), informações primitivas (cor, textos, formas, bordas, regiões homogéneas); informação factual (informação que não pode ser extraída da imagem, como o nome do fotógrafo, a data, ou as condições de como a foto foi tirada).

- Interação entre o utilizador e o sistema CBIR – melhorar a interação entre o sistema e o utilizador pode ajudar a obter melhores resultados. Permitir ao utilizador submeter uma query baseada na anterior, selecionar parte da imagem que achamos relevante, etc.

Uma sessão de CBIR típica pode ser resumida da seguinte forma:

- O utilizador produz uma query através de uma imagem que é submetida ao sistema CBIR.
- O sistema CBIR calcula a similaridade entre a imagem utilizada na query e as imagens da base de dados. Isto é feito de acordo com as descrições da consulta e as descrições das imagens da base de dados
- O sistema CBIR devolve uma lista de imagens ordenadas de acordo com a similaridade.
- O utilizador pode modificar ou criar nova query.

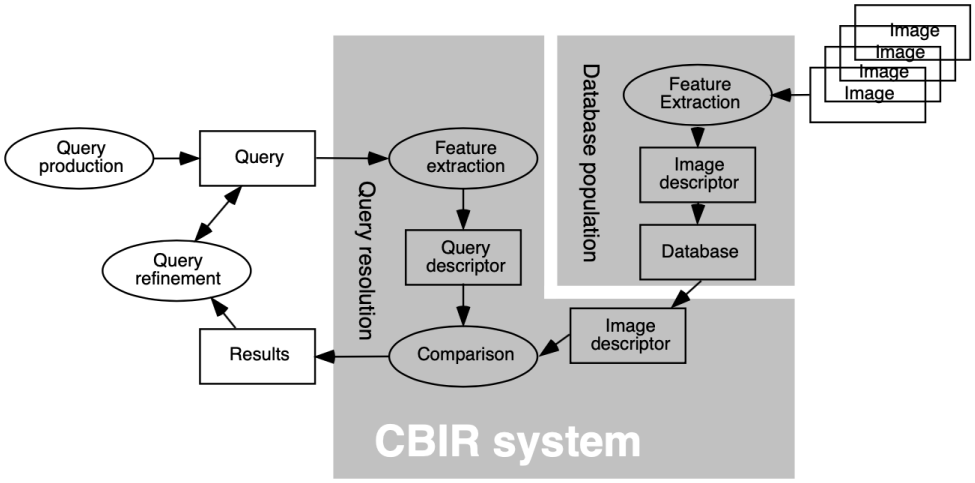

Figura 3 - Sistema de "Content-based Image Retrieval" com interação (Banfi, 2000)

Segundo Banfi (2000) um sistema CBIR ideal seria aquele que permite uma liberdade total de query, onde qualquer imagem tem descritores completos. Isto não é possível devido a alguns fatores: a extração de semântica da imagem de forma automática é difícil (aqui entram IA – Inteligência Artificial e ML – Machine Learning), e através de humanos é um processo muito demorado e com um custo muito grande.

Essa liberdade na query é uma tarefa muito pesada, provocando um consumo de recursos e tempo maior. Segundo Banfi (2000) combinar vários tipos de querys é possível, mas definir o ranking nos resultados poderá ser um problema.

Dada uma base de dados de imagens, o que se pretende de um sistema CBIR é que tenha a habilidade de selecionar um conjunto de imagens de acordo com a query submetida. O tamanho dos resultados pode ir de vazio até toda a base de dados. Como os resultados são baseados na semelhança, são então apresentados por essa ordem.
Para que um sistema CBIR tenha uma boa performance é necessário considerar (Banfi 2000):

- A indexação da base de dados, pois é essencial para comparar os resultados com a query, devendo ser uma tarefa de curta duração.
- Os resultados devem corresponder de acordo com à perceção humana, isto é o utilizador deve considerar os resultados satisfatórios.
- A eficiência dos resultados, uma vez que o objetivo principal é obter os resultados mais relevantes melhor posicionados que os restantes.

Para um humano, esse conjunto de pixéis formam um significado, provocando um reconhecimento de objetos, eventos, locais, etc., e pode ser das melhores formas de partilhar, perceber e memorizar informação. Para um computador uma imagem é uma grelha de pixéis coloridos e não significa nada, a não ser que alguém lhe diga como interpretá-la. A conversão de uma imagem para um recurso de baixo nível, permite à máquina conseguir comparar duas ou mais imagens.

Como falado anteriormente, surgiu então o CBIR (Content-Based Image Retrieval) que é uma tecnologia associada à "Computer Vision" para a pesquisa de imagens. Inicialmente eram sistemas de baixo nível, que analisavam o conteúdo da imagem, em vez de apenas procurar as palavras-chave associadas às informações das mesmas. Características como a cor e a forma passam a estar automaticamente relacionadas à imagem sem ser necessário indicar essa informação. Esta tecnologia tinha algumas limitações. Apesar de conseguir identificar corretamente o tamanho e as cores das imagens, imagens com contraste igual em zonas iguais poderiam ser relacionadas erradamente.

Posteriormente, estes sistemas foram evoluindo, e foi possível obter representações de nível médio, que consistem em sub-imagens, isto é, regiões ou detalhes salientes. Depois de determinar esses elementos, estes podem ser vistos como entidades independentes durante a pesquisa. No entanto uma imagem passa a ser um conjunto de palavras sem contexto e conexão.

As representações de alto nível visam colmatar a falta de semântica do nível anterior, isto é, introduzem a capacidade de atribuir contexto a um conjunto de imagens detetadas. Por exemplo, um utilizador, através de feedback, consegue relacionar e contextualizar de forma a que esse contexto esteja de acordo com o input da pesquisa. A dificuldade hoje em dia é fazer com que seja a máquina, através de Inteligência artificial, conseguir atribuir semântica à imagem. 1

Então, a principal lacuna do CBIR, e que ainda hoje em dia é um grande desafio, é a semântica (semantic gap) entre as características de alto e baixo nível. Isto é, a diferença entre o que constitui uma imagem e aquilo que as pessoas percebem da imagem (Xu et al. 2016). O SBIR pode ser feito através da extração das características de baixo nível das imagens, para identificar significados através de regiões ou objetos relevantes. Essas regiões/objetos irão para um processo de extração de semântica de forma a serem guardadas e mapeadas como podemos ver na Figura 4.

O mapeamento de semântica é usado para encontrar o melhor conceito que descreve aquele segmento (região/objeto) baseado nas informações de baixo nível. Uma forma de dar semântica à imagem é a de, por exemplo, examinando e extraindo informação textual das páginas web onde estas estão alojadas (Alzu'bi, Amira, e Ramzan 2015).

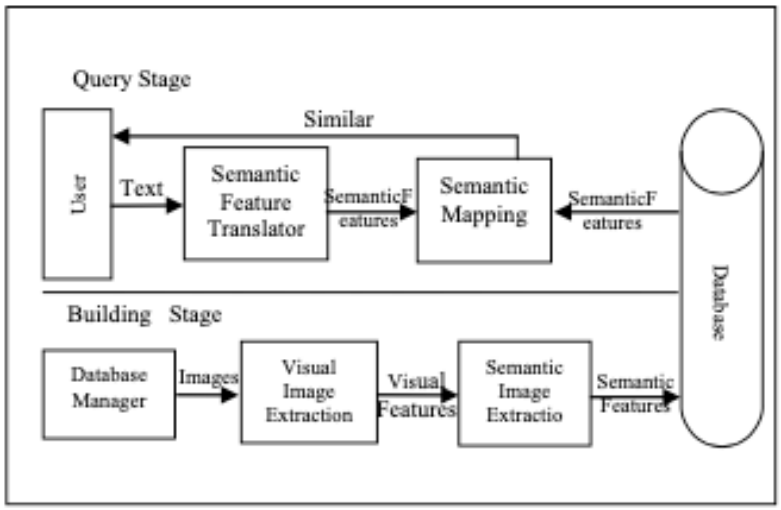

Figura 4 - Sistema típico de "Semantic Based Image Retrieval" (Alkhawlani et. Al 2015)

O AIA (Automatic Image Annotation) é o processo responsável pelo qual um computador define automaticamente os metadados na forma de "labels" ou "palavras-chave" de uma imagem digital. Refere-se à rotulação automática de imagens com base

no seu conteúdo. O conceito servirá para aprender automaticamente sobre os conceitos semânticos de um grande número de imagens de amostra para rotular outras novas.

2.2. Sistemas de pesquisa de imagens

No que diz respeito aos sistemas de pesquisa de imagens, inicialmente estes sistemas geralmente requeriam que toda a base de dados fosse percorrida para efetuar comparações. Em grandes coleções de imagens isto tornava-se moroso. Vários esquemas de indexação e filtragem foram, e continuam a ser trabalhados para reduzir o número de imagens da base de dados a ser examinada, melhorando a capacidade de resposta dos sistemas de pesquisa.

Uma das soluções é agrupar as imagens de modo a diminuir o conjunto de imagens a serem consideradas. A forma padrão em que os resultados são exibidos é uma lista classificada com as imagens mais semelhantes à consulta mostrada no topo da lista.

Os motores de busca foram implementando novas funcionalidades e a indexação baseada nos atributos visuais das imagens foram tornando-se mais precisas.

Em 2009 a Google lançou o projeto Image Swirl que apresentava grupos de imagens com características semelhantes, permitindo ao utilizador explorar várias imagens semelhantes de diferentes ângulos ou por exemplo de dia e noite. Apesar deste projeto Swirl já não estar em funcionamento, a ideia teve continuidade dando origem à funcionalidade "Related images" que ainda hoje está disponível nas pesquisas de imagens da Google[3].

A abordagem ao reconhecimento facial foi também importante para o desenvolvimento das pesquisas por imagem uma vez que implica um nível de detalhe maior e mais aprimorado.

Uma área explorada para melhorar os resultados das pesquisas por imagem foi, como já falado anteriormente, o "Feedback por relevância" onde o sistema de pesquisa solicita ao utilizador um feedback ao longo de várias interações, onde esse sistema vai adequando as imagens que melhor correspondem ao que o utilizador tem em mente. Ainda assim o "Feedback por relevância" não fornece ao sistema muitas informações além das que o utilizador achar relevantes. O feedback sobre as melhores imagens não fornece ao sistema muitas informações além do interesse do utilizador (Zhang et al. 2015).

[3] Syte, https://www.syte.ai/blog/visual-ai/brief-history-image-search, Data último acesso: 26/09/2020

A segmentação da imagem também é uma mais valia, uma vez que podemos estreitar os resultados apenas a uma parte da imagem que o utilizador identificou (Meharban e Priya, 2016).

Muitos destes sistemas estão desenhados para que seja usado um algoritmo de "Machine Learning" onde os resultados podem ser catalogados pelo utilizador de forma positiva, negativa ou nula. Isto melhora/treina os sistemas para as próximas interações. Também pode ser feita a abordagem onde o utilizador indica a percentagem da correspondência da imagem aos resultados esperados.

Permitir também ao utilizador que a query seja multimodal poderá ser vantajoso (Tomee et al. 2012), isto é, vários tipos de input na query, imagens ou misturar imagens com texto, sendo que a resposta também poderá ser multimodal. Analogamente podemos associar as pesquisas à forma como interagiríamos com um bibliotecário, isto é, poderíamos expor um conjunto de palavras chave, ou, se tivéssemos uma imagem de exemplo, fornecê-la ao bibliotecário. Este encarregar-se-ia de nos fornecer conteúdos relevantes de acordo o nosso "input" e com a sua classificação.

Embora os exemplos positivos ou negativos sejam importantes para catalogação dos resultados, poderá ser muito vantajoso que o utilizador consiga fornecer outros tipos de input, como texto, áudio, imagens, etc. Assim, o grau de precisão dos resultados será maior. Por exemplo, se o utilizador complementar com um verbo, poderá fazer sentido mostrar um vídeo demonstrando essa ação. Uma interface de "sketching" permite também ao utilizador fornecer um input diferente, o que pode potencialmente fornecer um grau de controlo maior sobre os resultados.

No que diz respeito à segmentação da imagem, é possível implementar nos sistemas de feedback o desenho de um "retângulo delimitador" (bounding box) dentro de um exemplo positivo onde restringirá a área de interesse.

O papel da Interface no processo de pesquisa também é de extrema importância. Geralmente está limitado a apresentar um conjunto de resultados organizados (em lista ou em grelha). Uma melhor apresentação desses resultados nomeadamente dando mais espaço de ecrã para imagens mais relevantes, reorganizar os resultados, permitem ao utilizador identificar melhor os resultados. A interface está baseada em alguns aspetos, como a fácil navegação.

As pesquisas na WEB em dispositivos móveis são um particular desafio devido à reduzida dimensão dos ecrãs, nomeadamente nos Smartphones. Por exemplo, o texto como método de entrada faz com que o teclado ocupe grande parte do ecrã,

empobrecendo a interação e a apresentação do conteúdo. Foi acrescentado valor aquando da introdução do "word prediction" que sugere as palavras a serem digitadas pelo utilizador, bem como o "auto-completion" que permite concluir a escrita da palavra. Estes métodos diminuíram o número de interações como também melhoraram o número de falhas na introdução do texto (Ghong et al. 2004).

A pesquisa dinâmica só recentemente passou a ser considerada pois antes estávamos condicionados à velocidade e latências da rede. Hoje em dia isso é mais viável. A ordenação hierárquica dos resultados ajuda os utilizadores a encontrar os melhores resultados. Isto toma uma importância maior nos dispositivos de ecrã reduzido, pois o utilizador tem menos espaço para consultar. Então tanto o pedido ("query"), como o resultado devem também estar ajustados com essas dimensões. Podemos classificar três estratégias para melhorar a apresentação do conteúdo (Robbins 2014): formatar o conteúdo para o scrolling vertical, organizar o conteúdo em grupos do tamanho do ecrã, e usar técnicas de "zooming" para navegar entre dados com várias escalas.

2.3. Características das imagens usadas nos sistemas de Image Retrieval

Uma característica é definida pela captura de certas propriedades visuais de uma imagem. Em geral as caraterísticas de uma imagem podem ser globais ou locais. As características globais descrevem o conteúdo geral da imagem, enquanto características locais descrevem regiões (grupo de píxeis). A vantagem no uso das características globais é a rapidez, no entanto são rígidas, podendo descurar outras não tão dominantes na imagem. Por outro lado, as características locais apresentam uma efetividade maior pois representam conjuntos de pontos. Esta abordagem local é mais robusta, no entanto apresenta um gasto computacional maior.

