

RICARDO JORGE MARTINS DA VEIGA

**POSE ESTIMATION SYSTEM BASED ON MONOCULAR
CAMERAS**



UNIVERSIDADE DO ALGARVE
Instituto Superior de Engenharia
2019

RICARDO JORGE MARTINS DA VEIGA

POSE ESTIMATION SYSTEM BASED ON MONOCULAR CAMERAS

**Master Thesis in Electric and Electronic Engineering
Specialty in Information Technologies and Telecommunications**

**Work done under the supervision of:
Professor Doutor João Miguel Fernandes Rodrigues
Professor Doutor Pedro Jorge Sequeira Cardoso**



**UNIVERSIDADE DO ALGARVE
Instituto Superior de Engenharia**

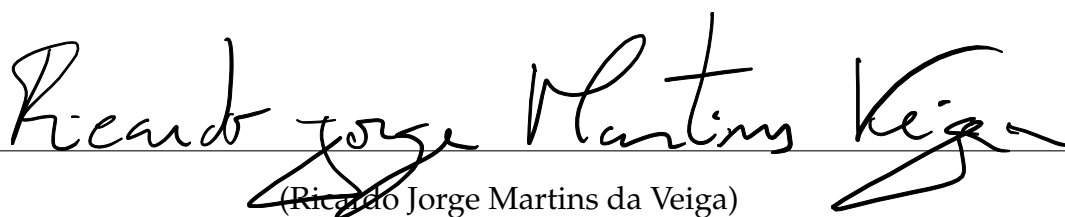
2019

POSE ESTIMATION SYSTEM BASED ON MONOCULAR CAMERAS

Declaração de autoria de trabalho

Declaro ser o autor deste trabalho, que é original e inédito. Autores e trabalhos consultados estão devidamente citados no texto e constam da listagem de referências incluída.

I hereby declare to be the author of this work, which is original and unpublished. Authors and works consulted are properly cited in the text and included in the reference list.



(Ricardo Jorge Martins da Veiga)

©2019, RICARDO JORGE MARTINS DA VEIGA

A Universidade do Algarve reserva para si o direito, em conformidade com o disposto no Código do Direito de Autor e dos Direitos Conexos, de arquivar, reproduzir e publicar a obra, independentemente do meio utilizado, bem como de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição para fins meramente educacionais ou de investigação e não comerciais, conquanto seja dado o devido crédito ao autor e editor respetivos.

The University of the Algarve reserves the right, in accordance with the terms of the Copyright and Related Rights Code, to file, reproduce and publish the work, regardless of the methods used, as well as to publish it through scientific repositories and to allow it to be copied and distributed for purely educational or research purposes and never for commercial purposes, provided that due credit is given to the respective author and publisher.

Abstract

Our world is full of wonders. It is filled with mysteries and challenges, which through the ages inspired and called for the human civilization to grow itself, either philosophically or sociologically. In time, humans reached their own physical limitations; nevertheless, we created technology to help us overcome it. Like the ancient uncovered land, we are pulled into the discovery and innovation of our time. All of this is possible due to a very human characteristic - our imagination.

The world that surrounds us is mostly already discovered, but with the power of computer vision (CV) and augmented reality (AR), we are able to live in multiple hidden universes alongside our own. With the increasing performance and capabilities of the current mobile devices, AR is what we dream it can be. There are still many obstacles, but this future is already our reality, and with the evolving technologies closing the gap between the real and the virtual world, soon it will be possible for us to surround ourselves into other dimensions, or fuse them with our own.

This thesis focuses on the development of a system to predict the camera's pose estimation in the real-world regarding to the virtual world axis. The work was developed as a sub-module integrated on the M5SAR project: Mobile Five Senses Augmented Reality System for Museums, aiming to a more immerse experience with the total or partial replacement of the environments' surroundings. It is based mainly on man-made buildings indoors and their typical rectangular cuboid shape. With the possibility of knowing the user's camera direction, we can then superimpose dynamic AR

content, inviting the user to explore the hidden worlds.

The M5SAR project introduced a new way to explore the existent historical museums by exploring the human's five senses: hearing, smell, taste, touch, vision. With this innovative technology, the user is able to enhance their visitation and immerse themselves into a virtual world blended with our reality. A mobile device application was built containing an innovating framework: MIRAR - Mobile Image Recognition based Augmented Reality - containing object recognition, navigation, and additional AR information projection in order to enrich the users' visit, providing an intuitive and compelling information regarding the available artworks, exploring the hearing and vision senses. A device specially designed was built to explore the additional three senses: smell, taste and touch which, when attached to a mobile device, either smartphone or tablet, would pair with it and automatically react in with the offered narrative related to the artwork, immersing the user with a sensorial experience.

As mentioned above, the work presented on this thesis is relative to a sub-module of the MIRAR regarding environment detection and the superimposition of AR content. With the main goal being the full replacement of the walls' contents, and with the possibility of keeping the artwork visible or not, it presented an additional challenge with the limitation of using only monocular cameras. Without the depth information, any 2D image of an environment, to a computer doesn't represent the tridimensional layout of the real-world dimensions. Nevertheless, man-based building tends to follow a rectangular approach to divisions' constructions, which allows for a prediction to where the vanishing point on any environment image may point, allowing the reconstruction of an environment's layout from a 2D image. Furthermore, combining this information with an initial localization through an improved image recognition to retrieve the camera's spatial position regarding to the real-world coordinates and the virtual-world, alas, pose estimation, allowed for the possibility of superimposing specific localized AR content over the user's mobile device frame, in order to immerse, i.e., a museum's visitor into another era correlated to the present artworks' historical period. Through the work developed for this thesis, it was also presented a better

planar surface in space rectification and retrieval, a hybrid and scalable multiple images matching system, a more stabilized outlier filtration applied to the camera's axis, and a continuous tracking system that works with uncalibrated cameras and is able to achieve particularly obtuse angles and still maintain the surface superimposition.

Furthermore, a novelty method using deep learning models for semantic segmentation was introduced for indoor layout estimation based on monocular images. Contrary to the previous developed methods, there is no need to perform geometric calculations to achieve a near state of the art performance with a fraction of the parameters required by similar methods. Contrary to the previous work presented on this thesis, this method performs well even in unseen and cluttered rooms if they follow the Manhattan assumption. An additional lightweight application to retrieve the camera pose estimation is presented using the proposed method.

Keywords: Artificial Neural Networks, Augmented Reality, Computer Vision, Deep Learning, Human-Computer Interaction, Indoor Layout Estimation, Machine Learning, Markerless-based Recognition, Walls Recognition, Superposition

Resumo

O nosso mundo está repleto de maravilhas. Está cheio de mistérios e desafios, os quais, ao longo das eras, inspiraram e impulsionaram a civilização humana a evoluir, seja filosófica ou sociologicamente. Eventualmente, os humanos foram confrontados com os seus limites físicos; desta forma, criaram tecnologias que permitiram superá-los. Assim como as terras antigas por descobrir, somos impulsionados à descoberta e inovação da nossa era, e tudo isso é possível graças a uma característica marcadamente humana: a nossa imaginação.

O mundo que nos rodeia está praticamente todo descoberto, mas com o poder da visão computacional (VC) e da realidade aumentada (RA), podemos viver em múltiplos universos ocultos dentro do nosso. Com o aumento da performance e das capacidades dos dispositivos móveis da atualidade, a RA pode ser exatamente aquilo que sonhamos. Continuam a existir muitos obstáculos, mas este futuro já é o nosso presente, e com a evolução das tecnologias a fechar o fosso entre o mundo real e o mundo virtual, em breve será possível cercarmos-nos de outras dimensões, ou fundi-las dentro da nossa.

Esta tese foca-se no desenvolvimento de um sistema de predição para a estimação da pose da câmara no mundo real em relação ao eixo virtual do mundo. Este trabalho foi desenvolvido como um sub-módulo integrado no projeto M5SAR: Mobile Five Senses Augmented Reality System for Museums, com o objetivo de alcançar uma experiência mais imersiva com a substituição total ou parcial dos limites do ambi-

ente. Dedicar-se ao interior de edifícios de arquitetura humana e a sua típica forma de retângulo cuboide. Com a possibilidade de saber a direção da câmara do dispositivo, podemos então sobrepor conteúdo dinâmico de RA, num convite ao utilizador para explorar os mundos ocultos.

O projeto M5SAR introduziu uma nova forma de explorar os museus históricos existentes através da exploração dos cinco sentidos humanos: a audição, o cheiro, o paladar, o toque e a visão. Com essa tecnologia inovadora, o utilizador pode engrandecer a sua visita e mergulhar num mundo virtual mesclado com a nossa realidade. Uma aplicação para dispositivo móvel foi criada, contendo uma estrutura inovadora: MIRAR - Mobile Image Recognition based Augmented Reality - a possuir o reconhecimento de objetos, navegação e projeção de informação de RA adicional, de forma a enriquecer a visita do utilizador, a fornecer informação intuitiva e interessante em relação às obras de arte disponíveis, a explorar os sentidos da audição e da visão. Foi também desenhado um dispositivo para exploração em particular dos três outros sentidos adicionais: o cheiro, o toque e o sabor. Este dispositivo, quando afixado a um dispositivo móvel, como um smartphone ou tablet, emparelha e reage com este automaticamente com a narrativa relacionada à obra de arte, a imergir o utilizador numa experiência sensorial.

Como já referido, o trabalho apresentado nesta tese é relativo a um sub-módulo do MIRAR, relativamente à deteção do ambiente e a sobreposição de conteúdo de RA. Sendo o objetivo principal a substituição completa dos conteúdos das paredes, e com a possibilidade de manter as obras de arte visíveis ou não, foi apresentado um desafio adicional com a limitação do uso de apenas câmaras monoculares. Sem a informação relativa à profundidade, qualquer imagem bidimensional de um ambiente, para um computador isso não se traduz na dimensão tridimensional das dimensões do mundo real. No entanto, as construções de origem humana tendem a seguir uma abordagem retangular às divisões dos edifícios, o que permite uma predição de onde poderá apontar o ponto de fuga de qualquer ambiente, a permitir a reconstrução da disposição de uma divisão através de uma imagem bidimensional. Adicionalmente, ao combinar

esta informação com uma localização inicial através de um reconhecimento por imagem refinado, para obter a posição espacial da câmara em relação às coordenadas do mundo real e do mundo virtual, ou seja, uma estimativa da pose, foi possível alcançar a possibilidade de sobrepor conteúdo de RA especificamente localizado sobre a moldura do dispositivo móvel, de maneira a imergir, ou seja, colocar o visitante do museu dentro de outra era, relativa ao período histórico da obra de arte em questão. Ao longo do trabalho desenvolvido para esta tese, também foi apresentada uma melhor superfície planar na recolha e retificação espacial, um sistema de comparação de múltiplas imagens híbrido e escalável, um filtro de *outliers* mais estabilizado, aplicado ao eixo da câmara, e um sistema de *tracking* contínuo que funciona com câmaras não calibradas e que consegue obter ângulos particularmente obtusos, continuando a manter a sobreposição da superfície.