2.3.1. Cor

A cor é uma das características largamente usada em sistemas de Image Retrieval pois é de fácil e rápida computação. É também uma característica intuitiva e importante para efetuar comparações. Os sistemas de Image Retrieval usam histogramas de cor, espaço de cores, vetores de coerência de cor, e descritores de cores dominantes. os histogramas de cor são os mais usados, que provado por Swain (Swain et al. 1991), são

muito mais poderosos que os de escala de cinza. Apesar das características de cor serem de fácil computação e de nos dar uma discriminação razoável da imagem, esta também tende a nos dar falsos positivos quando as coleções de imagem são muito grandes. De forma a incrementar o poder da análise e precisão surge então a ideia de dividir a imagem em sub-blocos, e fazer a extração das características de cor desses sub-blocos. Apesar da boa precisão, uma desvantagem é a fiabilidade da segmentação da imagem (Faloustos et al. 1994).

2.3.2. Textura

A Textura é uma propriedade da imagem que representa a superfície, a estrutura da imagem. Esta pode ser definida como regular e repetitiva de um elemento ou padrão. São padrões visualmente complexos compostos por regiões com outros sub-padrões com características de cor, brilho, forma, tamanho, etc.

2.3.3. Forma

Uma forma pode ser geralmente definida pela descrição de um objeto, independentemente da sua posição, orientação e tamanho na imagem. Assim, para um sistema de "Image Retrieval" eficiente, a rotação, translação ou escala não devem ter impacto no que diz respeito à identificação do objeto. No sentido de usar as formas como características de uma imagem é necessário determinar os limites da região ou do objeto (Zhang et al. 2002).

2.3.4. Localização no espaço

As localizações de determinadas características numa imagem têm também um papel importante e podem ser usadas para segmentar uma imagem. Esta localização pode ser descrita por Top/Bottom, Left/Right, Back/Front de acordo com a posição na imagem. Por exemplo, o céu e o mar podem ter as mesmas características de cor e textura, mas as suas características de localização são diferentes, o céu geralmente representa a parte superior (Top) da imagem, e o mar a parte inferior (Bottom) da imagem. Percebemos rapidamente que este tipo de características tem um papel importante nos sistemas de "Image Retrieval".

2.3.5. Image Retrieval em Redes Neurais

O IR pode ser categorizado em dois tipos: exato e relevante. O exact image retrieval refere-se a condições onde é necessário que a imagem seja 100% igual, enquanto que a "Relevant Image Retrieval" é baseado no conteúdo e há alguma flexibilidade na escala de similaridade.

Após o envolvimento das Redes Neurais Artificiais (RNAs, ANNs Artificial Neural Networks, ou simplesmente NN Neural Network), houve um ganho de desempenho significativo, pois as suas técnicas têm uma capacidade adaptativa muito grande e de melhor compreensão da imagem, sendo hoje em dia, o principal fator que está por trás dos sistemas CBIR (Yan et al. 2016).

Estas RNAs no fundo são sistemas de Machine Learing (ML) projetados para simular o comportamento do cérebro humano. Estes sistemas aprendem através de padrões encontrados para criar lógicas e regras para classificar as imagens. Por exemplo, podemos fornecer um conjunto de imagens (dados) com a referência de "gato", e a RNA irá construir regras para identificar aspetos em comum dentro de todo esse conjunto de imagens. A RNA será capaz de identificar imagens com "gato" sem que seja explicitamente indicado o que é um "gato". À medida que o volume de dados aumenta, as RNA vão-se tornando mais precisas, contemplando pequenas variações. À arquitetura desenhada para processar, relacionar, e perceber de forma eficiente uma grande quantidade de dados chama-se de Convolutional Neural Network (CNN), sendo também um dos modelos de "Deep Learning" mais populares, normalmente usada para classificação de imagens.

Apesar da primeira arquitetura CNN Lenet-5t ter surgido em 1998 (Lecun et al. 1998) com o objetivo de reconhecer dígitos escritos à mão, o grande interesse nas CNNs surgiu em 2012 com AlexNet Krizhevsky et al. (que usava 8 layers) e cresceu exponencialmente. Desde então apareceram a ZFNet (Zeiler e Fergus 2013), VGGNet (Simonyan et al. 2014), InceptionNet (Szegedy et al. 2014), GoogLeNet (Szegedy et al. 2015), a ResNet (He et al. 2015), R-CNN (Girshick et al. 2015), FractalNet (Lartsson et al. 2017), ou a Condesenet (Huang et al. 2018).

O próprio Krizhevsky realizou um estudo em 2017 demonstrando a evolução das CNN até à data.

Neural network	Top-1	Top-5	Number of layers	Number of operations (G-
AlexNet	39,7 %	18,9 %	8	70 M
ZF Net	37,50 %	14,8 %	8	70 M
VGG Net	25,60 %	8,10 %	19	155 M
GoogLeNet	29,00 %	9,20 %	22	10 M
Inception-v3	21,20 %	5,60 %	101	35 M
Inception-v4	20,00 %	5 %	152	35 M
Inception-ResNet-v2	19,90 %	4,90 %	467	65 M
ResNet-152	19,38 %	4,49 %	152	65 M

Figura 4 Evolução da taxa de erro na classificação de imagens - Krizhevsky (2017)

Os termos "Top-1" e "Top-5" são termos usados no que diz respeito à eficácia do algoritmo no processo de classificação. Isto é, das várias classificações prováveis que a CNN irá prever, a que tiver maior classificação (probabilidade) será a que estará no Top-1, e as cinco mais prováveis estarão no Top-5. Assim, a taxa de erro correspondente ao Top-1 é a percentagem de vezes que o modelo não identificou a classe mais provável como a classe correta, e a taxa de erro correspondente ao Top-5 significa que o modelo não incluiu a classe correta nas 5 classes com maior probabilidade que o modelo acha ser.

No que diz respeito à visão de uma imagem por parte de um sistema, esta é representada por uma matriz de valores de píxeis, sendo que uma imagem a preto e branco (greyscale) é apresentada por uma matriz 2D, em que cada posição representa um pixel da imagem, estando os valores entre 0 (preto) e 255 (branco)

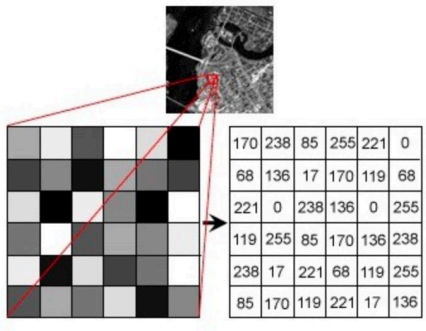

Figura 5 Matriz Representativa - Greyscale [4]

[4] Stanford, https://ai.stanford.edu/~syyeung/cvweb/tutorial1.html, Data do último acesso: 8/2/2021

No caso de a imagem ser colorida, esta é representada por uma matriz 3D, de forma a poder armazenar a combinação das cores. Para o caso de a imagem ser representada por RGB, a matriz terá a profundidade 3, de forma a poder armazenar os valores de "Red", "Green" e "Blue". Caso seja usado outro espaço de cores a profundidade da matriz poderá ser diferente, para o CMYK seria de 4 por exemplo ("Cyan", "Magenta", "Yellow", "Black").

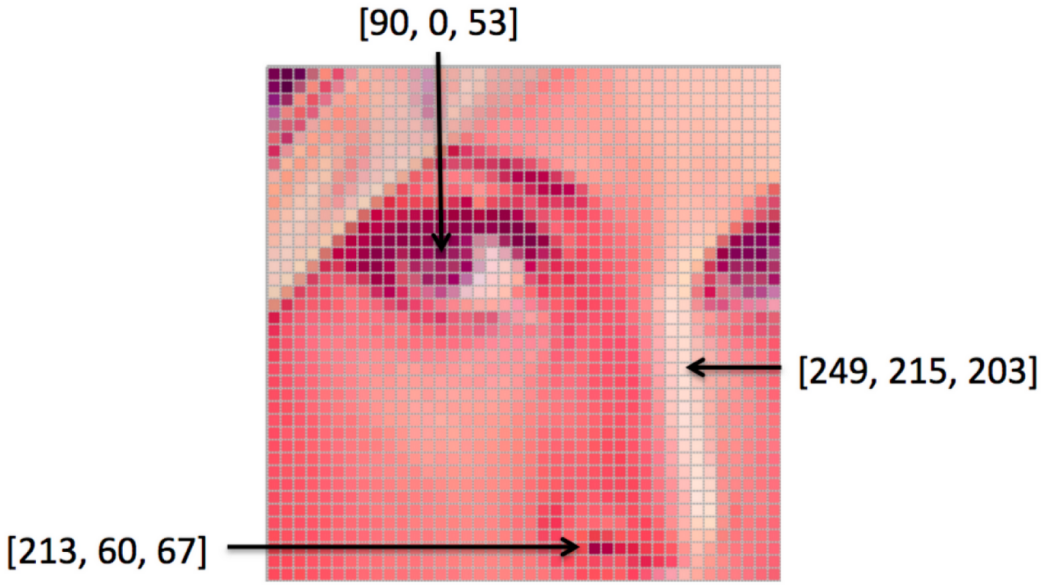

Figura 6 - Matriz Representativa RGB[5]

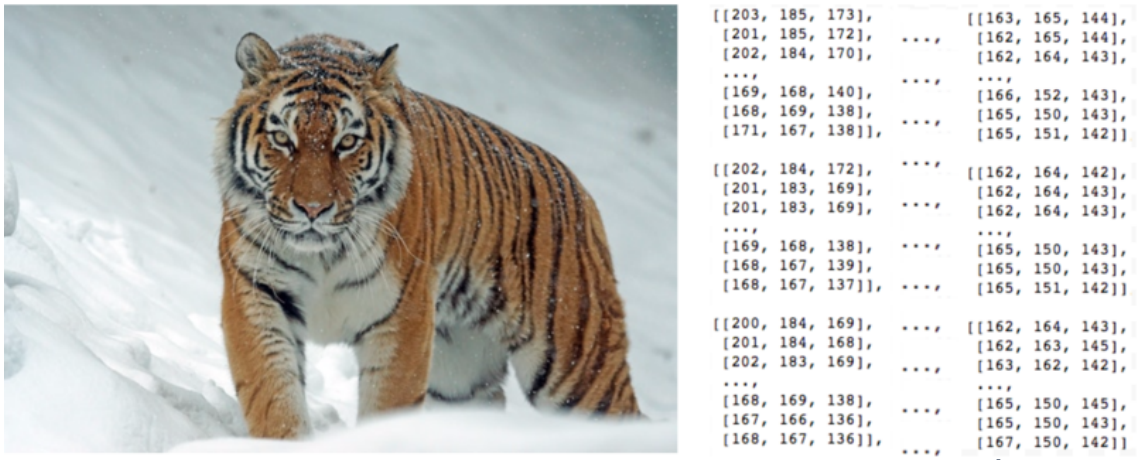

Figura 7 - Imagem vista por humanos (à esquerda), vista por um computador (à direita)[5]

Aqui entram os princípios de processamento de imagem, através de diversos algoritmos que podemos usar, dependendo do objetivo, como são as CNNs para a extração de características.

[5] Stanford, fonte: https://ai.stanford.edu/~syyeung/cvweb/tutorial1.html, Data do último acesso: 8/2/2021

Figura 8 - Exemplo de extração de caraterísticas[6]

A extração de características através de uma CNN é constituída por: Convoluções, Padding, Relu e Pooling.

As Convoluções são operações de somatório do produto de duas funções ao longo da região em que elas se sobrepõem, em razão do deslocamento existente entre elas. Uma vez que temos 2 dimensões (altura e comprimento) são necessários 2 somatórios.

$$(f * g)(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f(i, j) g(x - i, y - j)$$

Em relação às imagens, esse processo diz respeito a um filtro (kernel), que aplicado à imagem de input, transforma-a. Tem o papel de fazer filtragem (extração) de informações de interesse na imagem.

Então, o kernel é uma matriz utilizada para uma operação de matrizes, podendo ser aplicada várias vezes em diferentes regiões da imagem. Durante o deslizamento do kernel ao longo da imagem (como uma janela móvel) são multiplicados e somados os valores sobrepostos. Ao número de pixéis que a janela avança (passo) é dado no nome de "stride".

Assim, para uma imagem NxN, um kernel FxF e com "stride" S, o resultado da Convolução será uma matriz G tal que:

[6] Medium, https://medium.com/@gilneyjnr/extracao -de-caracteristicas-ciencia-de-dados-dd041bcff72b, Data do último acesso: 17/02/2021

$$G = \frac{N - F}{S} + 1$$

Assim tendo em conta o exemplo de uma imagem 7x7, aplicando um filtro (kernel) de 3x3 com "stride" 1, obtemos uma imagem 5x5:

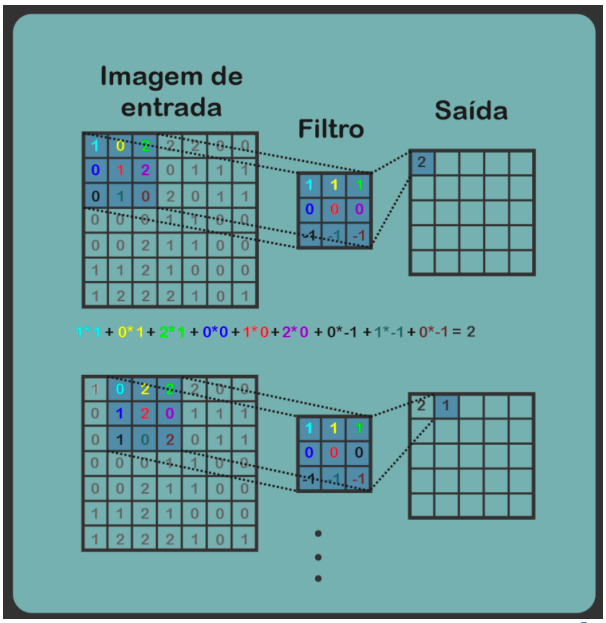

Figura 9 - Operação de Convolução de Filtro em imagem[7]

Neste exemplo aplicando o raciocínio descrito, temos então o seguinte cálculo:
(1*1)+(0*1)+(2*1)+(0*0)+(1*0)+(2*0)+(0*(-1))+(1*(-1))+(0*(-1)) = 2

Facilmente percebemos que a imagem "Convoluída" será menor que a imagem original.

Se não for conveniente esta redução da imagem original temos que aplicar um processo chamado de "Padding", que não é mais do que a inclusão de pixéis ao redor da imagem original. Para manter a imagem original com o mesmo tamanho devemos adicionar a quantidade de pixéis nas bordas de acordo com a seguinte fórmula:

$$P = \frac{F - 1}{2}$$

Voltando ao exemplo da Figura 9, percebe-se então, que para manter o tamanho original da imagem, o Padding deve ter o valor de 1, ou seja, deve ser adicionado 1

[7] Medium, https://medium.com/turing-talks/visão-computacional-o-que-é-convolução-ad709f7bd6b0, Data do último acesso 17/02/2021)

píxel às bordas da imagem. Geralmente são adicionados píxeis com o valor "0" ("zero-padding") de forma a minimizar o efeito de redução da dimensionalidade espacial da imagem convoluída. A imagem de exemplo ficaria então com a dimensão de 9x9.