Adicionalmente, um algoritmo inovador baseado num modelo de *deep learning* para a segmentação semântica foi introduzido na estimativa do traçado com base em imagens monoculares. Ao contrário de métodos previamente desenvolvidos, não é necessário realizar cálculos geométricos para obter um desempenho próximo ao *state of the art* e ao mesmo tempo usar uma fração dos parâmetros requeridos para métodos semelhantes. Inversamente ao trabalho previamente apresentado nesta tese, este método apresenta um bom desempenho mesmo em divisões sem vista ou obstruídas, caso sigam a mesma premissa Manhattan. Uma leve aplicação adicional para obter a posição da câmara é apresentada usando o método proposto.

Palavras chave: Interação Homem-Máquina, Aprendizagem Automática, Realidade Aumentada, Reconhecimento baseado em Padrões, Reconhecimento de Paredes, Redes Neurais Artificiais, Sobreposição, Visão Computacional

Acknowledgements

This thesis is the result of endless nights and days finding my passion for the unknown, for the challenging, and for innovation. I would first like to thank my advisor, Professor Doutor *João Rodrigues* for never stopping to believe in me, always pushing me a little bit further when I needed, even when I thought otherwise, and steering me into the right direction whenever I got lost, which was plenty. This and future work wouldn't exist without his support and guidance. I would also like to thank Professor Doutor *Pedro Cardoso* for always being there when I needed him, specially when I would get lost in ideas and he would bring me back to reality. His advices were invaluable to allow me to pursue a fulfilling career, or so I hope. A special thanks to my workplace colleagues, who made me laugh, made me grow, and also made me insane, for giving me some of my best memories through this journey. Without their continuous support and horsing around, I would never be able to sustain the mental craziness necessary to finish this thesis. A very fond thanks to my family, to my little ones, who sprouted into my life bringing, amidst the chaos, a sense of purpose and belonging. You will always be the shining lights of my heart, and the endless hope of my soul. Finally, I must express the deepest gratitude to my wife, who was always there for me, providing me with endless support and faith in my work, keeping me sane and focused through out my academic journey, and pulling me in whenever I got lost. This work was only possible due to your never ending love and encouragement.

Thank you.

Table of Contents

List of Tables	xix
List of Figures	xx
List of Abbreviations	xxiii
Chapter 1 Introduction	1
1.1 Scope of the Thesis	4
1.2 Objectives	5
1.3 Structure of the Thesis	6
1.4 Overview of the Thesis	7
Chapter 2 Mobile Augmented Reality Framework - MIRAR	9
2.1 Introduction	10
2.2 MIRAR framework	13
2.3 Object detection, recognition and tracking module	15
2.4 Environments detection	17
2.4.1 Tests	21
2.5 Human shape detection	27
2.5.1 Tests	29
2.6 Conclusions	30
Chapter 3 Augmented Reality Indoor Environment	
Detection: Proof-of-Concept	33
3.1 Introduction	34
3.2 Environment Detection	36
3.3 Conclusions	44
Chapter 4 AR Contents Superimposition on Walls and Persons	47
4.1 Introduction	48
4.2 Contextualization and State of the Art	49
4.3 Wall Detection and Information Overlapping	52
4.4 Person Detection and Clothes Overlapping	59
4.5 Conclusions	63
Chapter 5 Efficient Small-Scale Network for Room Spatial Layout Estimation	67
5.1 Introduction	68
5.2 Related Work	70
5.3 Method	73
5.3.1 Overview	73

5.3.2	Network Architecture	74
5.3.3	Layout Refinement	75
5.4	Experimental Results	76
5.4.1	Datasets	76
5.4.2	Accuracy	77
5.4.3	Experimental Results	78
5.5	Applications	80
5.5.1	Camera Pose Estimation	80
5.6	Conclusions and Future Work	81
Chapter 6	Conclusions	83
6.1	Future Work	85
6.2	Publications	86

List of Tables

3.1	Comparison of the performance between FLANN Based Matched and FLANN Index	38
5.1	Room layout estimation performance on Hedau dataset	77
5.2	Room layout estimation performance on LSUN dataset	78

List of Figures

2.1	M5SAR’s architecture	13
2.2	Examples of detected and tracked markers with the corresponding axis .	16
2.3	Extracted keypoints location for each template	18
2.4	Example of a sequence of frames with matched templates	21
2.5	Illustration of the number of matched frames	23
2.6	Initial results of matching while the view is obstructed by persons	26
2.7	Example of human detection and segmentation	28
3.2	Example of different amount of keypoints in a frame	40
3.3	Example of the lines found in the environment through different per- spectives.	42
3.4	Superimposition results	43
4.1	Example of pre-processing of templates	54
4.2	Example of some of the templates used during the bundle creation stage.	55
4.3	Pipeline of the environments’ superimposition algorithm	56
4.4	Example of confusion between left and right ankle	60
4.5	Pose estimation stabilization groups.	61
4.6	Examples of volume 2D views.	61
4.7	Created views condition	62
4.8	Left, volume keypoints. Right, example of a limb’s angle.	63
4.9	Volume keypoints	63
4.10	Examples of both modules working together.	65
5.1	LSUN dataset example	69
5.2	DeepLabV3 Encoder-Decoder architecture	72
5.3	Different type of room layouts available on LSUN	75
5.4	Distribution of samples per type of the LSUN dataset	76
5.5	Examples of room layout estimated by our method	79
5.6	Application: Relative Camera Pose Estimation	80

List of Abbreviations

6DoF	Six Degrees of Freedom
AKAZE	Accelerated-KAZE
ANN	Artificial Neural Networks
APP	Application
AR	Augmented Reality
ASPP	Atrous Spatial Pyramid Pooling
AUI	Adaptative User Interfaces
BF	Brute-Force
BRIEF	Binary Robust Independent Elementary Features
BRISK	Binary Robust Invariant Scalable Keypoints
CV	Computer Vision
CNN	Convolutional Neural Network
CRF	Conditional Random Field
CPU	Central processing unit
DB	Database
DCNN	Deep Convolutional Neural Network
FBM	FLANN Based Matcher
FCN	Fully Convolutional Neural Network
FI	FLANN Index
FIFO	First In, First Out

FLANN	Fast Library for Approximate Nearest Neighbours
KNN	K-Nearest Neighbors
LIDAR	Light Detection And Ranging
LSH	Locality-Sensitive Hashing
LSTM	Long Short-Term Memory
M5SAR	Mobile Five Senses Augmented Reality System for Museums
MIRAR	Mobile Image Recognition based Augmented Reality
ORB	Oriented FAST and Rotated BRIEF
PDTTSS	Portable Device for Touch, Taste and Smell Sensations
PEPE	Portable Environment Pose Estimation
RANSAC	Random Sample Consensus
RGB	Red Green Blue
RGBD	Red Green Blue Depth
RoI	Region of Interest
RNN	Recurrent Neural Network
SDK	Software Development Kit
SLAM	Simultaneous Localization And Mapping
SSD	Single-Shot Detection
SVM	Support Vectoring Machine

1

Introduction

Augmented Reality (AR) is one of the most emerging and promising technologies in the current landscape, being already defined as part of our future and not simply a gimmick. With the power of AR we are able to enhance our day-to-day lives, either by displaying additional always-updated informations using only the mobile devices, without any need for hardware research, development or maintenance while also removing the additional cost on scalability; or by interacting with the user using projection devices associated with cameras, allowing for unique experiences in diverse areas, such as marketing, education, industry; or by presenting the user with the gift of exploring multiple different worlds hidden in plain sight.

Being a specific area of Computer Vision (CV), AR shares some of the same obstacles that continue to challenge researchers. When combined with the use on mo-

mobile devices, these obstacles tend to increase in complexity, either due to the existence of different uncalibrated cameras, or the unpredicted movement from the mobile devices' users, or even the disparity in performance between multiple devices. One of these challenges surrounds the accurate prediction of the camera's position between the real-world and the virtual world. As most of the nowadays AR applications are developed to work over a simple 2D frame obtained from the mobile devices' camera, regardless of the increasing availability of mobile devices equipped with depth sensors, due to the fact that these kinds of solutions are aimed to perform in specific conditions, such as the unlocking of a mobile device using facial tridimensional landmarks, and are not yet globally standardized or available for backwards cameras. Nevertheless, the majority of the practical solutions developed over the years have been based on structure from motion techniques, and 3D cloud of points, which are demanding algorithms in terms of performance. Presently, the most used technique relies on a fusion between the gyroscope values retrieved in-between frames and the calculation of the pose estimation through flow analysis of the features.

When we achieve a stable and trustful calculation of the real-world camera's pose estimation, we open the virtual world to the user, allowing for endless applications, from indoor navigation, to floating AR content, or even the complete overhaul of any available division. Although there are already some alternatives to peek into another "dimensions", none of them evaluate the users' surroundings, which forces the user to be aware of their real-world while they are exploring, which breaks the immersive experience. Some of the available alternatives require extremely visually rich environment for their planar detection and continuous tracking.

In this thesis, the method developed to continuously calculate and predict the camera's pose estimation focused on the different geometric characteristics of our surroundings, allowing for the possibility of superimposing AR content on any vertical plane - walls. Beginning with an initialization of the users' localization on any previous scanned room, this hybrid mode can continuously predict the camera's direction even with highly obtuse angles between the user and the environment limits, or even

with different types of uncalibrated cameras. Through the work developed for this thesis, multiple keypoints features detectors and descriptor were benchmark in previously untested conditions regarding their scaling and obtuse angular performance; an additional analysis were made between brute force matching of said keypoints detected and artificial neural networks matching, including a deeper inquiry in one of the solutions already available, with the result of an increasing performance after the author's implementation. There is also presented a solution for 2D scanning a rectangular room and correlating its 3D boundaries to its vertical planes - walls. Another innovation presented is the filtration with Kalman filters of spatial outliers not only at the projected points, as was also done in this thesis, but also on the camera's axis, which smooth the AR projection without overly increasing the performance. One of the core developments of this thesis is the hybrid method of fusing a good homography, with additional refinements introduced on this thesis, with the geometrical vanishing point lines to be able to retrieve an almost perfect homography, which allows us to use uncalibrated cameras and achieve the same results. Furthermore, a continuous tracking method independent from the features matching database, allows for an advanced superimposition of AR content, achieving extremely obtuse angles while still maintaining the projection, even when the feature matching is lost.