A Convolução tem então o papel de fazer uma filtragem para a extração de informações de interesse na imagem. Dependendo dos valores usados nessa matriz de kernel é possível saber características diferentes.

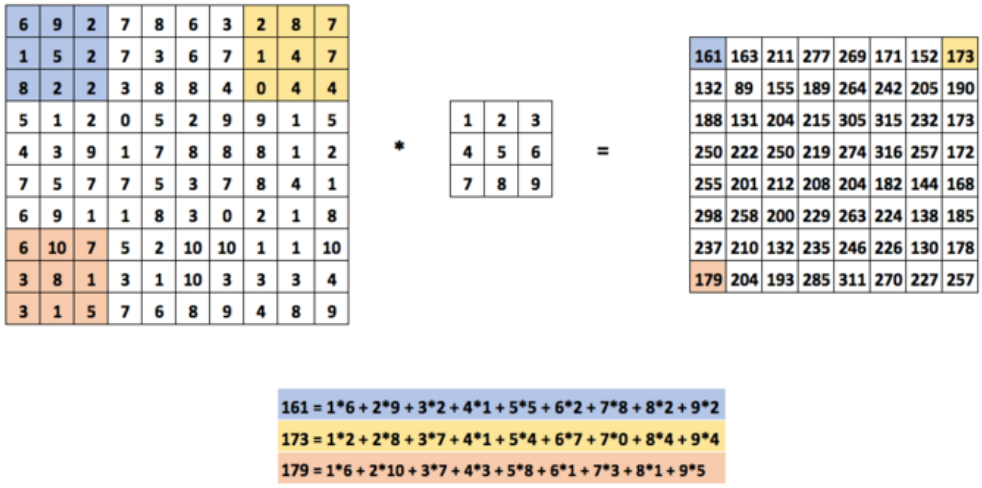

Figura 10 - Exemplo de uma convolução completa[8]

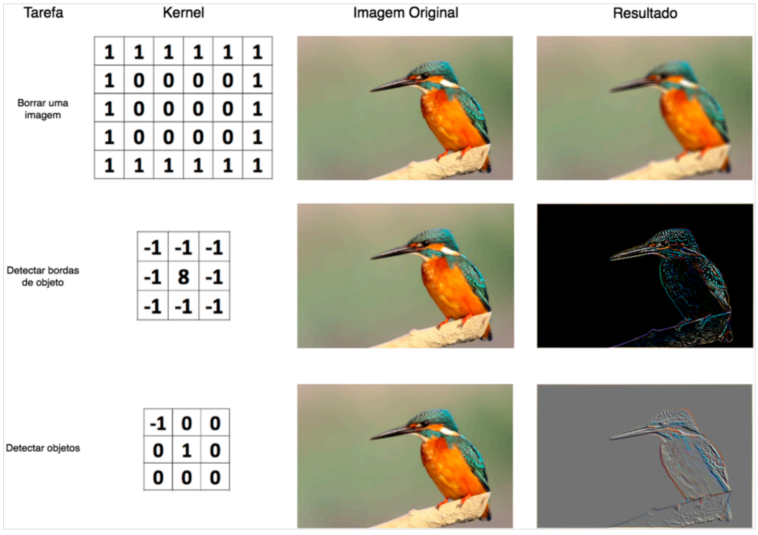

Figura 11 - Exemplos de kernels tradicionais[9]

[8] Medium, https://medium.com/neuronio-br/entendendo-redes-convolucionais-cnns-d10359f21184, Data do último acesso: 27/02/2021

[9] Medium, https://medium.com/data-hackers/neural-network-deep-learning-parte-1-introducaoo-teorica-5c6dcd2e5a79, Data do último acesso: 17/02/2021

2.4. Query by sketch ou Sketch-Based Image Retrieval (SBIR)

Os esboços têm sido usados para registar e ilustrar objetos e pessoas com base na perceção humana (Zhang et al. 2019). O Sketch Based Image Retrieval consiste em retornar imagens com base num esboço (desenho). Esses resultados podem ser de vários tipos como imagens, clip-art, ou também desenhos e esboços. Com o desenvolvimento dos dispositivos digitais tácteis, a pesquisa por esboços tornou-se recentemente uma área ativa de Visão Computacional. Vários tipos de esboços diferem na riqueza de detalhe ou na maneira como foram produzidos. Assim, todos os esboços são únicos e como resultado das tarefas de reconhecimento encontram ainda mais desafios que (Zhang et al. 2019) são os seguintes:

- Grandes variações (diferenças) entre esboços individuais. Isto é, apesar de um esboço ser representação básica de um conceito ou silhueta, os "desenhadores" podem usar vários estilos, ter capacidades artísticas diferentes, ou até mesmo personalidades diferentes, contribuem para uma representação diferente.
- Um esboço e uma fotografia são formas de *média* diferentes, tornando mais desafiador do que o reconhecimento de dois objetos do mesmo domínio, como por exemplo duas fotografias.
- A representação dos esboços é mais inconsistente. O desenho livre de esboços é gerado de acordo com as preferências do utilizador, e consistem apenas em poucos traços podendo também conter poucas cores. Assim torna-se difícil identificar caraterísticas relevantes devido à falta de informação.
- Problemas relacionados com o uso de memória e complexidade computacional ao investigar o SBIR em grande escala.

O SBIR tem sido estudado desde os anos 1990's, e desde então foram desenvolvidas frameworks limitadas a determinadas categorias como por exemplo para verificar marcas registadas (Shih et al. 2001), uma Framework com o objetivo de encontrar um documento através do esboço caligráfico (Fonseca et al. 2005), ou esboço devolvia um "cartoon" (Wang et al. 2005).

Enquanto o SBIR devolve imagens da mesma categoria o FG-SBIR (Fine-Grained Sketch Image Retrieval) tenta devolver fotografias relacionadas com o "sketch". Pode-

se afirmar que o SBIR é um método ao nível da categoria, e o FG-SBIR é um método ao nível de instância.

Como já referido anteriormente, uma série de diferentes tipos de "queries" têm sido trabalhados, tais como texto, exemplos de imagem, e os inputs por "sketch" são também uma forma de interagir com o sistema. Estes últimos são únicos pois são capazes de capturar a topologia espacial de um objeto visual e os seus detalhes. Estes sistemas podem ser de extrema utilidade para artistas, designer, ilustradores, bloggers ou alguém que queira expressar uma ideia com uma imagem.

2.5. Aplicações móveis existentes de Image Retrieval

Hoje em dia existem inúmeras aplicações para mobile tanto para dispositivos iOS como Android que permitem efetuar pesquisas por imagens. Se o input for texto, os motores de busca populares (como o Google ou Bing) são os mais óbvios e procurados pelos utilizadores. No caso de o input ser uma imagem, os utilizadores poderão ter muitas alternativas, e algumas estão aperfeiçoadas com modelos de ML treinados para identificar conteúdo de uma determinada categoria, como por exemplo a aplicação Plant Identifier na identificação de plantas, o Picture Mushroom na identificação de cogumelos, ou o Dog Scanner para a identificação de raças de cães. Outras aplicações móveis têm uma abordagem mais global como Veracity, Pinterest, Reversee, ou o Google Lens. De salientar que estas ferramentas possuem o tipo de input tradicional de upload de uma imagem através da câmara ou do dispositivo. Em termos de output a aplicação Google Lens permite uma amplitude maior de resultados comparativamente às restantes aplicações, desde imagens semelhantes, deteção de texto e respetiva tradução, sugestão de lojas de venda dos itens identificados, informações sobre objetos detetados, ou informações sobre uma localização ou monumento.

Este protótipo propõe, para além das formas tradicionais de input referidas anteriormente, a capacidade do utilizador criar, ou editar/alterar uma imagem existente, efetuando a query a cada momento de interação.

2.6. Computer Vision API's

Dada a procura por este tipo de serviços de reconhecimento de imagens, as grandes empresas de tecnologia estão a apostar no desenvolvimento de plataformas relacionadas

com a visão computacional, sendo uma área com bastante expansão no mercado empresarial. A Microsoft através da sua Computer Vision API, a Google com a Google Cloud Vision API, ou a Amazon com a Amazon Rekognition, são as grandes empresas a apostar no desenvolvimento deste tipo de tecnologias. Embora todas estas soluções disponham de utilização gratuita durante um determinado tempo, optou-se por escolher a plataforma da Google ser a mais robusta, de mais fácil utilização, e pertencer a uma empresa responsável pelo processamento de 92% das pesquisas a nível mundial[10] razões pelas quais será aprofundada um pouco mais de seguida.

A Google Cloud Platform é uma suite que oferece vários serviços na Cloud. No que diz respeito aos serviços de Visão Computacional (google.cloud.com), são oferecidos dois: AutoML Vision e a API Vision. Esta plataforma é muito robusta e complexa e com alta escalabilidade.

O AutoML Vision serve para criar e treinar modelos próprios de Machine Learning, enquanto que a API Vision, é uma REST API, que já oferece modelos pré-treinados permitindo-nos integrar vários recursos nas aplicações como por exemplo:

- Deteção de rótulos (Label Detection): informações de diversas categorias dentro de uma imagem. Podem ser identificados objetos gerais, locais, atividades, espécies animais, produtos, paisagens, etc.
- Deteção de objetos: permite detetar e extrair vários objetos de uma imagem. Pode ser feita a identificação na própria imagem através de limites retangulares na região da imagem que contém esse objeto.
- Encontra itens semelhantes
- Reconhecimento Óptico de Carateres (OCR): Permite a deteção e extração de texto, escrito à mão ou não, de uma imagem.
- Deteção de Rostos (Face Detection): detecta os rostos presentes em uma imagem, bem como o estado emocional.
- Identifica locais (Landmark Detection): identifica estruturas famosas, naturais ou contruídas pelo homem. Também indica as suas georreferenciações e logotipos de produtos conhecidos
- Deteção de Atributos Gerais (Image Proprieties): identifica as cores presentes e dominantes

[10] Statcounter, https://gs.statcounter.com/search-engine-market-share, Data do último acesso, 14/08/2020

- Deteção de conteúdo explícito (Safe Search): identifica conteúdo adulto ou violento em uma imagem.

De referir que imagens onde a confiança seja abaixo dos 50% não fazem parte dos resultados e por isso não são exibidas ao utilizador.

No que diz respeito à confiança dos resultados da Google Cloud Vision, relativamente à obtenção das labels, os estudos realizados (Chen et al. 2017) indicam que estes são positivos tendo um grau de precisão relativamente alto (acima dos 64%). O estudo baseou-se em um dataset de 4972 imagens de um conjunto restrito de categorias, obtendo as respetivas "labels", e onde a estas foram comparadas com uma base de dados existente através da plataforma Wordnet, responsável por fazer a relação léxica com essas labels encontradas.

3. Protótipo

Neste capítulo será feita uma introdução da arquitetura de sistema do protótipo, bem como do hardware e das "frameworks" utilizadas. É feita uma explicação da evolução e implementação das diversas funcionalidades, bem como a demonstração de algumas interações.

3.1. Arquitetura do sistema

Nesta subsecção será apresentado um diagrama que corresponde à arquitetura de sistema do protótipo desenvolvido durante a dissertação, bem como o hardware utilizado e o ambiente de desenvolvimento.

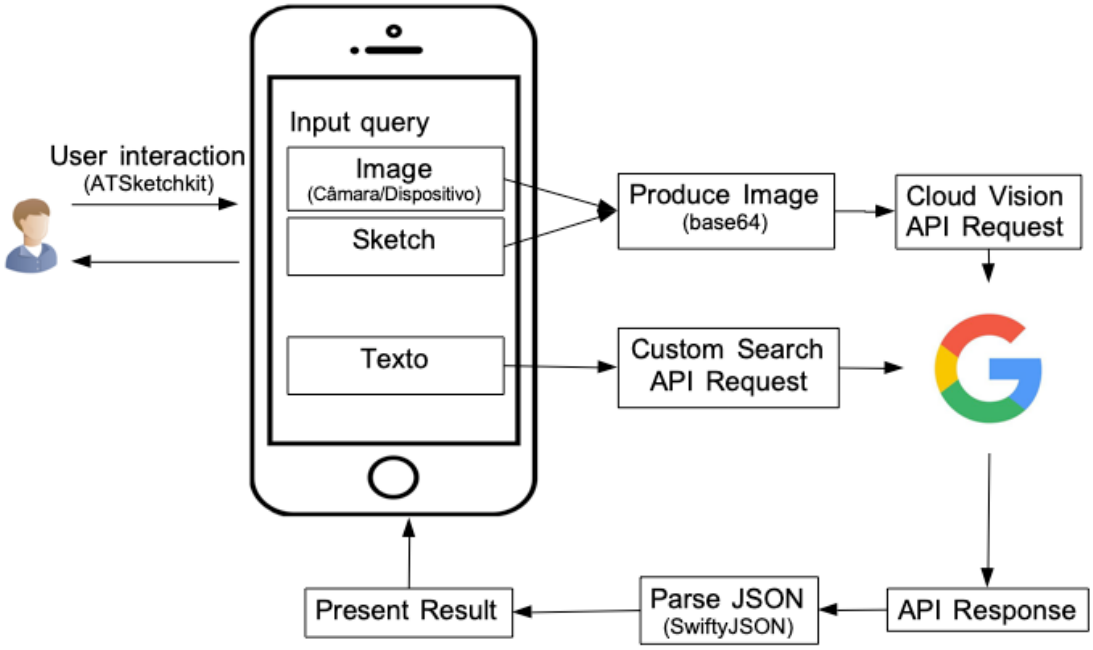

Figura 12 - Diagrama de Arquitetura de Sistema

Para a realização deste protótipo optou-se por usar a tecnologia disponibilizada pela Apple, nomeadamente a linguagem Swift através do IDE Xcode, e para testagem do protótipo foi utilizado um iPhone 7. Este equipamento, tal como todos smartphones da

marca Apple, tem um sistema operativo iOS. Este equipamento tem o display de 4.7’ com uma resolução de 1334x750. Dispõe de câmara fotográfica de 12 Mpx, 32 GB de memória interna, e processador Quad-Core de 2 GHZ. Estes requisitos são suficientes para a finalidade do protótipo. O tamanho do ecrã enquadra-se no tamanho alvo de estudo, uma vez que é também um dos propósitos do trabalho que a interface seja enquadrada para um display de tamanho reduzido, comparativamente a um tablet ou um computador.

Foram vários os motivos que pesaram na tomada de decisão por um sistema Apple, nomeadamente:

- Acesso a iPhone e também a um Macbook (imprescindível para a criação de aplicações para iOS usando a linguagem Swift);
- O sistema iOS é fechado e menos suscetível a erros;
- A existência de uma linguagem de desenvolvimento nativa (Swift) fortemente documentada. Esta linguagem veio substituir a linguagem Objective-C, que já tinha um forte historial (desde 1984), trazendo mais simplicidade e flexibilidade.
- A compatibilidade entre os dispositivos Apple é muito forte. Isto potencia a redução do número de problemas e no tempo para a resolução dos mesmos. A parametrização/configuração dos equipamentos para o ambiente de desenvolvimento é pequena.
- Os equipamentos Apple estão fortemente consolidados em sistemas de desenho, sendo estes os dispositivos preferencialmente escolhidos por designers e criativos;
- É um mercado atrativo para desenvolvedores, que, através das competências ganhas no desenvolvimento deste protótipo, contribuem futuramente em termos profissionais.

Figura 13 - iPhone 7

Com já referido anteriormente, para o desenvolvimento concretamente dito, foi usado o Xcode. Este, apesar de já não possuir o simulador para o iPhone 7, dispõe ainda do simulador do iPhone 8 (tem as mesmas características de tamanho de ecrã e resolução) e que permitiu com que as funcionalidades fossem testadas de forma virtual sem ter que enviar a *app* para o equipamento físico. Para fazer este envio da *app* para o iPhone, dada a tal compatibilidade entre os sistemas, foi apenas necessário ligar o cabo *lightning* para USB, sem ter que efetuar qualquer tipo de parametrização/configuração ou instalação de drivers. É possível fazer este envio também via "wireless" bastando apenas que ambos os equipamentos (iPhone e Macbook) estejam na mesma rede, no entanto o processo é mais demorado.

3.2. Frameworks e API's usadas

Nesta subsecção serão apresentadas Framworks e API's utilizadas na implementação do protótipo.

3.2.1. ATSketchkit

Como base para a construção do protótipo usou-se a Framework ATSketchkit. Esta Framework já dispõe das funções de desenho no ecrã, a alteração da cor e espessura da linha, borracha, e as funções de "undo", "redo" e "clear". Em cada toque/interação é

criada uma nova "layer". Também contém uma função que permite produzir uma imagem, isto é, permitia "renderizar" o conteúdo da imagem existente na tela. Esta função foi importante para implementar a função de "live search" na pesquisa de imagens.

A framework já dispunha de uma interface de exemplo que também foi usada numa fase inicial, e posteriormente reformulada de acordo com as necessidades e evolução do protótipo.

3.2.2. Google Cloud Vision

De forma a executar as pesquisas através de imagem, utilizou-se a Google Cloud Vision API. Para poder usar a API é necessário obter uma "Chave" que nos é fornecida após efetuar o registo de um projeto numa conta Google, ativar a faturação, ativar a API, e configurar o tipo de autenticação. A cada solicitação é enviada a informação da imagem juntamente com essa chave. De realçar que este serviço é gratuito durante 1 ano, mas restrito a 1000 pedidos por mês. O custo médio após as primeiras 1000 pesquisas, por recurso, é de US$ 1,5. As solicitações gratuitas foram suficientes para o desenvolvimento e realização de testes com utilizadores.

Este sistema já possui alguns recursos embora não fossem todos implementados. Dos recursos disponíveis foram implementados: a deteção de imagens semelhantes; e a deteção de "labels" (descritores das imagens).

3.2.3. Google Custom Search API

Esta API permite efetuar chamadas personalizadas ao motor de busca tradicional Google, sendo possível retornar apenas imagens de acordo com as palavras-chave introduzidas. São usadas solicitações RESTFul para obter os resultados no formato JSON.

Esta API é gratuita para 100 solicitações diárias, que quando esgotadas custam US$ 5 por cada 1000 solicitações. Para o caso de apenas usarmos pesquisas num máximo de 10 sites, a API não tem limite de solicitações. Tal como na Cloud Vision API, as solicitações gratuitas foram suficientes para o desenvolvimento do protótipo, bem como para a realização de testes com os utilizadores.

3.2.4. SwiftyJSON

De forma a conseguir trabalhar melhor com os resultados das "queries" usou-se a biblioteca SwiftyJSON. Esta biblioteca ajuda a interpretar JSON retornado pelo pedido do Google Cloud Vision, conseguindo de forma mais fácil identificar as chaves e os valores desse output.

3.2.5. SDWebimage

Esta biblioteca tem recursos que permitem descarregar imagens através do URL. Uma vez que os resultados das imagens semelhantes obtidos pela Google Cloud Vision API é uma "string" com o URL da imagem, foi necessária uma ferramenta que "descarregasse" a respetiva imagem.

Esta biblioteca tem a vantagem de conseguir gerir a cache automaticamente no que diz respeito a este tipo de pedidos, fazendo com que endereços já consultados anteriormente fossem guardados, permitindo uma melhoria da resposta na apresentação das imagens. Às imagens que por algum motivo não foi possível descarregar, foi possível também definir uma imagem por defeito.

3.3. Desenvolvimento

O desenvolvimento do protótipo foi feito em simultâneo com a pesquisa e de forma faseada, tentando atingir pequenos objetivos, por vezes de forma independente, e depois combinando essas implementações com o protótipo.

Sabendo a existência de frameworks disponíveis que permitiam usufruir das funcionalidades básicas de desenho, fez-se uma abordagem inicial de identificação e testagem das mesmas.

Entre várias frameworks optou-se por usar a Framework ATSketchkit. Foi importante perceber o funcionamento da mesma, analisando a documentação e o próprio código, de forma a entender que funções já estariam implementadas e de que forma. Esta Framework já dispunha de funções tais como o desenho pelo toque

conseguindo alterar a espessura e a cor, o "undo" e o "redo", e guardar a imagem. Estas funcionalidades foram fundamentais para que o protótipo pudesse arrancar pois seriam a base para a criação/edição de imagens.

O passo seguinte foi o da configuração e parametrização da Google Cloud Vision API. As plataformas de suporte da Google fornecem bons exemplos que me permitiram perceber primeiro como ativar e disponibilizar o serviço, e depois de como implementar a conexão/solicitação. Foi inicialmente testada em um projeto mais simples e independente, onde o "request" era efetuado com o upload de uma imagem já existente no dispositivo, e obtendo depois a resposta. Para tal é necessário fazer o *encode* da imagem para o tipo *base64*, pois a API apenas recebe o input da imagem nesse formato. Após perceber o funcionamento da API, esta foi integrada com sucesso no protótipo, também de forma faseada.

Uma vez que uma das funcionalidades importantes para o protótipo era a de editar/alterar uma imagem existente, foi incorporada essa funcionalidade de fazer o upload de uma imagem no ecrã, e enquadrá-la. Para isto foi necessário perceber a forma como eram tratadas as interações da Framework ATSketchkit. Então, percebeu-se que são usadas *layers* do tipo CALayer (classe que ajuda a controlar conteúdo baseado em imagem), e que a imagem carregada deveria ser colocada sempre na *layer* mais baixa de todas.

A solicitação à Google Cloud Vision API foi testada inicialmente através da interação com um botão. Esse método de solicitação foi posteriormente alterado para que, através de qualquer interação do utilizador no que diz respeito à edição da imagem, o "request" fosse realizado, isto é, quando o utilizador finaliza o toque na tela ou efetua o upload de uma imagem. A esta funcionalidade chamei de "Live Search". Foi pensado alterar o estado desta funcionalidade entre Ligado/Desligado, mas tratando-se de uma funcionalidade diferenciadora, optei por deixar sempre ativa.

Uma vez que os resultados obtidos vêm em formato JSON, o passo seguinte foi implementar a biblioteca SwiftyJSON que permitiu, de forma mais fácil, analisar e trabalhar os resultados específicos de determinadas áreas da listagem.

Como os resultados obtidos são em texto, estes resultados que indicam as imagens similares às do request são o URL dessas imagens. Para ajudar a apresentar essas imagens similares foi implementada a biblioteca SDWebimage. Desta forma foi possível através do URL, apresentar os resultados das imagens similares.

Foram adicionadas as funcionalidades de limpar a tela (“clear”) através de um botão do lado superior esquerdo, os botões de adicionar imagem e partilhar/exportar o conteúdo da imagem através de botões do lado superior direito.

Uma vez que o tamanho do ecrã tem imenso valor, foi sempre tido em consideração o tamanho usado para as mais diversas funcionalidades.

Na realização do protótipo foi tido em conta o padrão de arquitetura “Model-View-Controller” (MVC). Este padrão permite separar a representação da informação da interação com o utilizador de acordo com as três partes: Model, View, e Controller.

Apesar da lógica ser a mesma, a Apple propõe um padrão ligeiramente diferente do padrão MVC tradicional.

Segundo o padrão MVC tradicional:

- o “Model” define a estrutura dos dados, atualizando o sistema de acordo com as ações do utilizador aquando da sua interação com o sistema, isto é, recebe os comandos do “Controller” e envia para a “View”.
- A “View” é a camada que define a apresentação dos dados ao utilizador.
- A camada “Controller” é a intermediária entre as camadas “Model” e “View”, isto é, faz solicitações à camada “Model”, e notifica a camada “View”.

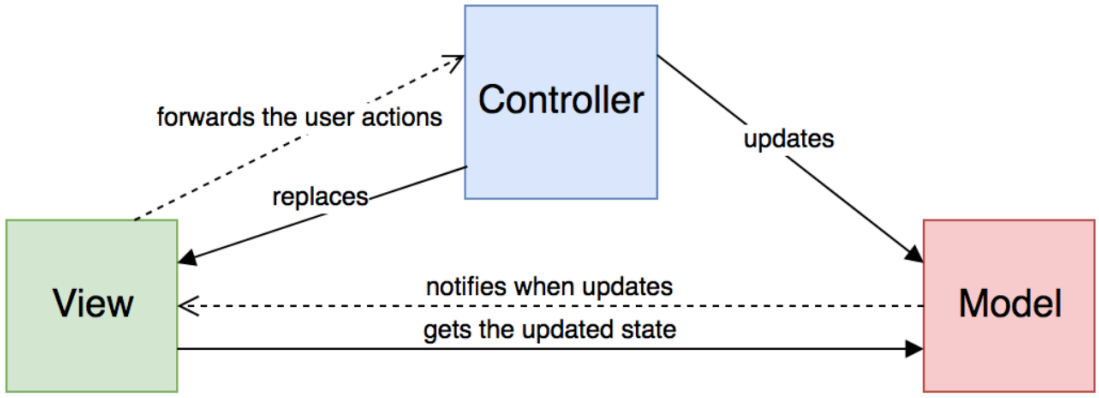

Figura 14 - Diagrama MVC Trdicional

Segundo o padrão Apple MVC (ou Cocoa MVC):

A Apple propõe uma variação no fluxo de ações onde os eventos são emitidos diretamente pela View. Isto coloca o componente Controller como intermediador, que

observa os eventos da View por meio de “delegates” (serão abordados mais à frente) ou outros mecanismos, e passa-os para o Model, que é atualizado. Este notifica de volta o Controller que por sua vez informa a View para ser atualizada. Deste modo, diferindo do MVC tradicional, na arquitetura Apple MVC, View e Model não se conhecem.

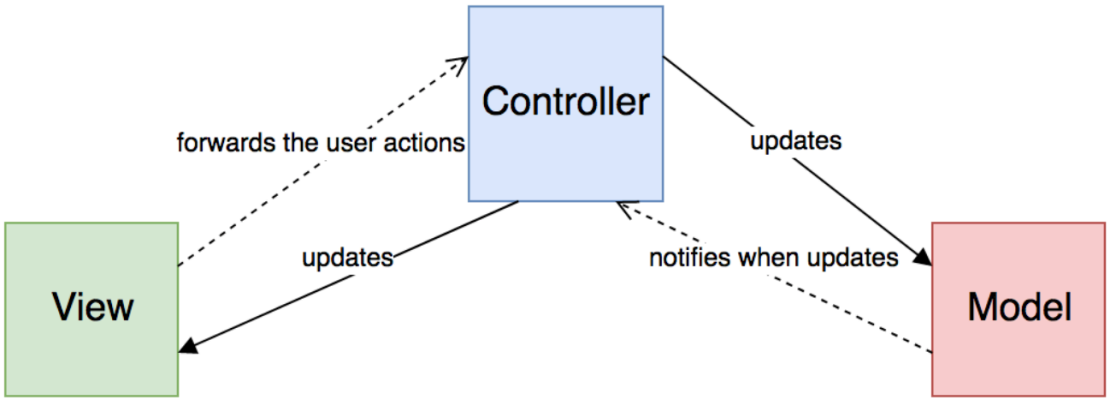

Figura 15 - Diagrama Apple MVC

Como a View e o Model não comunicam diretamente, a camada de formatação e apresentação, a gestão de eventos, networking, etc, são delegados ao Controller.

Este padrão Apple MVC encoraja a escrita e uso de UIViewControllers o que na prática torna o Controller altamente acoplado à View.

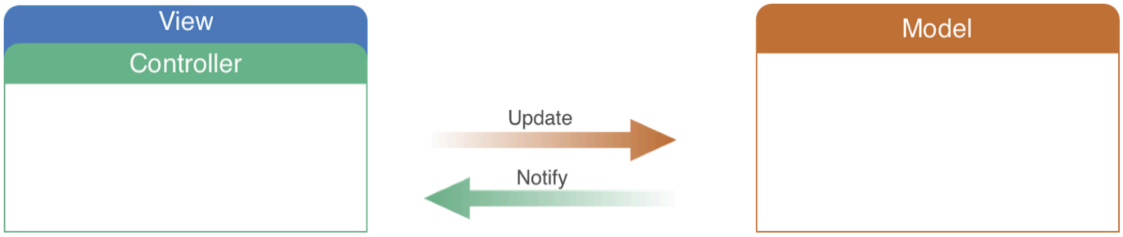

Figura 16 - Diagrama Apple MVC (Comunicação)

3.4. Interface e interação

A interface deve ter em conta o tamanho de ecrã disponível. Para este protótipo, a interface foi pensada para tamanhos de ecrã de smartphones. Apesar do protótipo funcionar também em dispositivos móveis com ecrã maior, como tablets, estes no entanto, não foram alvo de estudo pois o que se pretendia era testar em dispositivos de fácil acesso que todos dispomos, e que são de tamanho reduzido.

Dada a importância da área de desenho (canvas), tentou-se que houvesse pouca invasão deste espaço como botões ou outras informações (Figura 17).