All this previous work was limited to the use of only computer vision techniques and monocular cameras to achieve the desired outcome.

Over the last years, research has revolutionized itself through machine learning and deep learning. The birth of convolutional neural networks capable of achieving state of the art results across the vast fields of computer vision changed most of the consolidated algorithms. Aligned with this emerging area, a new method is introduced to estimate the room spatial layout using only a monocular image. Through a lightweight model for backbone and a semantic segmentation model, near state of the art results are achieved with a fraction of the parameters of the current methods. Initially a coarse semantic segmentation is achieved from where the layout type is hypothesized and ranked with a discriminative classifier. Afterwards, a sliding window method is used

for refinement. With the approached spatial information available, three vanishing points are estimated instead of calculated, which can become computationally cumbersome, allowing us to predict the camera pose estimation.

1.1 Scope of the Thesis

The content of this thesis began integrated as an additional module in the already running project Mobile Five Senses Augmented Reality System for Museums¹ (M5SAR), funded by Portugal2020, CRESC Algarve 2020 I&DT, n° 3322, promoter SPIC² (Sonha Pensa Imagina Comunica, Lda.) and co-promoter University of the Algarve³. The project began in January of 2016 and finished in October of 2018.

The project M5SAR's main goal was the development of an AR system solution aimed to enhance the museums' visits through the exploration of the humans' five senses: hearing, smell, taste, touch, and vision. It consisted in creating a synergy between a mobile device application and an additional device where a tablet or smartphone would perfectly fit, allowing for a unique and innovative experience when visiting any historical museum. This hardware device is capable of exploring 3 of those senses: smell, taste and touch, communicating with the application via a Bluetooth connection. The development of this device is out of the scope of this thesis. The software development encompassed the creation of an Augmented Reality application for mobile devices using only the available monocular cameras, being in its core the framework Mobile Image Recognition based Augmented Reality (MIRAR), which contained three modules: Object Detection, Environment Detection, and Person Detection. This framework would explore the remaining two senses: hearing and vision. The Object Detection module aimed at recognizing the artwork through computer vision markerless-based image recognition, using beacons to limit the amount of images to compare. Through this module, with the created device attached, it was possible to superimpose AR content over the artworks, where an immersive storytelling would

¹<https://sites.google.com/view/m5sar-microsite>

²<http://spic.pt>

³<http://www.ualg.pt>

begin activating the corresponding senses according to the story. The Environment Detection module consisted in the recognition of the walls of any rectangular cuboid room available at a museum in order to allow for the superimposition of content relative to the artworks' era, granting the possibility of for the visitor to peek into another age, further immersing it into the museums' experience. The Person Detection module was capable of recognizing the humans' shape and movement, which allowed for the superimposition of clothes from the artworks' generation. The Environment Detection module development is the main focus of this thesis.

The work presented in this thesis began in September of 2017 and its development continued after the end of the project M5SAR.

1.2 Objectives

The main objective of this thesis is to develop a system that would allow for the detection of the walls on a rectangular cuboid environment using only a monocular camera without the assistance of any 3D scanning or additional devices, and superimpose dynamic content over the walls. This implementation is aimed at mobile devices, mainly smartphones, and should be able to run in real-time. The specific objectives are described below:

- Development of a 2D scanning protocol and bundle creation;
- Location awareness with markerless-based image recognition;
- Camera's pose estimation calculation;
- Scalable image search mechanism;
- Improved homography refinement and recover;
- A progressive tracking system;
- Superimposition of dynamic content.

1.3 Structure of the Thesis

The structure of this thesis is presented by the compilation of the author's most relevant publications, where the thesis' objectives are further explored and analysed, with the additional publications being listed in Sec. 6.2. Therefore, each of the following chapters is presented as an already published publication, containing its own abstract, introduction, state of the art, methodology, results, and conclusion. Albeit the bibliographies were removed for simplicity and aesthetics, they are available at the end of this thesis. As the result of being a continuous work developed with the same objective, some information between the publications may overlap, without disregarding the innovations over the presented algorithm. In order to clarify the contributions made by the author of this thesis' to each chapter, it is detailed below an overall description of the author's work over the 3 main chapters of this thesis:

- Rodrigues, J.M.F., **Veiga, R.**, Bajireanu, R., Lam, R., Pereira, J., Sardo, J., Cardoso, P.J.S., and Bica, P. (2018) **Mobile augmented reality framework - MIRAR**. In 12th International Conference on Universal Access in Human-Computer Interaction, integrated in the 20th HCII, Las Vegas, USA, pp. 102–121

On Chapter 2, the author's main contribution is presented mainly on Section 2.4, introducing the initial skeleton of the search algorithm using artificial neural networks and a performance comparison using different parameters, such as the template's resolutions, the use of ORB or BRISK for feature detection and/or description, and the FLANN or Brute Force matching methods.

- **Ricardo J. M. Veiga**, João A. R. Pereira, João D. P. Sardo, Roman Bajireanu, Pedro J. S. Cardoso, João M. F. Rodrigues (2019). **Augmented Reality Indoor Environment Detection: Proof-of-Concept**. In WSEAS Transactions on Mathematics, ISSN / E-ISSN: 1109-2769 / 2224-2880, Volume 18, 2019, Art. 28, pp. 203-210

In this publication, on Chapter 3, the author of the thesis presents and hybrid version of his previous work. Advancing with a more robust detection algorithm mixing different scales of artificial neural networks, it is also presented a

more refined homography filtration and reconstruction. This is possible using the previous developed wall detection, which uses the environments' available geometrical characteristics, such as the common existence of planes that always convert to the vanishing point. An initial stabilization was also presented.

- Rodrigues J.M.F., **Veiga R.J.M.**, Bajireanu R., Lam R., Cardoso P.J.S., Bica P. (2019) **AR Contents Superimposition on Walls and Persons**. In: Antona M., Stephanidis C. (eds) Universal Access in Human-Computer Interaction. Theory, Methods and Tools. HCII 2019. Lecture Notes in Computer Science, vol 11572, pp. 638-645, Springer, Cham. DOI: 10.1007/978-3-030-23560-4_46

The contributions for this publication, on Chapter 4, consists in the improvement and evolution of the previous work and also the introduction of a progressive tracking based on the camera's pose estimation, see Section 4.3. It was also presented the initial fusion between the thesis work and the human content superimposition.

- **Veiga, Ricardo J.M.**, Cardoso, Pedro J.S., Rodrigues, João M.F. (2020) **Efficient Small-Scale Network for Room Spatial Layout Estimation** In Submission to 14th International conference on Universal Access in Human-Computer Interaction, integrated in the 22nd HCII, Copenhagen, Denmark, 19-24 July

On Chapter 5, a novelty method is proposed of indoor spatial layout estimation using a network with the fraction of the parameters of current methods. There is also a new post-processing algorithm for layout ranking and refinement. Near state of the art results are achieved, and an small application for camera pose estimation is introduced.

1.4 Overview of the Thesis

This present chapter introduced the theme of this thesis, as its main objectives, contributions and scope, consisting on three different published papers related to the envi-

ronments' detection, tracking and superimposition. In Chapter 2 is introduced a performance comparison between different templates' resolutions, features' detectors and descriptors, and keypoints' matching methods. Chapter 3 continues the previous work introducing a hybrid matching method using artificial neural networks, an improved homography refinement fused with the previous work done over the environments' geometrical characteristics. In Chapter 4 a progressive tracking is introduced, allowing for a smooth superimposition even when the matching is lost, and a fusion with the human content superimposition is presented. Chapter 5 presents a novelty solution for room spatial layout estimation using a smaller network accompanied by an application for camera pose prediction. Finally, in Chapter 6 we have the conclusions of this thesis and all the work done over the previous chapters, as well the future work and a list of all the publications that resulted from the work presented on this thesis. It is important to highlight that due to the continuous nature of the work presented in this thesis, some of the content across the main chapters will be very similar, notwithstanding the innovation introduced.

2

Mobile Augmented Reality Framework - MIRAR

Abstract

The increasing immersion of technology on our daily lives demands for additional investments in various areas, including, as in the present case, the enhancement of museums' experiences. One of the technologies that improves our relationship with everything that surrounds us is Augmented Reality. This paper presents the architecture of MIRAR, a Mobile Image Recognition based Augmented Reality framework. The MIRAR framework allows the development of a system that uses mobile devices to interact with the museum's environment, by: (a) recognizing and tracking on-the-

fly, on the client side (mobile), museum's objects, (b) detecting and recognizing where the walls and respective boundaries are localized, as well as (c) do person detection and segmentation. These objects, wall and person segmentation will allow the projection of different contents (text, images, videos, clothes, etc.). Promising results are presented in these topics, nevertheless, some of them are still in a development stage.

2.1 Introduction

Augmented Reality (AR) (Azuma et al. (2001)) is a technology that, thanks to the mobile devices increasing hardware capabilities and new algorithms, quickly evolved in the recent years, gaining a huge amount of users. AR empowers a higher level of interaction between the user and real world objects, extending the experience on how the user sees and feels those objects by creating a new level of edutainment that was not available before. The M5SAR: Mobile Five Senses Augmented Reality System for Museums project (Rodrigues et al. (2017)) aims for the development of an AR system that acts as guide for cultural, historical and museum events. This is not a novelty, since almost every known museum has its own mobile applications (App), e.g. Information-Week (2017); TWSJ (2017). While the use of AR in museums is much less common, it is also not new, see e.g. HMS (2017); Qualcomm (2017); SM (2017); Vainstein et al. (2016). The novelty in the M5SAR project is to extend the AR to the human five senses, see e.g. Rodrigues et al. (2017) for more details.