A interface foi sofrendo alterações de modo a melhorar a interação com o utilizador (Figura 18). Foram tidos em conta o espaçamento entre os botões e o tipo de ilustração representativa dos botões para que fosse mais intuitivo. Por exemplo, numa primeira fase foi usado o símbolo "+" para representar a ação de adicionar imagem, mas depois, foi substituída por um ícone uma câmara fotográfica. Uma vez que foi colocada a barra de pesquisa por palavra-chave no topo do ecrã, de forma a dar a sensação de maior amplitude de tela, foi o de retirado o fundo cinza do botão inferior que permite abrir as opções de edição de imagem. De realçar que todos os ícones usados na elaboração do protótipo são nativos da plataforma de desenvolvimento.

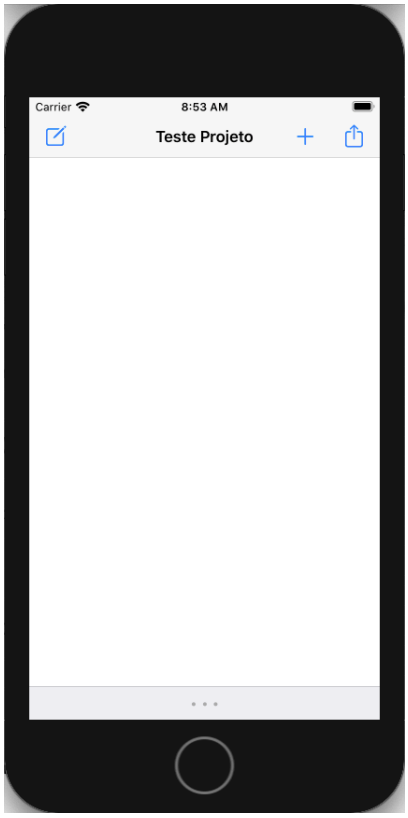

Figura 17 - Tela Limpa (Fase inicial)

Figura 18 - Tela limpa (Fase Final)

Os botões de interação para pesquisar imagens através de palavras-chave, ou carregar imagens na tela provenientes da câmara ou existentes no dispositivo foram colocados na parte superior, bem como o botão de “limpar a tela”. Este último foi colocado no lado oposto aos anteriores e isolado, uma vez tem uma função diferente e o seu uso por engano poderia provocar a perda de um eventual trabalho realizado.

O campo para a pesquisa através de palavra-chave foi colocado também na parte superior e está sempre disponível como é possível constatar na Figura 19.

É possível ajustar algumas opções na ferramenta de criação de sketches como a cor do lápis e a espessura do lápis e da borracha, como se verifica na Figura 20. Estas ferramentas permitem também a edição/alteração de uma imagem anteriormente carregada, como pintar ou apagar, anular ou refazer uma ação. Assim estas ferramentas para edição/criação (lápis, borracha, undo e redo) estão por defeito ocultas, ficando visíveis apenas aquando de uma interação do utilizador com um botão na parte inferior do ecrã.

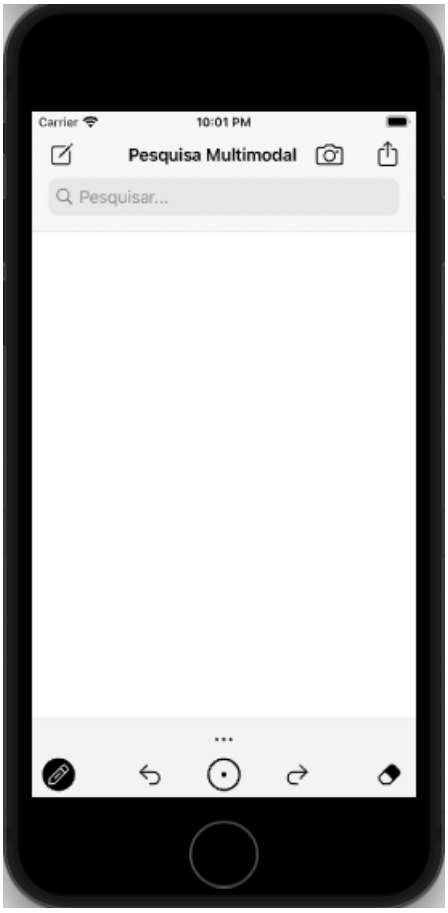

Figura 19 - Funções de edição visíveis

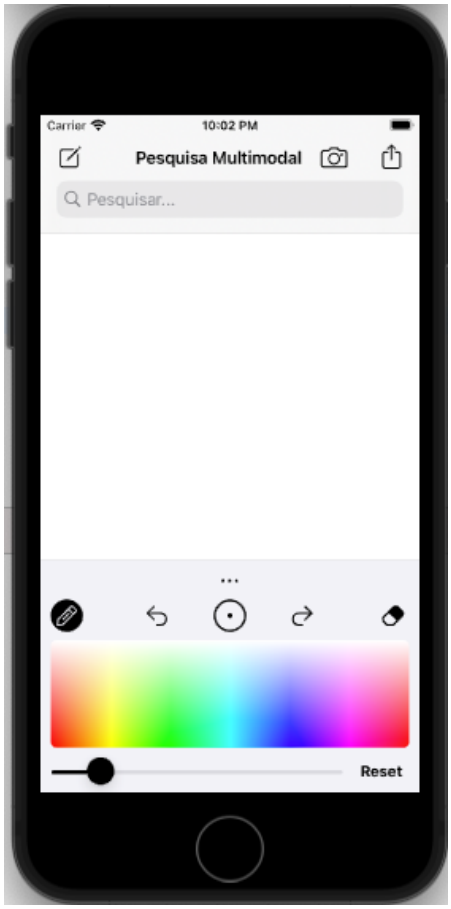

Figura 20 - Definição de espessura e cor da linha

Foi decidido também colocar uma barra de "preview" com "thumbnails" dos resultados no topo do ecrã, de forma a permitir ao utilizador identificar se os resultados obtidos são do seu interesse. Após clicar nessa barra de "preview" o utilizador passa então ao "screen" de visualização de resultados.

Foi colocada uma barra a apresentar as "labels" identificadas na imagem de input. Esta barra pode ser deslocada na horizontal de forma a poder consultar todos os resultados das "labels" encontrados. Ao clicar numa "label" é feita uma pesquisa respetiva por palavra-chave, mostrando novos resultados.

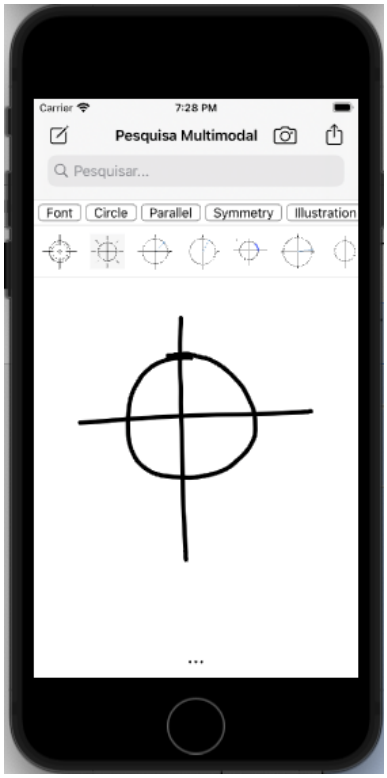

Figura 21 - Exemplo de pesquisa por desenho (barra de labels e thumbnails)

É possível expandir os resultados de imagens obtidos em grelha (Figura 23), ficando os mais relevantes (de acordo com a classificação da Cloud Vision API) em cima e com tamanho maior.

Os resultados obtidos por qualquer tipo de input, sejam eles por palavra-chave, desenho, ou importação de imagem da câmara ou do dispositivo, permitem ser reutilizados, substituindo a imagem anteriormente através de um "long press" em cima da respetiva imagem.

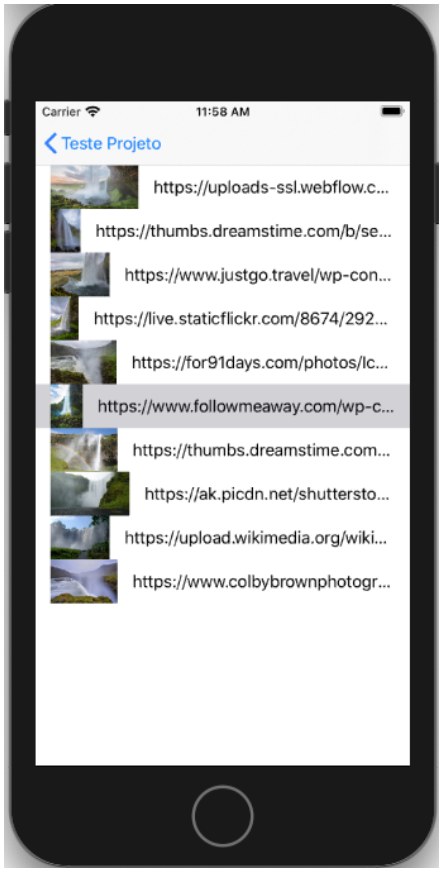

Figura 22 - Imagens Similares (Fase Inicial)

Figura 23 - Imagens Similares (Fase Final)

Uma vez que a catalogação das “labels” é em inglês, a pesquisa por palavra-chave deve ser realizada nesse idioma.

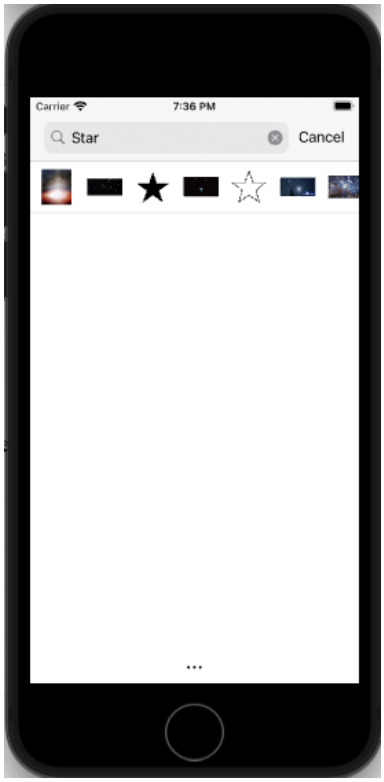

Figura 24 - Pesquisa por palavra-chave

Figura 25 - Preview dos resultados da pesquisa por palavra-chave

Assim, o protótipo permite efetuar pesquisas por imagens nas modalidades (*input*) de Texto, Imagem e Desenho. Os resultados das pesquisas são apresentados sempre que o utilizador interage com o sistema aquando da criação de conteúdo, sendo possível usar esses resultados e continuar a pesquisar a partir destes. Este sistema foi implementado com sucesso para dispositivos móveis de ecrã reduzido (*Smartphones*) para a iOS. Foi utilizada a Framework ATSketchkit com funcionalidades de desenho, e para a pesquisa de imagens as API's Google Cloud Vision quando o *input* é uma imagem ou um desenho, e a Google Custom Search quando o *input* é texto.

4. Avaliação

Neste capítulo serão apresentadas as metodologias de avaliação, bem como o estudo dos resultados obtidos, e uma reflexão dos mesmos de forma a aferir qual o método de pesquisa com melhor e mais fácil interação, e perceber se a apresentação dos resultados em tempo real ("Live Search") ajuda nas pesquisas.

4.1. Metodologia

Para a realização dos testes ao protótipo foram definidas tarefas para que o utilizador interagisse com todas as modalidades de pesquisa do sistema. Foi feita uma avaliação usando as escalas System Usability Scale (SUS) (Brooke, 1996)e Creativity Support Index (CSI) (Cherry e Latulipe 2014). O SUS é um conjunto de dez perguntas relacionadas com a usabilidade do sistema que são colocadas numa escala de 1 a 5, sendo o valor 1 correspondente a uma discordância total, o valor 3 uma posição neutra, e o valor 5 indica uma concordância total. O CSI tem o mesmo princípio e estuda um conjunto de 6 fatores: Colaboração, Satisfação, Pesquisa, Expressividade, Imersão e Compensação pelo Esforço. Para obter as métricas desta escala são realizadas 12 questões numa escala de 0 a 10, e de mais 15 questões com duas hipóteses de acordo com vários emparelhamentos dos fatores. Esta escala reflete o quão a ferramenta apoia a criatividade para uma determinada tarefa ou atividade em que o utilizador estava envolvido, e que depende das preferências individuais e do nível de conhecimento com a ferramenta. Apesar desta escala ser aconselhável aplicar sempre que o utilizador termina uma tarefa testando uma determinada funcionalidade ou modo (tal como no método SUS), uma vez que se pretendia ter uma visão geral do sistema, esta escala foi aplicada apenas no fim dos testes.

Foi ainda realizado um pequeno questionário de forma a aferir diretamente opiniões relativas ao sistema e de autoavaliação como a experiência no uso de dispositivos móveis, e experiência no manuseamento de sistemas de pesquisa de imagens e desenho.

No início de cada teste foi realizado um questionário de forma a fazer uma análise demográfica aos participantes. Este questionário inclui questões como o género, a idade, as habilitações literárias, o estado profissional,

Foram propostas, a cada utilizador (participante), dois tipos de tarefas:

Tipo 1. O utilizador deve pesquisar por algo explicitamente solicitado, usando os três diferentes modos de pesquisa: Texto (T), Desenho (D), e Upload de Imagem (I). Ou seja, utilizador pesquisaria pelo mesmo conteúdo, mas usando todas as modalidades disponíveis. No fim de completar a pesquisa em cada modo, o utilizador responde ao SUS relativamente a esse modo.

Tipo 2. O utilizador deve pesquisar por algo solicitado, podendo usar uma imagem incompleta existente no dispositivo (L). Para este tipo de tarefa foi também solicitada a partilha da imagem através da funcionalidade parametrizada no protótipo para o efeito.

No tipo de tarefa onde foi solicitado algo explicitamente (Tipo 1), foram feitas pesquisas em diferentes categorias (Objeto, Símbolo/Logo, Conceito), onde foram designadas as seguintes:

- Categoria 1 Símbolo/Logo: Pesquisa por "Estrela"
- Categoria 2: Conceito: Pesquisa por "Nublado"
- Categoria 3: Objeto: Pesquisa por "Carro" (Objeto)

Neste tipo de tarefa (Tipo 1), as categorias foram alternadas entre os utilizadores e a ordem das modalidades também foi alternada de acordo com o método do "quadro latino".

O objetivo destas 3 categorias, foi o de testar vários tipos de conteúdos com diferentes dificuldades e detalhe, nas formas diferentes de expressão.

Na tarefa do Tipo 2, o conteúdo da pesquisa solicitado foi de "Símbolo da paz".

Foram realizados testes a 12 utilizadores de acordo com a seguinte distribuição:

- O primeiro participante (P1) começou pela pesquisa do Tipo 1 "Estrela", seguindo a sequência de modos de pesquisa TDI, seguido da pesquisa do Tipo 2.
- O segundo participante (P2) começou pela pesquisa Tipo 1 "Nublado", seguindo a sequência de modos de pesquisa DIT, seguido da pesquisa do tipo 2.
- O terceiro participante (P3) começou pela pesquisa Tipo 1 "Carro", seguindo a sequência de modos de pesquisa ITD, seguido da pesquisa do tipo 2.

- O quarto participante (P4) começou pela pesquisa Tipo 1 “Nublado”, seguindo a sequência de modos de pesquisa TDI, seguido da pesquisa do tipo 2.
- O quinto participante (P5) começou pela pesquisa Tipo 1 “Estrela”, seguindo a sequência de modos de pesquisa DIT, seguido da pesquisa do tipo 2.
- O sexto participante (P6) começou pela pesquisa Tipo 1 “Carro”, seguindo a sequência de modos de pesquisa ITD, seguido da pesquisa do tipo 2.
- O sétimo participante (P7) começou pela pesquisa Tipo 1 “Estrela”, seguindo a sequência de modos de pesquisa TDI, seguido da pesquisa do tipo 2.
- O oitavo participante (P8) começou pela pesquisa Tipo 1 “Carro”, seguindo a sequência de modos de pesquisa DIT, seguido da pesquisa do tipo 2.
- O nono participante (P9) começou pela pesquisa Tipo 1 “Nublado”, seguindo a sequência de modos de pesquisa ITD, seguido da pesquisa do tipo 2.
- O décimo participante (P10) começou pela pesquisa Tipo 1 “Carro”, seguindo a sequência de modos de pesquisa TDI, seguido da pesquisa do tipo 2.
- O décimo primeiro participante (P11) começou pela pesquisa Tipo 1 “Estrela”, seguindo a sequência de modos de pesquisa DIT, seguido da pesquisa do tipo 2.
- O décimo segundo participante (P12) começou pela pesquisa Tipo 1 “Nublado”, seguindo a sequência de modos de pesquisa ITD, seguido da pesquisa do tipo 2.

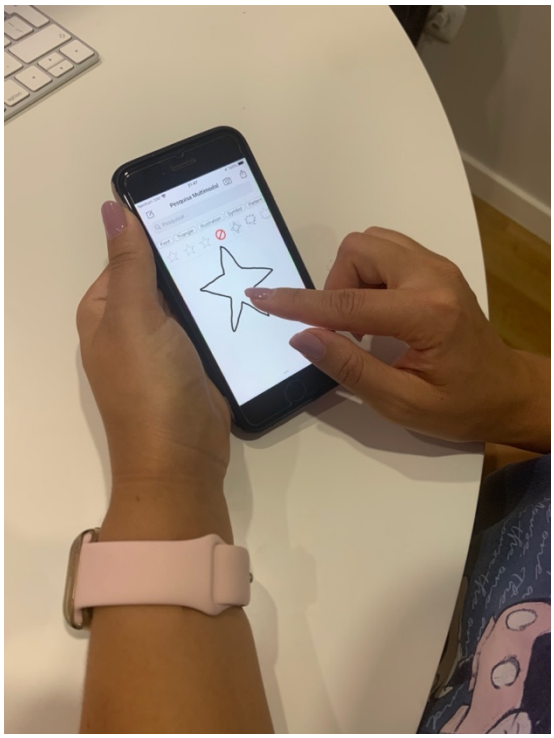

Figura 26 - Testes de Utilizador

Antes do início de cada teste foi feita uma contextualização do protótipo, e foi também dada uma explicação de todo o processo de testagem e dos questionários a serem respondidos.

Os testes foram realizados num ambiente calmo e agradável, onde os participantes dispuseram do tempo que acharam necessário para a conclusão das tarefas. As tarefas eram dadas por terminadas pelo participante quando este se sentia satisfeito, quando este obtivesse o resultado que tinha em mente, ou em caso de abandono da tarefa caso o participante achasse que o esforço não estava a ser proporcional aos resultados obtidos.

Foi solicitado ao utilizador que, ao longo da realização da testagem, fizesse um relato em voz alta da sua experiência.

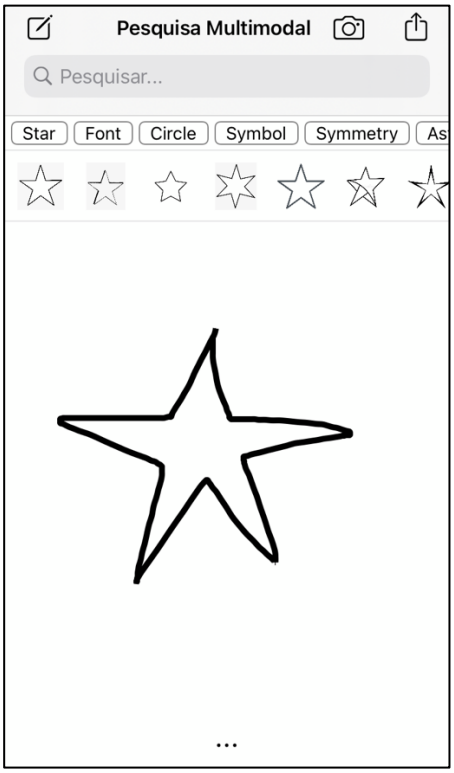

Figura 27 - Printscreen Testes de Utilizador - Modalidade Desenho - Estrela (Símbolo/Logo)

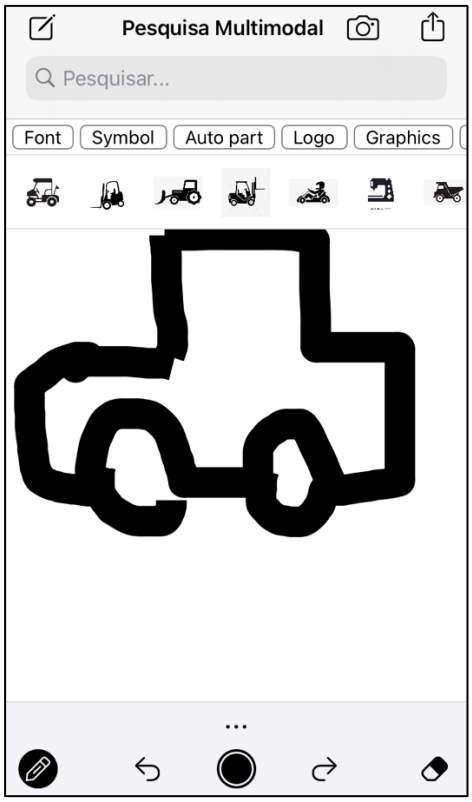

Figura 28 - Printscreen Testes Utilizador - Modalidade Desenho - Carro (Objeto)

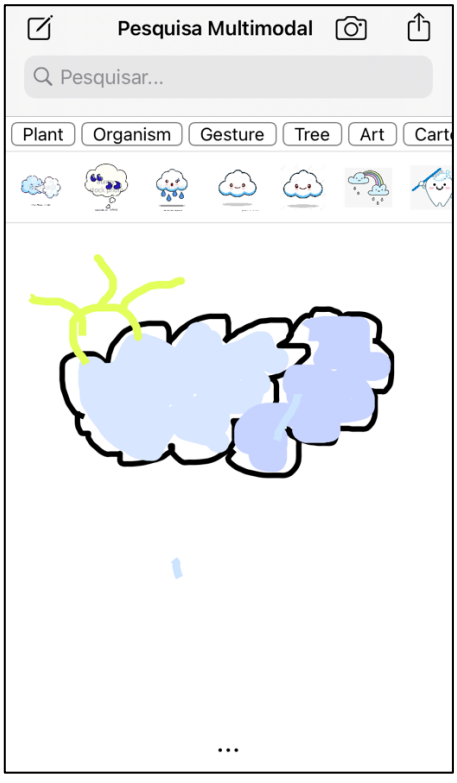

Figura 29 - Printscreen Testes Utilizador - Modalidade Desenho - Nublado (Conceito)

4.2. Resultados

Dos 12 participantes nos testes de avaliação do protótipo, 8 eram do sexo masculino e 4 do sexo feminino. A idade dos participantes variou entre os 29 e os 46 anos. A média de idades é de 36,83 com um desvio padrão (σ) de 6,38. Todos os participantes estavam empregados.

Em relação às habilitações académicas dos participantes, estas estão representadas no seguinte gráfico:

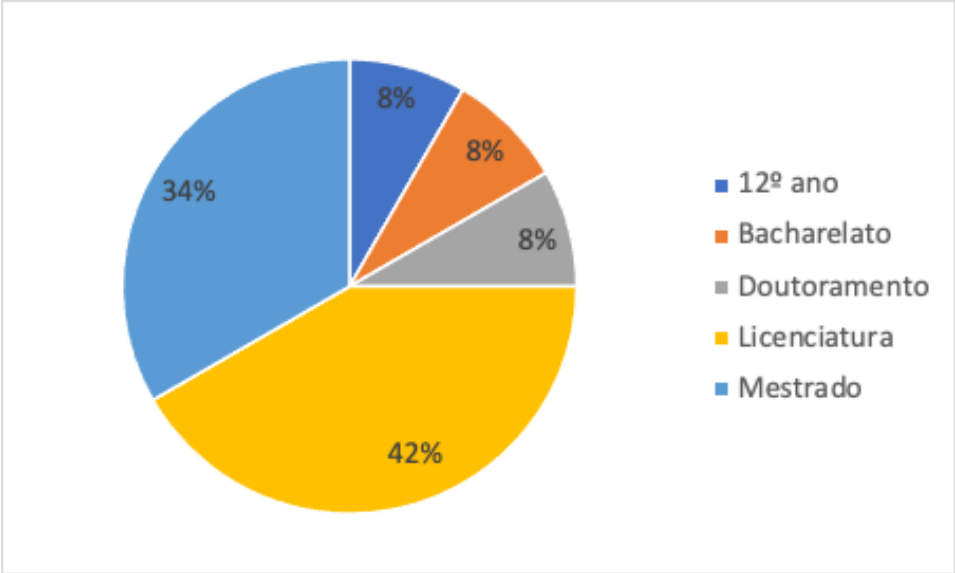

Figura 30 - Habilitações académicas dos participantes

Foram realizadas também algumas questões de autoavaliação aos participantes, no que diz respeito à sua experiência na interação com tecnologia relevante para o estudo.

As questões colocadas foram: "Tenho facilidade no uso/manuseamento de dispositivos móveis", "Tenho muita experiência no uso/manuseamento de aplicações de desenho", "Tenho muita experiência no uso/manuseamento de aplicações de pesquisa de imagens", e "Tenho muita experiência no uso/manuseamento de sistemas de pesquisa inversa de imagens" .

A representação das respostas obtidasé visível através do seguinte gráfico:

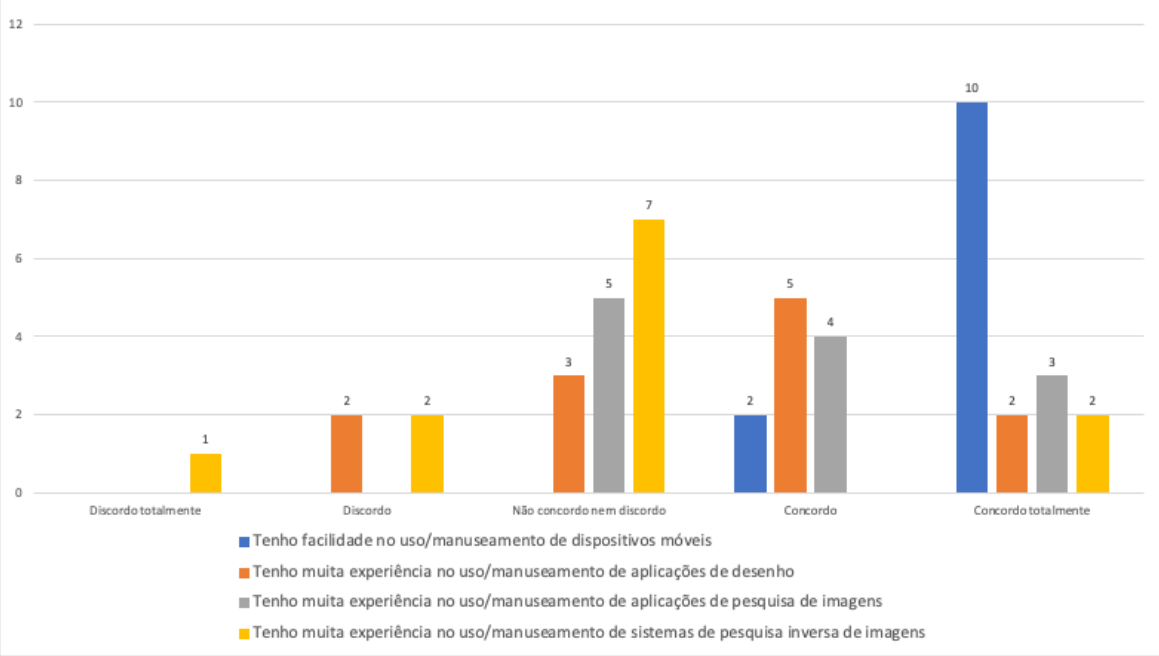

Figura 31 - Gráfico Autoavaliação - Participantes dos testes

É possível aferir então que todos os participantes têm facilidade no uso de dispositivos móveis. Apenas 3 reconheceram ter pouca ou muito pouca experiência no uso de sistemas de pesquisa inversa de imagens, e apenas 2 admitem possuir muita experiência nesta área. Apenas 2 participantes admitiram pouca experiência na manipulação de aplicações de desenho.

No que diz respeito à avaliação do sistema de acordo com a escala SUS, os resultados são calculados mediante a contribuição de cada questão da seguinte forma: às questões 1,3,5,7, e 9 é retirado um valor à pontuação de cada questão, e para as questões 2,4,6,8 e 10 a contribuição é o valor de 5 menos a pontuação de cada uma destas questões (Brooke, 1996).

A seguinte tabela demonstra a avaliação global de acordo com os tipos de pesquisa utilizados:

Tabela 1 - Escala SUS

	SUS Texto	SUS Imagem	SUS Desenho
Média	92,92	86,04	72,71
Desvio Padrão	10,05	14,52	16,56

É possível constatar que de uma forma geral todas as modalidades tiveram uma boa avaliação de usabilidade, sendo a modalidade de texto a melhor avaliada. De salientar que, apesar da modalidade de desenho ser uma novidade para alguns utilizadores, esta teve também uma boa avaliação com um valor acima de 70. Esta modalidade foi também a que apresentou um desvio padrão maior, evidenciando menor homogeneidade, o que poderá indicar que em algumas situações esta modalidade poderá não ter uma boa usabilidade em caso de variação negativa, ou até mesmo uma muito boa usabilidade em caso de variação positiva.