This paper focus on MIRAR, Mobile Image Recognition based Augmented Reality framework, one of the M5SAR's modules. MIRAR focuses on the development of a mobile multi-platform AR (Azuma et al. (2001)) framework, with the following main goals: (a) to perform "all" computational processing in the client-side (mobile device), minimizing, this way, costs with server(s) and communications; (b) to use real world two- and three-dimensional (2D and 3D) objects as markers for the AR; (c) to recognise environments, i.e., walls and its respective boundaries; (d) to detect and segment human shapes; (e) to project contents (e.g., text and media) onto different objects, walls

and persons detected and displayed in the mobile device's screen, as well as enhance the object's displayed contents, by touching on the device's screen regions on those objects; and (f) to use the mobile device's RGB camera to achieve these goals. A framework that integrates these goals is completely different from the existing (SDK, frameworks, content management, etc.) AR systems (Artoolkit (2017); Catchoom (2017); Kudan (2017); Layar (2017); Pádua et al. (2015)).

The MIRAR sub-module for object recognition and environment detection presented in this paper is AR marker-based, often also called image-based (Cheng and Tsai (2013)). AR image-based markers allow adding pre-set signals (e.g., from paintings, statues, etc.) easily detectable in the environment, and the use computer vision techniques to sense them. There are many image-based commercial AR toolkits (SDK) such as Catchoom (2017) or Kudan (2017), and AR content management systems such as Layar (2017), including open source SDKs (Artoolkit (2017)). Each of the above solutions have pros and cons. Between other problems, some are quite expensive, others consume too much memory (it is important to stress that the present application will have many markers, at least one for each museum piece), and others take too much time to load on the mobile device.

The increasing massification of AR applications brings new challenges to the table, such as the demand for planar regions detection ("walls"), with the more popular being developed within the scope of Simultaneous Localization And Mapping (SLAM) (Bailey and Durrant-Whyte (2006); Durrant-Whyte and Bailey (2006)). Usually, the common approach for image acquisition of 3D environments uses RGB-D devices or light detection and ranging (LIDAR) sensors (Hulik et al. (2014); Ring (1963); Xiao et al. (2013); Sousa et al. (2014)). There are also novelty advances within environment detection, localization or recognition, either using Direct Sparse Odometry (Engel et al. (2018)), or using descriptors, like ORB SLAM (Mur-Artal et al. (2015)) or even Large-Scale Direct Monocular SLAM (Engel et al. (2014)). However, as mentioned, the MIRAR framework focuses on mobile devices with only monocular cameras. Following this, an initial study of an environment detection sub-module was

previously presented in Pereira et al. (2017), being the purposed method a geometric approach to the extracted edges of a frame. It should be considered that a frame is always captured from a perspective view of the surrounding environment, with the usual expected environment being characterized by the existence of numerous parallel lines which converge to a unique point in the horizon, called vanishing point (Duan (2011); Serrão et al. (2015)).

The last topics addressed in the MIRAR framework regards the detection of human shapes in real world conditions. This continues to be a challenge in computer vision due to the existence of multiple variants, e.g., object obstructions, light variations, different viewpoints, the existence of multiple humans (occlusions), poses, etc., nevertheless, the detection of human shapes is an area with many studies and developments (Fang et al. (2017); Ouyang and Wang (2013); Sermanet et al. (2013); Tian et al. (2015); Zhang et al. (2016a)).

In summary, the MIRAR's object recognition sub-module uses images from the museum's objects, and the mobile device's camera to recognise and track on-the-fly, on the client-side, the museum's objects. The environment detection and recognition sub-module is supported upon the same principles of the object's recognition, but uses images from the environment, walls, to recognise them. Finally, the human detection and segmentation uses Convolutional Networks for the detection and an image processing algorithm for foreground (person) extraction. The main contribution of this paper is the integration of these three topics into a single mobile framework for AR.

The paper is structured as follows: The MIRAR framework and architecture is introduced in Sec. 2.2. Section 2.3 presents the main MIRAR's sub-module, namely the object detection, followed by the wall detection sub-module in Sec. 2.4 and the human shape detection in Sec. 2.5. The paper concludes with a final discussion and future work, Sec. 2.6.

2.2 MIRAR framework

Before detailing the MIRAR framework it is important to give a brief overview of the M5SAR system, shown on top of Fig. 2.1. On the figure's left side, the basic communications flow between the server and mobile device is outlined and, on the right side, the simplified diagram of the mobile App and the devices "connected" (via bluetooth) with the mobile device is shown. The displayed Beacons (Estimote (2017)) are employed in the user's localisation and the Portable Device for Touch, Taste and Smell Sensations (PDTTSS) (Sardo et al. (2017)) used to enhance the five senses. In summary, the M5SAR App architecture is divided into three main modules: (A) Adaptive User Interfaces (AUI), see Rodrigues et al. (2017); (B) Location module, a detailed explanation is out of this paper's focus, and (C) MIRAR module (see Fig. 2.1 bottom).

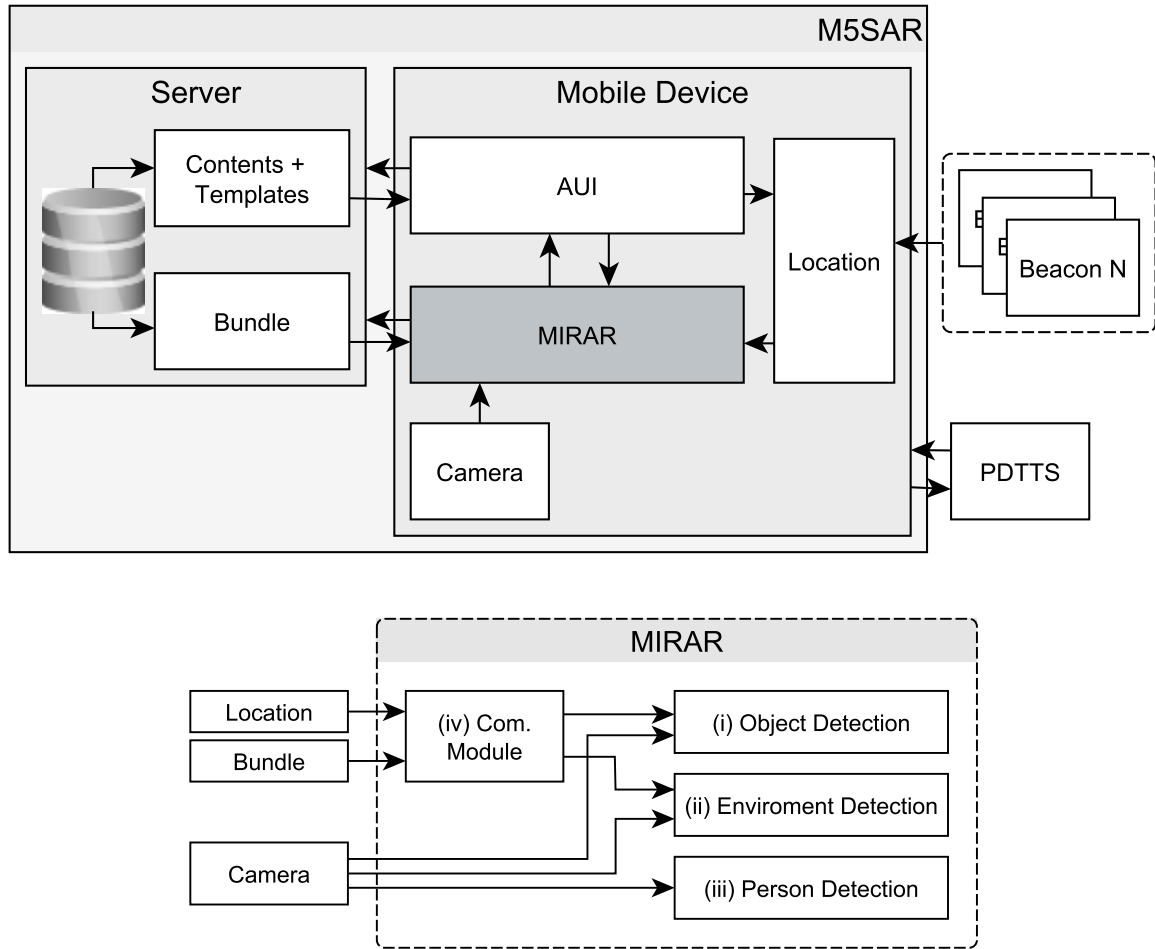


Figure 2.1: Top: overall simplified system architecture. Bottom: MIRAR block diagram.

The MIRAR has four main features: (a) the detection and recognition of museum objects, triggering a card in the (M5SAR) App (Rodrigues et al. (2017)); (b) the detection, recognition and tracking of objects as the user moves along the museum, allowing to touch different areas of the objects displayed in the mobile screen and showing information about that region of the object, MIRAR sub-module (i); (c) detection and modelling of the museum walls, and projecting information into the detected walls (e.g., images, movies, text) related with the recognized object's epoch, sub-module (ii); (d) detection of persons that are moving in the museum, and, for instance, to dress them with clothes from the object's epoch, sub-module (iii).

Sub-modules (i) and (ii) need to communicate with the server, i.e., the MIRAR module sends the user's position to the server, based on the previous object detections and the localisation given by the beacon's signals. From the server, the MIRAR module receives a group of object markers (image descriptors; see next section), here called bundles, that contain all the objects available in the located room or museum section. In a way to minimise communications, the App stores in the memory (limited to each device's memory size) the bundles from the previous room(s), museum section(s), and as soon as it detects a new beacon signal it downloads a new bundle. Older bundles are discarded in a FIFO (first in, first out) manner.

It is also important to stress that, since the sensor used to acquire the images from the environment is the mobile's camera, in order to save battery, the camera is only activated when the AR option is selected in the UI. When the activation occurs, the user can see the environment in the mobile screen and effectuate the previously mentioned actions. As an additional effort to save battery, the device will enter a low-power state if the user turns the phone upside down, by dimming the phone's screen and interrupting the processing.

As final remarks, the App was implemented using Unity (2018), the computer vision algorithms were deployed using the OpenCV (2017) library (Asset) for Unity, and tests and results consider that the mobile device is located inside a museological space. The next section will present the object detection and tracking module.

2.3 Object detection, recognition and tracking module

The object detection sub-module aim at detecting objects present in the museum, being the algorithm divided in 2 components: (a) detection and recognition, and (b) tracking. While the recognition is intended to work on every museum object, the tracking will only work in masterpieces¹. The masterpieces' tracking allows to place contents in specific parts of the UI, so that the user will touch on those areas in order to gain more information about a particular region of the detected object.

Before describing this module in further details, it is important to distinguish from templates and markers. Here, templates are images (photographs) of the objects stored in the server's database (DB) while, on the other hand, a marker is the set of features (keypoints) with their respective (binary) descriptors for a certain template, see Fig. 2.2 and Pereira et al. (2017). The authors' employ the ORB descriptor for keypoint detection and descriptors implementation (Figat et al. (2014); Pereira et al. (2017); Rublee et al. (2011)).

A generic image recognition and tracking algorithm for AR has the following main steps: (1) extract the markers (keypoints and descriptors) from a template; (2) extract keypoints and compute descriptors from query images (i.e., for each mobile device camera's frame); (3) match the descriptors of both the template and query; and, when needed, (4) calculate the projection matrix to allow perspective wrapping of images, videos, and other contents.

An initial recognition algorithm was presented in Rodrigues et al. (2017), with further advances presented in Pereira et al. (2017), as follows. Similar to Baggio (2012), in Step (1) it is utilised the image to extract keypoints and compute descriptors. The borders are the exception (e.g., the painting frame) which were removed, since usually there is no relevant information in those areas. Nevertheless, the templates are processed in different scales (image sizes): starting at the pre-defined camera frame size, 640×480 px (pixels), the templates are scaled up and down (by a $1/3$), resulting in a

¹Masterpieces are objects that have an enlarged (historical and cultural) value in the museum's collection.



Figure 2.2: Top: Example of existing objects in Faro Municipal Museum. Bottom: examples of detected and tracked markers with the corresponding axis.

total of 3 scales per template. To further increase the framework’s performance, these markers continue to be created on a server and sent to the client (mobile device) on demand, to be de-serialized. Step (2) from the frame acquired by the camera, the keypoints are simply extracted and their respective descriptors computed (using the ORB descriptor).

Regarding Step (3), the query image descriptors are (3.1) brute-force matched, using K-Nearest Neighbours (KNN), with $K = 2$, against the descriptors of the available markers. Next, (3.2) the markers’ descriptors are matched to the query’s descriptors. Following, (3.3) a ratio test is performed, i.e., if the two closest neighbours of a match have close matching distances (65% ratio), then the match is discarded (Baggio (2012)), because this would be an ambiguous match. This ratio evaluation is the test where most matches are removed. For this reason, this test is performed first to improve performance later on. Then, (3.4) it is performed a symmetry match where only the matches resulting from the KNN in (3.1) that are present in (3.2) are accepted. After this, a (3.5) homography refinement is applied. This refinement uses the RANSAC method to verify if the matched keypoints in the query image maintain the same con-

figuration between them (same relative position) as they had in the template image. If any of the keypoints stay out of this relation, then they are considered outliers and removed from the match set. (3.6) If after all of these refinements there are at least 8 matches left, then it is considered as a valid classification.

In the (3.6.i) classification stage the query image is compared using Brute-Force (BF) to all marker scales for each of the available templates. This, in turn, returns a classification based on the count of (filtered) matches, when there are at least 8 descriptors matches. The marker that retrieved the most number of matches is considered the *template to be tracked*. Afterwards (3.6.ii), if the tracking stage is necessary, i.e., if a masterpiece is present, the matching only occurs with the markers of the 3 scales of the *template to be tracked* previously selected in classification phase. If the object (*template to be tracked*) is not visible in the scene for 1 second then it is considered lost and the recognition process initiates again. Last, but not least, Step (4) of the generic algorithm is done using perspective wrapping (pose estimation) in order to place content on the same plane as the detected image (marker).

Figure 2.2 bottom shows some examples of tests done in the Faro Municipal Museum where the classification number is shown in red. For more algorithm details and results see Pereira et al. (2017).

2.4 Environments detection

As previously mentioned, the objective of this sub-module is to be able to discern the location and position of the walls of a given environment, and afterwards replace them with other contents. The algorithm presented here does not yet supports our investigation over this subject mentioned in Pereira et al. (2017), although it will eventually merge with the preceding work. In the present case, similar to Sec. 2.3 markers are used (keypoints and descriptors) for each template, with the bundles being previously generated. For this sub-module, the templates are of the entire walls, and not only of the museum's object, see Fig. 2.3 top row, which allows for the retrieval of the expected

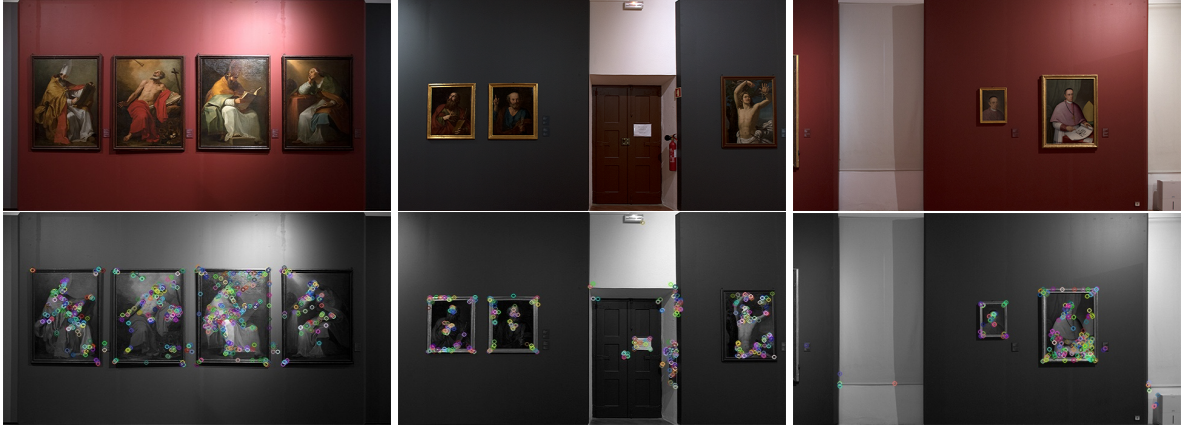


Figure 2.3: Top: Example of templates. Bottom: Extracted keypoints location for each template.

wall shape using a pose estimation algorithm. Various implementations of pose estimation have already been presented for 3D objects using RGB-D sensors (Buch et al. (2013)) or through monocular images (Riba Pi (2015)), and urban environments (Hallquist and Zakhor (2013)). The main contribution of this sub-module is the initial implementation of the wall estimation for primarily indoor detection and recognition while the user navigates through museums, which are presented in this chapter, with a future user localization feature already being developed. Regarding our implementation of wall estimation, it is important to note that the aim for this sub-module is a seamlessly fully integrated AR application for mobile devices; therefore, the presented algorithm is focused and adjusted for performance on smartphones.

Contrary to what was presented for the object detection, instead of using ORB descriptors, we found that BRISK (Leutenegger et al. (2011)) descriptors perform better for this task, which will be explained later in Sec. 2.4.1. For comparison between markers, not only Brute-Force was tested, but also the *Fast Library for Approximate Nearest Neighbours* (FLANN) (Muja and Lowe (2012)). The necessity of evaluating both matchers for this task will be posteriorly explained.

The current algorithm, after the bundle has been created and loaded, is applied to each frame from the mobile device camera as follows: (1) A number of the most significant keypoints are retrieved (filtered) and the respective descriptors computed; (2) Optimal matches are found and filtered; (3) Pose estimation is performed after discard-

ing the defective homography; (4) Polygons corresponding to the matched templates are superimposed on the frame.

Beginning with Step (1), all the keypoints (using the BRISK keypoint detector) found from the frame provided are ordered by their response, which defines the ones with stronger information, and only an amount of keypoints is maintained, which in our case was $Num_{KP} = 385$ (this number was empirically computed, see Sec. 2.4.1). More keypoints will not improve results and it increases computational time. If this number is decreased significantly, many "template (wall) - frame" matches are lost. Afterwards, the respective descriptor for each keypoint is computed using the BRISK descriptor.

In Step (2), before searching through all the stored templates, it is verified if the location of the user is known through the previous frames, thus allowing for the matching search to begin with the surrounding templates. The method used for matching was K-Nearest Neighbours (the same as in Sec. 2.3), with $K = 2$, either by BF matching, or using FLANN. While the BF compares all the retrieved markers' descriptors from the frame with the stored markers from the templates, it was created an index with FLANN that uses multi-probe LSH (Lv et al. (2007)). The parameters used were 6 hash tables, with 12 bits hash key size, and 1 bit to be shifted to check for neighbouring buckets. The number of times we defined for the index to be recursively traversed was 100, as we observed a good balance between additional processing time and the increased precision. It is important to refer that, as opposed to BF, FLANN does not return a complete matching between the markers, but instead it gives an approximate correspondence. The remaining matches are filtered through the Lowe's ratio test, where we discard the pairs with close matching distances (65% ratio), allowing only the more distinct ones to remain. Subsequently, if at least 10 good matches are found, then the perspective transform is retrieved through the homography refinement using the RANSAC method, where the original pattern of keypoints from the templates are compared with the ones from the frame, considering the ones with the same configuration as inliers, and the others as outliers.

Regarding Step (3), for the pose (wall) estimation templates to be properly found, the perspective matrix must be found valid. It should be noted that the existing planes across the provided frames will be randomly presented with acute perspective angles, or at deeper distances. Concerning the templates' format, for this sub-module we chose to include the desired full wall delimitations to be found, even if the regular walls did not offer relevant information to be retrieved, with the keypoints gathered in clusters along the museums' objects, see in Fig. 2.3.

The chosen template format after the pose estimation returned the approximated horizontal limits of the walls. In order to improve accuracy and performance, it was necessary to discard the non relevant perspective matrices. To do so, we analysed the matrix extracted from the homography and applied a group of tests. We calculated the determinant of the top left 2×2 matrix and limited the output between 1 and 100, given that, with the perspective transform, if the values of said determinant were to be negative there would be an inversion, and as the templates were created for the expected projection, there should be none. The limit of 100 was imposed because in case there was a large value for the determinant, then the aspect ratio would have been overly deformed. Furthermore, after finding the coordinates, their order is compared against the original template, e.g., if the (x_0, y_0) of the template is on the top left and the (x_1, y_1) on the lower left; then, after the perspective transform, this orientation should remain. Afterwards, it is verified if the angles between each 3 points are not overly convex, as they are expected to be nearly perpendicular. Finally, as the environment/room is "regular", which means the presence of vertical walls without deformations (no circular walls) or extensive 3D artwork, all the non vertical resulting polygons with an error of 15% are discarded.