Uma vez que expressar um conceito através de desenho é de um grau de dificuldade maior do que expressar-se através do desenho de um símbolo (Liu et al. 2017), decidiu-se recalcular o SUS sem contemplar a pesquisa por "Nublado" (conceito).

A seguinte tabela expressa esse cálculo:

Tabela 2 - Escala SUS (Sem considerar nas pesquisas o conceito "Nublado")

Sem Pesquisa por Conceito "Nublado"			
	SUS Texto	SUS Imagem	SUS Desenho
Média	90,65	84,68	77,67
Desvio Padrão	9,32	14,29	13,45

É possível então confirmar uma ligeira melhoria nos resultados, o que indicia, para os testes realizados, que a usabilidade do sistema poderá estar relacionada com a dificuldade em se expressar, com a capacidade em desenhar (apenas 2 utilizadores admitiram verbalmente ter alguma habilidade para desenhar), do maior esforço para conseguir alcançar o objetivo (todos os participantes que testaram o sistema usando o conceito "Nublado" precisaram de corrigir ou recomeçar o desenho, e apenas um considerou ter encontrado um resultado satisfatório), ou das limitações de ferramentas do sistema na criação/edição (alguns participantes sugeriram que as ferramentas de alteração de espessura, cor, borracha, anular e refazer, deveriam estar mais visíveis e acessíveis). É possível também observar que a dispersão de resultados é menor, tornando os resultados mais homogéneos.

No que diz respeito ao teste CSI que avalia a criatividade, tal como dito anteriormente, foi aplicado uma vez no fim dos testes de forma a dar uma avaliação global do sistema.

A primeira parte deste teste consiste num grupo de 12 questões, 2 sobre cada fator, determinando a avaliação de cada fator. O quadro seguinte expressa essa avaliação:

Tabela 3 - Escala CSI

Fator	Avaliação
Satisfação	7,86
Pesquisa	8,50
Expressividade	8,07
Imersão	4,00
Compensação pelo Esforço	8,36
Avaliação Global CSI	74,81

A avaliação global CSI tem uma escala de 0 a 100, e os fatores uma escala de 0 a 10.

É possível observar que a avaliação foi boa uma vez que, excetuando o fator Imersão que tem um valor (negativo) abaixo de 5, os restantes têm todos uma classificação acima de 7,4. A Avaliação Global foi de 74,81 sendo também boa.

A segunda parte do teste consiste em 15 questões com 2 hipóteses de resposta, onde cada uma indicia a preferência por um dos fatores. Assim foi calculada uma média da contagem dos fatores escolhidos, e representados na seguinte tabela:

Tabela 4 - Média Contagem Fatores - Escala CSI

Média Contagem "Colaboração"	0,57
Média Contagem "Satisfação"	3,14
Média Contagem "Pesquisa"	2,57
Média Contagem "Expressividade"	3,57
Média Contagem "Imersão "	1,85
Média Contagem "Compensação pelo esforço"	2,71

Segundo este quadro, conseguimos verificar que o fator "Colaboração" foi o menos importante para os utilizadores com um valor 0,57. Os participantes demonstraram através deste teste os fatores mais importares foram Satisfação e a Expressividade.

Foram também colocadas questões de forma a ter uma obter uma avaliação explicita do protótipo. Segue o quadro resumo:

Tabela 5 - Preferências dos participantes

	Escolha participantes		
	Texto	Imagem	Desenho
Qual a modalidade de pesquisa que utilizaria mais?	7	3	2
Qual a modalidade que achou mais fácil de interagir?	10	1	1
Qual a modalidade que apresentou melhores resultados?	10	2	0
Encontrou o que pesquisou?	12	9	8
Achou a funcionalidade de "Live Search" útil?	-	-	11

De acordo com a tabela, constatamos que a modalidade de Texto foi a modalidade mais popular nos participantes, sendo também a mais assertiva no que diz respeito à obtenção de resultados.

5. Discussão

Ao efetuar uma análise a todo o processo e aos resultados obtidos, é evidente que a modalidade de pesquisa preferida pelos participantes é a modalidade de texto, pois foi a que retornou 100% dos resultados esperados pelos utilizadores, sendo a tarefa que requeria menos esforço, e sendo também uma modalidade familiar a todos os participantes.

A análise SUS permitiu concluir que, na modalidade de Desenho, a aplicação tem uma melhor usabilidade no caso de os desenhos serem mais objetivos, concretamente na tarefa para desenhar uma "Estrela" (Símbolo/Logo) ou para desenhar um "Carro" (Objeto). Esta escala SUS confirmou também que a modalidade com maior facilidade de uso é a modalidade de pesquisa por Texto, seguida da modalidade de pesquisa por Imagem e por fim por Desenho. Apesar disto, a grande maioria dos participantes conseguiu ver valor na funcionalidade de "Live Search", pois permitia-lhes terem a noção se estavam a ir de acordo com o objetivo através do feedback no momento. Comparando com a avaliação de Bangor et. al (2009), é possível dizer este protótipo tem a avaliação de "Bom".

Notou-se na maioria dos participantes alguma insegurança aquando da realização do teste na modalidade de Desenho, pois havia sempre a preocupação de criar o melhor input possível. A grande maioria dos utilizadores reconheceu não possuir destreza no desenho. A disponibilização das "labels" e pesquisa pelas mesmas foi uma funcionalidade usada com facilidade pelos utilizadores, isto é, conseguiram fazer "scroll" pelas "labels" e clicar em cima para pesquisar, de forma intuitiva. Isto permitiu também aos utilizadores chegar com menor esforço ao resultado final, pois em alguns casos eram identificadas as "labels" de acordo com o desenho, mas a imagem encontrada não correspondia àquilo que o utilizador esperava. Apenas 1 utilizador não achou a funcionalidade de "Live Search" relevante, e foi o único utilizador dos que fez a pesquisa por desenho da "Estrela", que não a conseguiu encontrar.

Segundo alguns utilizadores a funcionalidade de usar uma imagem semelhante encontrada, permitiu também melhorar o seu desenho, facilitando as pesquisas seguintes.

A análise ao CSI demonstra que o protótipo teve boa avaliação (74,81), o que permite concluir que o mesmo poderá ser uma boa ferramenta de suporte à criatividade. Apesar do valor do Fator de Imersão ser baixo (4,00), após a conclusão dos testes,

alguns utilizadores continuaram a usar o protótipo de forma lúdica, fazendo observações ao sistema como por exemplo: “interessante”, “divertido” ou “engraçado”. Alguns dos participantes dos testes, que trabalham no ramo tecnológico, admitiram nunca ter pensado na funcionalidade de pesquisa por desenho, e com a funcionalidade de retorno imediato dos resultados.

No que diz respeito à utilização das funcionalidades de alteração de cor, espessura, anular/refazer, e borracha, 4 utilizadores não conseguiram encontrar essas funcionalidades, dizendo que o botão estava impercetível e que essas funcionalidades deveriam estar sempre visíveis. Aquando da solicitação do carregamento de uma imagem para a tela 4 utilizadores cometeram erros de interação, confundindo o botão de “Partilha” com o de “Carregamento”. Aquando da reutilização de imagens semelhantes encontradas, os utilizadores assumiam que a imagem poderia ser selecionada através de um toque simples, no entanto essa funcionalidade está parametrizada para um toque longo.

Foi referido apenas por 2 utilizadores a dificuldade em desenhar com o dedo, e foi sugerida, por 4 utilizadores, a utilização do protótipo num tablet.

6. Conclusões

Esta dissertação teve com finalidade estudar as diferentes modalidades de pesquisa por imagens em dispositivos móveis. Para o efeito foi criada uma aplicação desenhada para "smartphone" (iOS) que desse ao utilizador a oportunidade de pesquisar imagens usando as modalidades de Texto, Imagem e Desenho. A implementação desta aplicação teve como base a Framework ATSketchkit, e a utilização das API's Google Cloud Vision e Google Custom Search. Foi implementada uma funcionalidade diferenciadora, relacionada com o desenho, que permitia a obtenção de resultados de pesquisa aquando da interação do utilizador com o sistema.

Foi feita uma revisão de literatura que serviu para um melhor enquadramento no tema e ter conhecimentos básicos sobre as imagens, as suas pesquisas e tecnologias relacionadas.

Após a conclusão da implementação do protótipo, foram realizados testes a 12 utilizadores que avaliaram as diferentes modalidades de pesquisa por imagem e o protótipo em si.

Os resultados demonstram que a modalidade preferida e que obteve melhores resultados foi a modalidade de pesquisa por Texto. Apesar da modalidade de pesquisa por Desenho ter sido a que teve a menor preferência dos utilizadores, estes viram valor na mesma, sendo inclusive uma surpresa positiva para alguns participantes.

6.1. Limitações

As limitações encontradas no desenvolvimento do protótipo estão maioritariamente relacionadas com a Framework ATSketchkit e as API da Google Cloud.

A Framework ATSketchkit disponibiliza ferramentas básicas de criação/edição de imagem, como alterar a cor, alterar espessura da linha, e apagar.

Apenas são permitidas 1000 pesquisas mensais gratuitas durante 1 ano para a pesquisa inversa de imagens através da API Google Cloud Vision e para a pesquisa por palavra chave através da API Google Custom Search são permitidas 100 pesquisas gratuitas por dia durante 3 meses.

Apesar de ser possível desenvolver para iOS de forma gratuita, apenas é possível instalar a aplicação em 1 dispositivo de forma gratuita. Para distribuir a app é necessário se inscrever no "Apple Developer Program" com um custo anual de US$ 99.

6.2. Trabalho futuro

Os testes realizados poderiam ser feitos com utilizadores com uma maior destreza no desenho, o que permitiria aferir melhor o desempenho na modalidade de pesquisa por Desenho. Uma vez que as ferramentas de edição de imagem implementadas são ferramentas simples, estas podem condicionar o desempenho de utilizadores mais capazes nessa área. Então uma alteração a fazer no futuro seria a de melhorar essas ferramentas de desenho.

Outras possíveis adições às funcionalidades seriam, para além de mostrar informações de imagens semelhantes e as "labels", implementar as restantes funcionalidades da API Google Cloud Vision.

De forma a melhorar os resultados obtidos, poderia ser feita a combinação de pesquisas multimodais, por exemplo, conciliar a pesquisa por desenho (ou imagem) juntamente com palavras-chave.

Poderia também ser melhorada a interface do sistema com botões mais explícitos e visíveis.

7. Referências Bibliográficas

Al Falou, Ayman. *Advanced Secure Optical Image Processing for Communications*. IOP Publishing, 2018. https://doi.org/10.1088/978-0-7503-1457-2.

Alkhawlani, Mohammed, e Mohammed Elmogy. «Text-Based, Content-Based, and Semantic-Based Image Retrievals: A Survey» 04, n. 01

Alzu'bi, Ahmad, Abbes Amira, e Naeem Ramzan. «Semantic Content-Based Image Retrieval: A Comprehensive Study». *Journal of Visual Communication and Image Representation* 32 (Outubro de 2015): 20–54. https://doi.org/10.1016/j.jvcir.2015.07.012.

Aslandogan, Y Alp, Chuck Thier, Clement T Yu, Jon Zou, e Naphtali Rishe. «Using Semantic Contents and WordNet™ in Image Retrieval».

Azevedo-Marques, Paulo Mazzoncini de, Marcelo Hossamu Honda, José Antônio H. Rodrigues, Rildo Ribeiro dos Santos, Agma Juci Machado Traina, Caetano Traina Júnior, e Josiane Maria Bueno. «Recuperação de imagem baseada em conteúdo: uso de atributos de textura para caracterização de microcalcificações mamográficas». *Radiologia Brasileira* 35, n. 2 (Março de 2002): 93–98. https://doi.org/10.1590/S0100-39842002000200009.

Bagyammal, T, e Latha Parameswaran. «Context Based Image Retrieval Using Image Features».

Baker, Van, Bern Elliot, Svetlana Sicular, Anthony Mullen, e Erick Brethenoux. «Quadrante Mágico para serviços de desenvolvimento de IA em nuvem», 30.

Banfi, Folco. «Content-Based Image Retrieval Using Hand-Drawn Sketches and Local Features: A Study on Visual Dissimilarity».

Bhunia, Ayan Kumar, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, e Yi-Zhe Song. «Sketch Less for More: On-the-Fly Fine-Grained Sketch-Based Image Retrieval». Em *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9776–85. Seattle, WA, USA: IEEE, 2020. https://doi.org/10.1109/CVPR42600.2020.00980.

C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and effective querying by image content," Journal of intelligent information systems, vol. 3, 1994, pp. 231–262.

Chatfield, Ken, Relja Arandjelović, Omkar Parkhi, e Andrew Zisserman. «On-the-Fly Learning for Visual Search of Large-Scale Image and Video Datasets». *International Journal of Multimedia Information Retrieval* 4, n. 2 (Junho de 2015): 75–93. https://doi.org/10.1007/s13735-015-0077-0.

Chen, Shih-Hsin, e Yi-Hui Chen. «A New Content-Based Image Retrieval Method Based on the Google Cloud Vision API»

Cherry, Erin, e Celine Latulipe. «Quantifying the Creativity Support of Digital Tools through the Creativity Support Index». *ACM Transactions on Computer-Human Interaction* 21, n. 4 (Agosto de 2014): 1–25. https://doi.org/10.1145/2617588.

«Codex: The Journal of the Louisiana Chapter of the ACRL» 3, n. 2 (2015).

Collomosse, John, Tu Bui, e Hailin Jin. «LiveSketch: Query Perturbations for Guided Sketch-Based Visual Search». Em *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2874–82. Long Beach, CA, USA: IEEE, 2019. https://doi.org/10.1109/CVPR.2019.00299.

Long Beach, CA, USA: IEEE, 2019. https://doi.org/10.1109/CVPR.2019.00299.

Datta, Ritendra, Dhiraj Joshi, Jia Li, e James Z. Wang. «Image Retrieval: Ideas, Influences, and Trends of the New Age». *ACM Computing Surveys* 40, n. 2 (Abril de 2008): 1–60. https://doi.org/10.1145/1348246.1348248.

Deshpande, Shantanu, e Naman Goyal. «Sketch Based Image Retrieval», 4.

Devi, Varsha, Junaid Baber, Maheen Bakhtyar, Ihsan Ullah, Waheed Noor, e Abdul Basit. «Performance Evaluation of SIFT and Convolutional Neural Network for Image Retrieval». *International Journal of Advanced Computer Science and Applications* 8, n. 12 (2017). https://doi.org/10.14569/IJACSA.2017.081268.