The last Step (4), the retrieved coordinates are converted into polygons that are superimposed upon the original frame for each of the matched templates, corresponding to the expected surface of the wall in the environment; a sequence of the output results can be seen in Fig. 2.4. With this outcome it is possible to project content not only replacing the walls, as presented before in Pereira et al. (2017), but also to present float-



Figure 2.4: Example of a sequence of frames with matched templates. See also Fig. 2.6.

ing AR content. It is important to stress that when the angle between the wall and the mobile user is “too sharp” is not yet possible to find the boundaries of the wall, which can be shown in Fig. 2.4 bottom row.

2.4.1 Tests

In order to test the reliability of the algorithm, tests were done before converting the algorithm to the mobile platform; nevertheless, all the videos used for the tests were acquired by mobile devices (smartphones and tablets). The tests were done using a desktop computer with an Intel CPU i5-6300 running at 2.4 GHz with the algorithm limited to run in single-thread. The videos consisted of a total amount of 4,306 frames of expected user navigation through the museum, with both the horizontal and vertical orientations used. An additional video containing persons in between the camera and the walls also showed good results, as seen in Fig. 2.6. It is important to note that it

is expect that this sub-module will not always detect and recognize the environment in all the frames; therefore, the most important measure of success is the amount of frames with valid matches found.

The tests were conducted in following ways: the templates' width size between 320px and 640px; the frame's width size between 640, 480 and 320px; and increasing the number of minimal good matches, starting with 10 and using steps of 5. All the tests were performed using BF and FLANN.

Regarding the variation of the minimal good matches between markers to begin the template matching, as expected, with the increase of this threshold the amount of frames with a found template match were dramatically reduced, while it was observed the maintenance of a similar processing time, either for each frame as for each matched frame, showing little to no variation. The results in terms of "pose" estimation of the polygons over the output frame were also improved. Upon reviewing this results we decided to use the minimal value of 10, given that it returned the highest number of frames matched without overly increasing the undesired results; for example, changing this value to 15 reduced the frames matched by 35 – 40%. With the variation of the templates width size, it was expect to add additional detected matches for when the wall is distant and is presented smaller on the frame. The results showed that even when it occasionally happens, it doesn't justify adding different scales of templates for this sub-module at the current version in exchange of processing time performance, as it was presented in Rodrigues et al. (2017).

The obtained processing times for the current algorithm, while reducing the templates and frames width, decreased from $28,3 \pm 13,8$ ms to $17,9 \pm 5,9$ ms with BF, and from $33,2 \pm 14,2$ ms to $24,1 \pm 17,8$ ms. Even though it presented some improvement on the time performance, the amount of frames matched dropped 73 – 63% respectively.

Considering the necessity a higher rate of matching, the performance of different templates and frames sizes were compared. The illustration in Fig. 2.5 presents the amount of frames matched with the colour black within the total frames of different expected user interaction videos. On the left is shown the frame number (1,..., 4306),

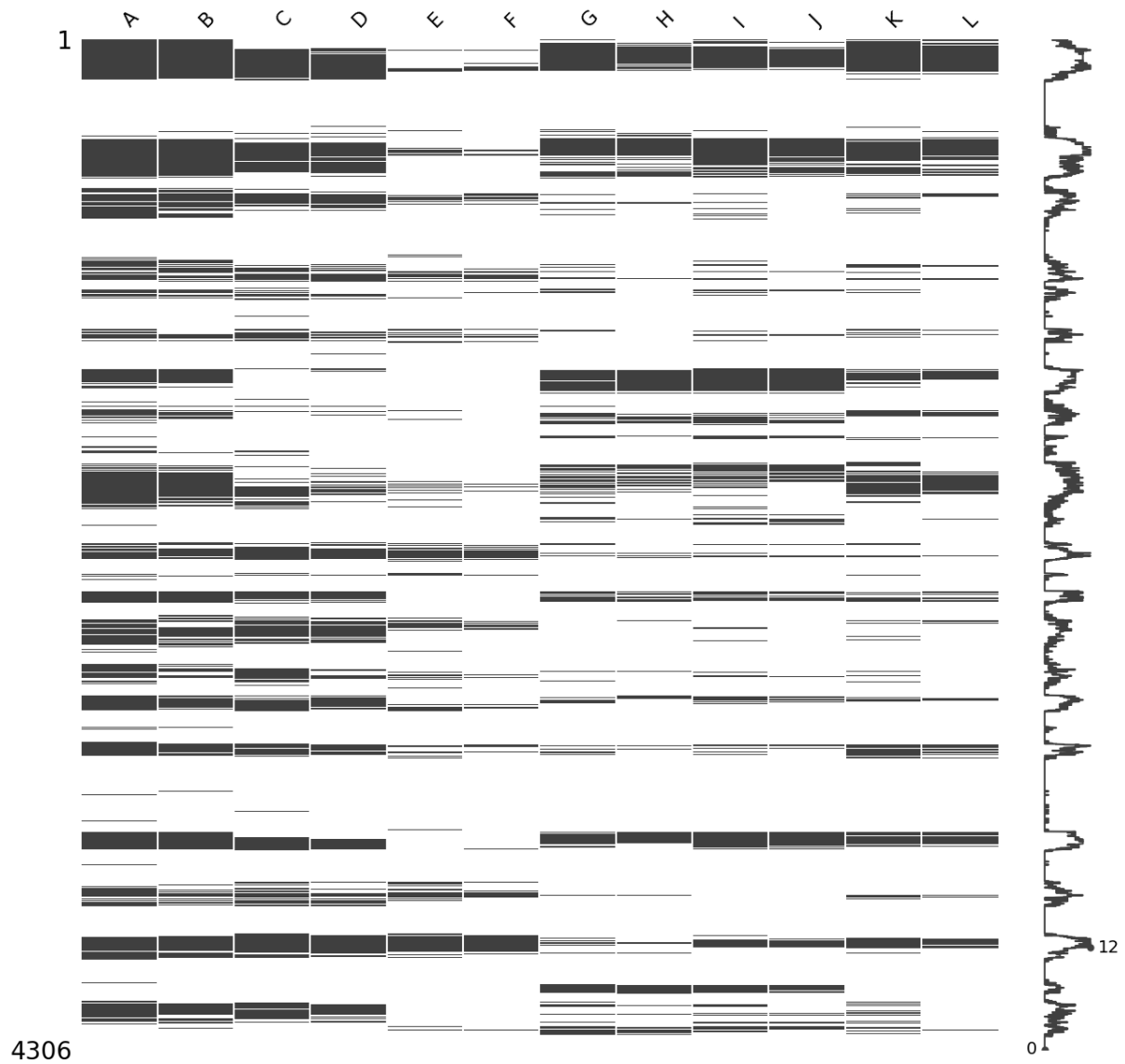


Figure 2.5: Illustration of the number of matched frames (in black) with templates, changing the widths of both, i.e., from A-F is shown for each pair Flann (A, C and E) and BF (B, D and F) the matched templates with a width of 640px, the frames' width varies from 640 (A and B), to 480 (C and D) and 320px (E and F). From G-L the same but now with the template with the width of 320px. At right, an histogram of the total of matches for each frame is shown.

and on the right, the histogram of matched template-frame along the different widths and matching algorithms, namely FLANN and BF. The intent of this comparison, other than reporting the total of matched frames for each different test, was to analyse if the scale factor would introduce new results, either by reducing the frame width while retaining a larger template width, which is shown from A-F, or by reducing the template width, while again changing the size of the frame, as can be seen from G-L, with a

template width of 320px. The test were also divided in FLANN (A, C, E, G, I and K) and BF (B, D, F, H, J and L). The variation of frames' width is organized as 640 (A and B), 480 (C and D) and 320px (E and F), with the same from G-L.

Using the illustration shown in Fig. 2.5 it becomes easier to analyse the effect of the different parameters, being one of the main comparisons the use of FLANN or BF for feature matching. Is it relevant to note that, as BF needs a complete certainty for matching, which would be achieved more easily if the desired matching template was in full frame, FLANN does not, which introduces the possibility of the existence of false positives. As a continuous outputs of false positives from the same template is uncommon, it became easier to discard them. Although FLANN versus BF is usually restricted to a large amount of keypoints, in our case for 640px templates the average was 384.5 ± 119.48 keypoints per image, where FLANN surpasses BF in processing time performance, for our case it was primarily used with the intend of retrieving additional matched frames. Regarding the obtained results, FLANN returned additional matched frames with 38% of the total number of 4.306 frames matched, and BF returned 32% for 640px templates, while for 320px we obtained 20% and 15%, respectively. Although there is an increase in processing time for FLANN in order of 17% facing BT, when analysing the results shown in Fig. 2.5, it is possible to verify a more sparse occurrence of matched frames for FLANN versus BF, allowing for a higher probability of matching while the user navigates the museum, which will be used in order to recalibrate (in the future) the user localization and focus, improving the projection of content tracking and stability.

Focusing on the 640px width templates, it is possible to observe the expected lower matching while reducing the frames' width, for the total of available markers for each frame was reduced against the templates average number of extracted keypoints. With the 320px templates, the results showed a different outcome. While the total matched frames was also reduced, it became almost invariable across the tests, which means that with a lower processing time the same results would be achievable. One interesting point is noticeable near the medium point, where there was more matched frames

acquired with lower templates' width. This is explained with the distance of the matching template, i.e., if the frame's width is 640px and the templates' 320px, if the respective location of the template is inside the frame at distance, it will be closest to the lower templates' width than the larger, and as we are using BRISK descriptors, even if they are invariant to scale, there is a threshold to the maximum of that invariance. In conclusion, for these results it can be seen that in the future the implementation of different scales of templates to improve the localization tracking may be needed.

As referred before, for this module a BRISK descriptor was used instead of ORB. Although the amount of frames with matched templates was similar in between, ORB performed slightly better, with 771 frames with match against 726 of BRISK, from a total amount of frames of 1.857. The outcome of said matches presented worse projected polygons, meaning that even with the homography additional sanity tests, the occurrence of bad homographies increased. Furthermore, an increase in the occurrence of false positives with a factor of ten times between ORB and BRISK was observable, which largely contributed to the bad homographies received. The performed times from ORB to BRISK lowered from $37.8 \pm 9.5\text{ms}$ to $27.1 \pm 10.4\text{ms}$ while using FLANN, and $36.7 \pm 9.6\text{ms}$ to $26.0 \pm 8.5\text{ms}$ with BF. Is important to observe that for our tests using ORB, FLANN does not seem to affect the performance times. Lastly, the amount of average keypoints retrieved from the templates actually decreased from ORB to BRISK, with a total of 5.824 keypoints, with an average of 485.3 ± 24.4 keypoints per template, to a total of 4.614 keypoints, with an average 384.5 ± 119.5 , which shows ORB being more consistent than BRISK for the amount retrieved, although it did not prove that with the additional keypoints the results would improve.

As the objective of this sub-module is the recognition and detection of the walls throughout the visit within some rooms of the museum, it is expected a loss of tracking/recognition and its recovery at a slightly different location or angle, which demands that the recognized template shape be the as close as possible to the desired, every time there is a match within frames. Since the amount of good expected results with BRISK surpassed the ORB descriptor, we decided to perform the current algo-



Figure 2.6: Top row, examples of different vertical matching outcomes. Middle row, initial results of matching while the view is obstructed by persons. Bottom row, examples of the Hough (1962) Transform applied to different frames.

rithm using only the BRISK descriptor, while the algorithm for object detection shown at Sec. 2.3 remains using ORB. The different outcomes between both the descriptors for this challenge could be due to the fact that the BRISK descriptor is invariant to scale, while the ORB is not. Furthermore, while for object detection we used scaled versions of the templates for matching, as the object is expected to fill the screen of the mobile device, with this module there was an additional inclusion of different scales, as it should be considered that the user will be navigating the different rooms of the museum; therefore, the templates, in this case the walls, will appear on the mobile device with different geometric shapes and distances; see also Pereira et al. (2017).

Additional examples of results obtained can be observed in Fig. 2.6, where is possible to see on top the algorithm working with vertical frames and some of the still occurring bad outcomes retrieved from faulty perspectives transforms from the homography. On the 2nd row, results of the current algorithm can be observed, while the

view is partially obstructed with people. Additionally, it is important to remark that this module is being ran only from the frames obtained, without additional sensors or 3D information, and so, the results obtained with obstructed views are a welcoming result for the current implementation.

In Fig. 2.6 bottom row it is possible to observe the results of part of the former algorithm (Pereira et al. (2017)) applied to the retrieved frames. While with the former implementation the process would begin by elimination of all non relevant lines, or in our former case, all the non vertical, horizontal and vanishing lines, the future fusion of both developments will be more focused in only retrieving the nearest lines to the already extracted polygons through the Hough (1962) Transform, improving the polygons veracity to the actual walls' shapes and adding a level of longer distance detection, either by additional calculations through the use of the vanishing point together with the vertical and horizontal lines, as can be seen in Pereira et al. (2017), or with an eventual introduction of a pre-known room shape, which, combined with the user localisation, would achieve better results, presenting the opportunity for the use of indoor 3D models, further increasing the user immersion with AR.

On the next section we introduce an initial study for the detection and segmentation of the human shape.

2.5 Human shape detection

Human shape detection, as mentioned in the Introduction, already presents its challenges; furthermore, for this sub-module we have to consider the detection in real-time on a mobile device, while the user is moving through six degrees of freedom (6DOF), which will increase the level of complexity (Bhole and Pal (2012); Park and Yoo (2014)). Recent researches approach the human shape detection either by a top-down or a bottom-up method. Top-down means that the persons' shape are first detected and afterwards an estimation of their poses is achieved (He et al. (2017); Hernández-Vela et al. (2012); Papandreou et al. (2017)), while with bottom-up the humans' limbs are individ-

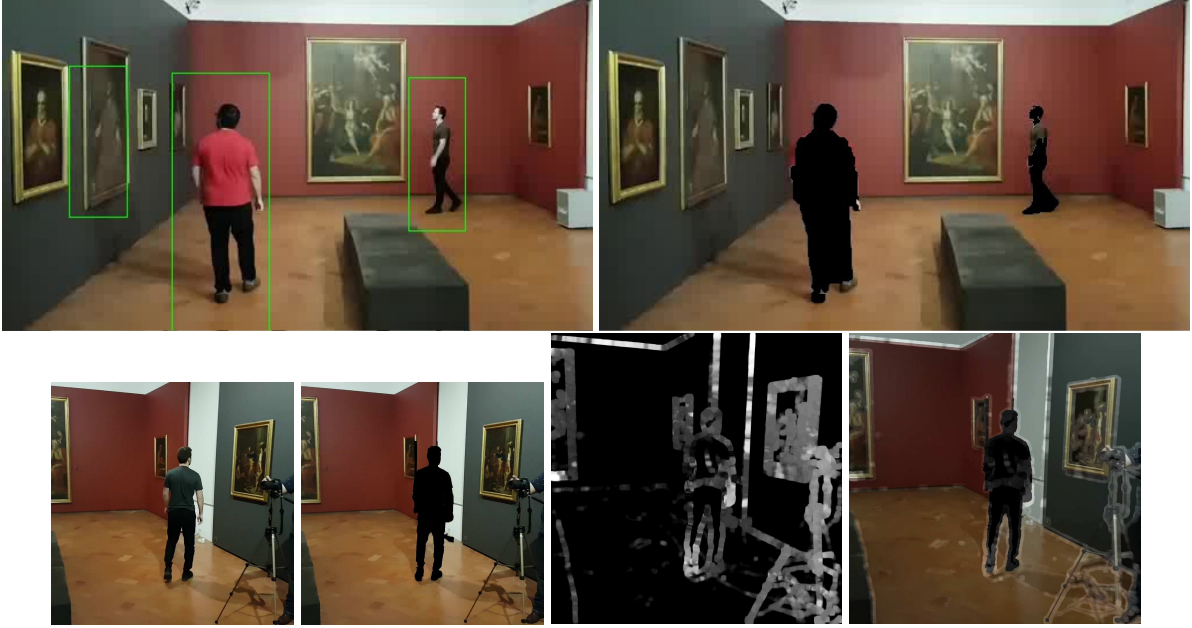


Figure 2.7: Top row-left, example of human detection with SSD-Mobilenet. Right, the result of human segmentation by GrabCut. Bottom row, from left to right, original image (frame), human segmentation, computed optical flow of two consecutive frames and overlap of segmentation and optical flow (the two former images).

ually detected, generating groups of body parts in order to form humans' poses (Cao et al. (2017); Fang et al. (2017)). For the initial study of this module we used a top-down approach, being the objective the detection and segmentation of the humans' shapes, allowing for the projection of AR content over it, as for example, the ability "to dress" the museums' users with clothes corresponding to the desired surrounding epoch of the museums' objects.

In order to overcome the complex challenges imposed by the detection of human shapes in video, captured by a moving camera of a mobile device, we used a convolutional neural networks (CNN), built in TensorFlow (Google (2018)). To detect human shapes in a video feed with reasonable rate of *fps* we used a single shot detection (SSD) network, and we used the MobileNet model for the neural network architecture; as its name suggests, it is designed for mobile applications (Howard et al. (2017a)). The other technique used in the process of human detection/segmentation is the GrabCut algorithm (Park and Yoo (2014)). It has a limitation where it needs to define the foreground and background areas; hence, we propose a fully-automatic human segmentation method by using the bounding box as a basis for the foreground and background

areas.

The algorithm for this module is executed for each frame following these steps: (1) Apply SSD-Mobilenet (Huang et al. (2017)), used for human detection, which outputs a bounding box around the detected humans (see Sec. 2.5.1 for the justification). (2) Resize the extracted bounding box by an increase of 10%; the original bounding box would cut some parts of the human shape in some image conditions, thus this improves the foreground precision. Step (3) follows with a cut of the input image, the cut is made with twice (2x) the size of the initial bounding box, with the same centre, and inside, the cropped area is used as background. Finally, (4) we use the GrabCut (Rother et al. (2004)) algorithm for human shape segmentation.

2.5.1 Tests

Three models were tested in the mobile device for the human shapes detection, SSD-Mobilenet (Huang et al. (2016)), YOLO (Redmon and Farhadi (2016)), and DeepMulti-Box (Erhan et al. (2014)). Empirical tests were done in a Museum (Faro Municipal Museum) using an ASUS Zenpad 3S 10 tablet, showing that in real world conditions the SSD-Mobilenet presented a better accuracy and speed, validating what is mentioned in Huang et al. (2016). Initially, we used COCO (Lin et al. (2014)) frozen pre-trained weights for SSD-Mobilenet. The evaluation setup, as mentioned, consisted on a ASUS Zenpad 3S 10 tablet and a windows machine with an Intel i7-6700 CPU @ 3.40GHz. A total of 86 frames of data were run 15 times through the network, with their performance being recorded. The resolution of the input frames was 640px and the spatial size of the convolutional neural network was 320px. To filter out weak detections we use a confidence threshold of 0.25. Our tests for this model, using the tablet, returned an average processing time of 346.0ms, while the computer achieved 33.7ms, being these values only regarding Step (1).

After the human detection, and for the remaining Steps (2-4) in our algorithm, with the use of GrabCut for segmentation, we achieved an average processing time of 127.3ms using the computer. In Fig. 2.7 top row is possible to observe the output

frame showing promising results regarding human segmentation. An observable disadvantage for this module within the museum environment can also be observed with a painting of a person being detected as living person. Although it is a good feature, for this task it is an undesirable result. Furthermore, in Fig 2.7 bottom row is possible to inspect that, when the conditions provide a discriminative foreground and background areas, the GrabCut algorithm can perform with high precision.

To solve the problem caused by mis-segmentation of the limbs of a segmented person, as for example in the left image, where the arms are indistinguishable from the torso, we started to apply Gunnar Farneback's algorithm to compute dense optical flow between two consecutive frames (Farneback (2003); Fleet and Weiss (2006)). This allows to complement the GrabCut segmentation process by using the consistency of pixel values in two frames. Using Farneback's algorithm allows to estimate the optical flow in a sequence of frames, and it is possible to use it to locate the borders of limbs that do not appear in the GrabCut segmentation. The algorithm shows an optical flow field with distinguished values between torso and arms because they have different speed movements, see Fig. 2.7 bottom right.

2.6 Conclusions

This chapter presents the current Mobile Image Recognition based Augmented Reality (MIRAR) framework architecture. Even in its current state, MIRAR had already presented good results in the object detection, recognition, and tracking sub-modules. The integration with the new approach for the wall detection and recognition shows satisfactory results, taking in consideration that it is still a work in progress. For the human shapes detection, initial results were shown; nevertheless, more consistent tests need to be performed in different museum conditions.