Eitz, M., K. Hildebrand, T. Boubekeur, e M. Alexa. «Sketch-Based Image Retrieval: Benchmark and Bag-of-Features Descriptors». *IEEE Transactions on Visualization and Computer Graphics* 17, n. 11 (Novembro de 2011): 1624–36. https://doi.org/10.1109/TVCG.2010.266.

Fang Wang, Le Kang, e Yi Li. «Sketch-Based 3D Shape Retrieval Using Convolutional Neural Networks». Em *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1875–83. Boston, MA, USA: IEEE, 2015. https://doi.org/10.1109/CVPR.2015.7298797.

Frazão, Xavier Marques. «Deep Learning Model Combination and Regularization Using Convolutional Neural Networks», 72.

Fuertes, J.M., M. Lucena, N. Pérez de la Blanca, e J. Chamorro-Martínez. «A Scheme of Colour Image Retrieval from Databases». *Pattern Recognition Letters* 22, n. 3–4 (Março de 2001): 323–37. https://doi.org/10.1016/S0167-8655(00)00128-8.

Gong, Jun, e Peter Tarasewich. «GUIDELINES FOR HANDHELD MOBILE DEVICE INTERFACE DESIGN».

H. Tamura and N. Yokoya, “Image database systems: A survey,” Pattern recognition, vol. 17, 1984, pp. 29–43

H.W. Hui, D. Mohamad, and N.A. Ismail, “Approaches, challenges and future direction of image retrieval,” Journal of Computing, vol. 2, 2010, pp. 193–199.

Hochuli, André Gustavo. «Redes Neurais Convolucionais».

Jang, Sungjune, Lawrence H. Kim, Kesler Tanner, Hiroshi Ishii, e Sean Follmer. «Haptic Edge Display for Mobile Tactile Interaction». Em *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 3706–16. San Jose California USA: ACM, 2016. https://doi.org/10.1145/2858036.2858264.

Júnior, Elias Borges Macena. «Aplicação de Técnicas de Content-Based Image Retrieval (CBIR) em Imagens Radiográficas», 2016, 72.

Khan, Asifullah, Anabia Sohail, Umme Zahoora, e Aqsa Saeed Qureshi. «A Survey of the Recent Architectures of Deep Convolutional Neural Networks». *Artificial Intelligence Review* 53, n. 8 (Dezembro de 2020): 5455–5516. https://doi.org/10.1007/s10462-020-09825-6.

Koller, Oscar, Hermann Ney, e Richard Bowden. «Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data Is Continuous and Weakly Labelled». Em *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3793–3802. Las Vegas, NV, USA: IEEE, 2016. https://doi.org/10.1109/CVPR.2016.412.

Liu, Li, Fumin Shen, Yuming Shen, Xianglong Liu, e Ling Shao. «Deep Sketch Hashing: Fast Free-Hand Sketch-Based Image Retrieval». Em *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2298–2307. Honolulu, HI: IEEE, 2017. https://doi.org/10.1109/CVPR.2017.247.

Lu, Eric Hsueh-Chan, e Jing-Mei Ciou. «Integration of Convolutional Neural Network and Error Correction for Indoor Positioning». *ISPRS International Journal of Geo-Information* 9, n. 2 (29 de Janeiro de 2020): 74. https://doi.org/10.3390/ijgi9020074.

. «Integration of Convolutional Neural Network and Error Correction for Indoor Positioning». *ISPRS International Journal of Geo-Information* 9, n. 2 (29 de Janeiro de 2020): 74. https://doi.org/10.3390/ijgi9020074.

Magalhães, Ícaro Lima. «Um estudo comparativo entre padrões arquiteturais para o desenvolvimento de aplicativos para a plataforma iOS», 51.

Malik, Fazal, e Baharum Baharudin. «Analysis of Distance Metrics in Content-Based Image Retrieval Using Statistical Quantized Histogram Texture Features in the DCT Domain». *Journal of King Saud University - Computer and Information Sciences* 25, n. 2 (Julho de 2013): 207–18. https://doi.org/10.1016/j.jksuci.2012.11.004.

«Manuscript-2ndrevision1.docx», sem data.

Martinez, José, e Sylvie Guillaume. «Colour Image Retrieval Fitted to “classical” Querying». Em *Image Analysis and Processing*, editado por Alberto Del Bimbo, 1311:14–21. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. https://doi.org/10.1007/3-540-63508-4_100.

Martins, Marcelo Ribeiro. «ANÁLISE AUTOMATIZADA DE DOCUMENTOS COM O GOOGLE CLOUD COGNITIVE SERVICES», 33.

Meharban, M.S., e Dr.S. Priya. «A Review on Image Retrieval Techniques». *Bonfring International Journal of Advances in Image Processing* 6, n. 2 (30 de Abril de 2016): 07–10. https://doi.org/10.9756/BIJAIP.8136.

Pang, Kaiyue, Ke Li, Yongxin Yang, Honggang Zhang, Timothy M. Hospedales, Tao Xiang, e Yi-Zhe Song. «Generalising Fine-Grained Sketch-Based Image Retrieval». Em *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 677–86. Long Beach, CA, USA: IEEE, 2019. https://doi.org/10.1109/CVPR.2019.00077.

Parui, Sarthak, e Anurag Mittal. «Similarity-Invariant Sketch-Based Image Retrieval in Large Databases». Em *Computer Vision – ECCV 2014*, editado por David Fleet, Tomas Pajdla, Bernt Schiele, e Tinne Tuytelaars, 8694:398–414. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014. https://doi.org/10.1007/978-3-319-10599-4_26.

Portenier, Tiziano, Qiyang Hu, Paolo Favaro, e Matthias Zwicker. «SmartSketcher: Sketch-Based Image Retrieval with Dynamic Semantic Re-Ranking». Em *Proceedings of the Symposium on Sketch-Based Interfaces and Modeling*, 1–12. Los Angeles California: ACM, 2017. https://doi.org/10.1145/3092907.3092910.

Radenović, Filip, Giorgos Tolias, e Ondřej Chum. «Deep Shape Matching». Em *Computer Vision – ECCV 2018*, editado por Vittorio Ferrari, Martial Hebert, Cristian

Sminchisescu, e Yair Weiss, 11209:774–91. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018. https://doi.org/10.1007/978-3-030-01228-1_46.

Sain, Aneeshan. «Cross-Modal Hierarchical Modelling for Fine-Grained Sketch Based Image Retrieval», 14.

Sangkloy, Patsorn, Nathan Burnell, Cusuh Ham, e James Hays. «The Sketchy Database: Learning to Retrieve Badly Drawn Bunnies». *ACM Transactions on Graphics* 35, n. 4 (11 de Julho de 2016): 1–12. https://doi.org/10.1145/2897824.2925954.

Shao, Hong, Yueshu Wu, Wencheng Cui, e Jinxia Zhang. «Image Retrieval Based on MPEG-7 Dominant Color Descriptor». Em *2008 The 9th International Conference for Young Computer Scientists*, 753–57. Hunan, China: IEEE, 2008. https://doi.org/10.1109/ICYCS.2008.89.

Shen, Yuming, Li Liu, Fumin Shen, e Ling Shao. «Zero-Shot Sketch-Image Hashing». Em *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3598–3607. Salt Lake City, UT: IEEE, 2018. https://doi.org/10.1109/CVPR.2018.00379.

Silva, Ronildo Oliveira Da. «Análise de Desempenho da Google Cloud Vision API em Leitura de Textos Provenientes de Imagens Naturais», 2019, 54.

Singh, Nidhi, Kanchan Singh, e Ashok K. Sinha. «A Novel Approach for Content Based Image Retrieval». *Procedia Technology* 4 (2012): 245–50. https://doi.org/10.1016/j.protcy.2012.05.037.

Song, Jifei, Yi-Zhe Song, Tao Xiang, Timothy Hospedales, e Xiang Ruan. «Deep Multi-Task Attribute-Driven Ranking for Fine-Grained Sketch-Based Image Retrieval». Em *Procedings of the British Machine Vision Conference 2016*, 132.1-132.11. York, UK: British Machine Vision Association, 2016. https://doi.org/10.5244/C.30.132.

Thomee, Bart, e Michael S. Lew. «Interactive Search in Image Retrieval: A Survey». *International Journal of Multimedia Information Retrieval* 1, n. 2 (Julho de 2012): 71–86. https://doi.org/10.1007/s13735-012-0014-4.

Thompson, Santi, e Michele Reilly. «"A Picture Is Worth a Thousand Words": Reverse Image Lookup and Digital Library Assessment». *Journal of the Association for Information Science and Technology* 68, n. 9 (Setembro de 2017): 2264–66. https://doi.org/10.1002/asi.23847.

Tripathy, Sanjaya Shankar, Ravi Shekhar, e R. Sukesh Kumar. «Texture Retrieval System Using Intuitionistic Fuzzy Set Theory». Em *2011 International Conference on Devices and Communications (ICDeCom)*, 1–5. Mesra, Ranchi, India: IEEE, 2011. https://doi.org/10.1109/ICDECOM.2011.5738490.

Verma, B., P. Sharma, S. Kulkarni, e H. Selvaraj. «An Intelligent On-Line System for Content Based Image Retrieval». Em *Proceedings Third International Conference on Computational Intelligence and Multimedia Applications. ICCIMA'99 (Cat. No.PR00300)*, 273–77. New Delhi, India: IEEE Comput. Soc, 1999. https://doi.org/10.1109/ICCIMA.1999.798542.

«Visual Analytics for Semantic Based Image Retrieval (Sbir): Semantic Tool». *INTERNATIONAL JOURNAL OF LATEST TRENDS IN ENGINEERING AND TECHNOLOGY* 7, n. 2 (2016). https://doi.org/10.21172/1.72.548.

Westerveld, Thijs. «Image Retrieval: Content versus Context», 9.

Xu, Peng, Qiyue Yin, Yonggang Qi, Yi-Zhe Song, Zhanyu Ma, Liang Wang, e Jun Guo. «Instance-Level Coupled Subspace Learning for Fine-Grained Sketch-Based Image Retrieval». Em *Computer Vision – ECCV 2016 Workshops*, editado por Gang Hua e Hervé Jégou, 9913:19–34. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016. https://doi.org/10.1007/978-3-319-46604-0_2.

Yan, Ke, Yaowei Wang, Dawei Liang, Tiejun Huang, e Yonghong Tian. «CNN vs. SIFT for Image Retrieval: Alternative or Complementary?» Em *Proceedings of the 24th ACM International Conference on Multimedia*, 407–11. Amsterdam The Netherlands: ACM, 2016. https://doi.org/10.1145/2964284.2967252.

Yasmin, M., M. Sharif, I. Irum, e S. Mohsin. «An Efficient Content Based Image Retrieval Using EI Classification and Color Features». *Journal of Applied Research and Technology* 12, n. 5 (Outubro de 2014): 877–85. https://doi.org/10.1016/S1665-6423(14)70594-2.

Yasmin, Mussarat, Sajjad Mohsin, e Muhammad Sharif. «Intelligent Image Retrieval Techniques: A Survey». *Journal of Applied Research and Technology* 12, n. 1 (Fevereiro de 2014): 87–103. https://doi.org/10.1016/S1665-6423(14)71609-8.

Yu, Qian, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, e Chen-Change Loy. «Sketch Me That Shoe», 9.

Y. Rui, T.S. Huang, and S.-F. Chang, "Image retrieval: Current techniques, promising directions, and open issues," Journal of visual communication and image representation, vol. 10, 1999, pp. 39–62.

Zhang, Xianlin, Xueming Li, Yang Liu, e Fangxiang Feng. «A Survey on Freehand Sketch Recognition and Retrieval». *Image and Vision Computing* 89 (Setembro de 2019): 67–87. https://doi.org/10.1016/j.imavis.2019.06.010.

Zhang, Zhaolong, Yuejie Zhang, Rui Feng, Tao Zhang, e Weiguo Fan. «Zero-Shot Sketch-Based Image Retrieval via Graph Convolution Network». *Proceedings of the AAAI Conference on Artificial Intelligence* 34, n. 07 (3 de Abril de 2020): 12943–50. https://doi.org/10.1609/aaai.v34i07.6993.

https://cloud.google.com/vision/docs/before-you-begin
https://cloud.google.com/vision/docs/setup
https://docs.oracle.com/cd/B19306_01/appdev.102/b14302/ch_cbr.htm
https://www.bankmycell.com/blog/how-many-phones-are-in-the-world
https://www.syte.ai/blog/brief-history-image-search/
https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/
https://gs.statcounter.com/os-market-share/mobile/worldwide
http://www.go2web.com.br/pt-BR/blog/o-impacto-da-internet-sobre-a-sociedade-uma-perspectiva-global.html
https://www.arrow.com/en/research-and-events/articles/neural-networks-and-computer-vision
https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2
https://www.g2.com/products/google-cloud-vision-api/competitors/alternatives

Anexos

Exemplo: JSON para uma query de imagem obtendo informação das “labels”


```
"responses" : [
    {
      "labelAnnotations" : [
        {
          "topicality" : 0.97086762999999998,
          "mid" : "\/m\/0838f",
          "description" : "Water",
          "score" : 0.97086762999999998
        },
        {
          "topicality" : 0.96246240000000005,
          "mid" : "\/m\/01bqvp",
          "description" : "Sky",
          "score" : 0.96246240000000005
        },
        {
          "topicality" : 0.9541309,
          "mid" : "\/m\/05s2s",
          "description" : "Plant",
          "score" : 0.9541309
        },
        {
          "topicality" : 0.94861510000000004,
          "mid" : "\/m\/015s2f",
          "description" : "Water resources",
          "score" : 0.94861510000000004
        },
        {
          "topicality" : 0.89993780000000001,
          "mid" : "\/m\/05h0n",
          "description" : "Nature",
          "score" : 0.89993780000000001
        },
        {
          "topicality" : 0.88925469999999995,
          "mid" : "\/m\/03d28y3",
          "description" : "Natural landscape",
          "score" : 0.88925469999999995
        },
        {
          "topicality" : 0.88041747000000004,
          "mid" : "\/m\/0j2kx",
          "description" : "Waterfall",
          "score" : 0.88041747000000004
        },
        {
          "topicality" : 0.86375933999999999,
          "mid" : "\/m\/03cjrt",
          "description" : "Highland",
          "score" : 0.86375933999999999
        },
        {
          "topicality" : 0.85246420000000001,
          "mid" : "\/m\/0csby",
          "description" : "Cloud",
          "score" : 0.85246420000000001
        },
        {
          "topicality" : 0.84970460000000003,
          "mid" : "\/m\/01fnns",
          "description" : "Vegetation",
          "score" : 0.84970460000000003
        }
      ]
    }
```

```
    ]
})
```