For future work, the recognition of 3D objects is an immediate focus in terms of creating a robust bank of tests, and so is the refinement of the object recognition and tracking module. This can be achieved by refining the matches with homography and

trying to find an optimised set of keypoints from multiple scales. For the wall detection, the focus will be on improving the stability and further filtering the occasional bad results, introducing pre-tuned templates to increase the range of detection, while preserving performance, the inclusion of a tracking system, and a merger with the previous work presented on edges detection for geometric prediction to stabilise the resulting polygons, reaching for a predictive localization of the surrounding indoor environment. The current different choices of descriptors between objects and wall detection will also be addressed.

For the human shapes detection, the segmentation done with the use of the Grab-Cut algorithm needs to be complemented in order to acquire a good human segmentation, since it will allow the projection of contents onto those shapes/persons. In the future we plan to use optical flow estimation (with initial results already shown) in the final segmentation process in order to improve the segmentation results. Additional work needs to be done to reduce the execution times of the detection and segmentation.

As a final conclusion, the MIRAR shows, even in this current stage, promising results, and it is expected to be an excellent tool to give a more impactful relation between the museum's user and the museum's objects.

3

Augmented Reality Indoor Environment Detection: Proof-of-Concept

Abstract

The conventional museum experience offers the visitors glimpses of the past with the narrative limited to the static art that garnishes it. Through technology we already can mix the past with the future, immersing the visitors in a true dynamic journey across the same walls that guard our history. One of this technologies is the Augmented Reality, which aims to enhance our surroundings into a new era of creativity and dis-

covery. This chapter presents the proof-of-concept of an indoor portable environment pose estimation module (PEPE) present inside M5SAR, a project that aims to develop a five senses augmented reality system for museums. The current state of development of this module shows that is already achievable real-world wall(s) detection and a new environment superimposition over the detection, i.e., it is now possible to have a dynamic museum experience with the ability of transforming rooms into historic live stages.

3.1 Introduction

Augmented Reality (AR) (Azuma et al. (2001)) has benefited from the increased hardware capabilities of smartphones and novelty algorithms, resulting in a fast evolution over a short time, rapidly growing its number of users. It allows for a higher level of interaction between user and real-world objects, expanding this experience and creating a brand new level of edutainment. The M5SAR: Mobile Five Senses Augmented Reality System for Museums project (Rodrigues et al. (2017)) aims for development of an AR system that acts as guide for cultural, historical and museum events. Most museums have their own mobile applications (App), see e.g. InformationWeek (2017); TWSJ (2017), and some also have AR applications, see e.g. HMS (2017); Qualcomm (2017); SM (2017); Vainstein et al. (2016). The innovation in the M5SAR project is to extend the AR to the human five senses, see e.g. Rodrigues et al. (2017) for more details.

The Mobile Image Recognition based Augmented Reality Framework (MIRAR) framework is one of the modules of M5SAR project (Pereira et al. (2017)), aims to: (a) perform all computational processing in the client-side (mobile device); (b) use in real world with 2D and 3D objects as markers for the AR; (c) recognise environments, i.e., walls and its respective boundaries; (d) detect and segment human shapes; (e) project contents (e.g., text and media) onto different objects, walls and persons detected and displayed in the mobile device's screen. A framework that integrates these goals is completely different from the existing (SDK, frameworks, content management, etc.)

AR systems (Artoolkit (2017); Catchoom (2017); Kudan (2017); Layar (2017); Pádua et al. (2015)).

This chapter focus on one of the MIRAR sub-modules (sub-module c)), the environment detection and overlapping of information. Considering a typical museum wall, there is usually artwork such as paintings and tapestry hanging on the walls, creating an unique rich environment full of visual information. Following the previous method introduced on the main object recognition module of MIRAR (Pereira et al. (2017)), we will use the features detection and description matching methods for the environment recognition. Considering the expect walls as planes, and taking into account the limited input information obtained from a monocular camera and smartphone performance, any methods of 3D matching, such as bundle adjustments, iterative closest point, among others, were discarded. Furthermore, with planes, it is possible not only to perform a faster recognition using the same methods used for object recognition, but also use the vanishing lines provided by the common geometric rules, for which we considered the existence of paintings' frames as a guarantee for the existence of vanishing lines.

In this chapter the contribution is to fuse both approaches in order to achieve a better wall detection and also user's localization so that we can more accurately project content upon the walls through the use of AR superimposition.

The MIRAR sub-module for object recognition and environment detection presented in this paper is AR marker-based, often also called image-based (Cheng and Tsai (2013)). AR image-based markers allow adding easily detectable pre-set signals in the environment, using computer vision techniques to sense them. There are many image-based commercial AR toolkits (SDK) such as Catchoom (2017) or Kudan (2017), and AR content management systems such as Layar (2017), including open source SDKs (Artoolkit (2017)). Some are expensive, others consume too much memory (and the present application will have at least one marker for each museum piece), while others load slowly on the mobile device. The increasing massification of AR applications brings new challenges, such as the demand for planar regions detection (walls),

with the more popular being developed within the scope of Simultaneous Localisation And Mapping (SLAM) (Bailey and Durrant-Whyte (2006); Durrant-Whyte and Bailey (2006)). RGB-D devices or light detection and ranging (LIDAR) sensors (Hulik et al. (2014); Ring (1963); Xiao et al. (2013); Sousa et al. (2014)) usually used for image acquisition of 3D environments. Some advances within environment detection, localisation or recognition include using Direct Sparse Odometry (Engel et al. (2018)), or using descriptors, like ORB SLAM (Mur-Artal et al. (2015)) or even Large-Scale Direct Monocular SLAM (Engel et al. (2014)). However, the MIRAR framework focuses on mobile devices with monocular cameras only. Following this, an initial study of an environment detection sub-module was presented in Pereira et al. (2017), using a geometric approach to the extracted edges of a frame. A frame is always captured from a perspective view of the surrounding environment, with the usual expected environment being characterised by the existence of numerous parallel lines which converge to the vanishing point (Duan (2011); Serrão et al. (2015)).

The chapter is structured as follows: The environment detection and AR overlapping is presented at Sec. 3.2 and concluding with a final discussion and future work, Sec. 3.3. For the MIRAR framework and architecture please see Sec. 2.2.

3.2 Environment Detection

The conventional museum's environment is rich in details provided by the multiple artwork that embellishes it. This scattered information is always present along the visitor's navigation throughout the museum when in presence of artwork. In continuity of our previous work presented in former publications (see e.g. Rodrigues et al. (2016); Pereira et al. (2017); Rodrigues et al. (2018); Sardo et al. (2017)), the vast presence of unique features along the museum allows us not only to be able to superimpose content over the walls, but also to locate the visitor's position within the museum. The visitor's localisation is also used within our main module of object recognition, but using Bluetooth beacons instead.

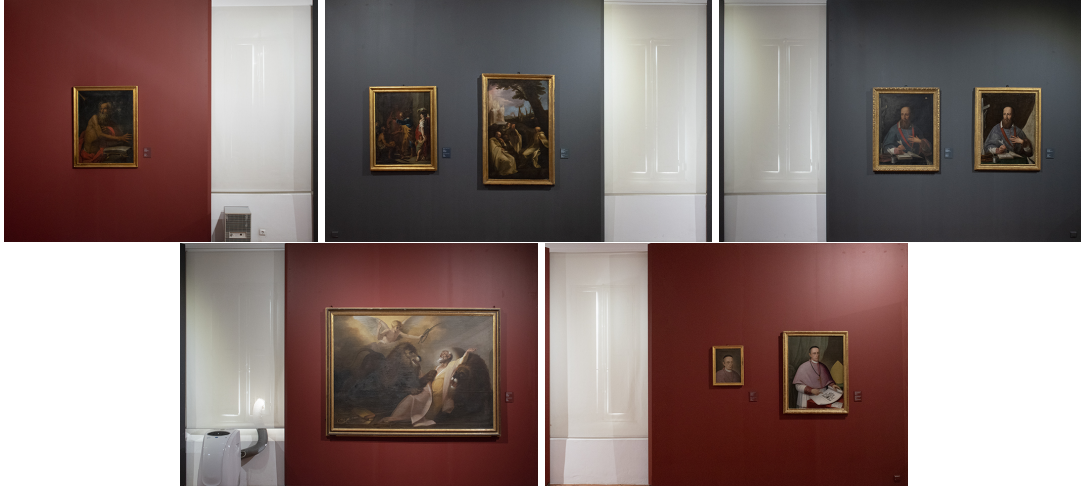


Figure 3.1: From top to bottom: Example of five templates following each other.

Also, in previous papers (Rodrigues et al. (2018); Veiga et al. (2017)) two distinct approaches were presented to solve the environment detection, one focusing on the geometry shape of a regular museum's division, assuming that all the vanishing lines are presented by the present walls within such division; while the other focuses on the recognition of already known parts of said walls. In this paper both methods are fused together in order to achieve a harmonious detection, recognition and localisation of the environment, to dynamically superimpose different types of content over the walls, such as images, video, animations, or 3D objects.

Before continuing, it is important to remind that due to the necessity of regular cuboid rooms and image recognition, the method presented in this paper is intended to be only used on previously scanned and prepared environments. It is also relevant to remind that the purpose of this AR application is to be able to run seamlessly on any current monocular smartphones, from which only a RGB image is provided by the camera, without any additional depth information.

The current algorithm divides itself in four different stages: the bundle creation (a), the recognition and localisation (b), tracking (c), and superimposition (d).

It is self-explanatory that the first stage is not performed inside the runtime, see below for a detail explanation, while the other two complement each other. All of the results presented were obtained running on an Intel i7-4820K CPU running on a single-thread. Beginning with the bundles' creation (a), for this task there are two distinct

Features	Number of Features	FLANN Based Matcher	FLANN Index	Performance Difference	Parameters
AKAZE	9500	10.62 ms	3.09 ms	-70.90 %	Default
BRISK	14521	22.64 ms	10.19 ms	-54.99 %	Default
BRISK	11299	17.01 ms	3.5 ms	-79.42 %	thresh=30, octaves=5, patternScale=float(2.0)
ORB	10891	11.72 ms	5.35 ms	-54.35 %	Default
ORB	35311	40.62 ms	22.90 ms	-43.62 %	nFeatures=2000

Table 3.1: Comparison of the performance between FLANN Based Matched and FLANN Index, presenting the results obtained from the matching of a real world image to an index of 71 prepared images.

types of bundles created: a FLANN Index (FI) bundle, and a FLANN Based Matcher (FBM) bundle. The reason for this peculiar choice is based on performance evaluations made while testing the multiple alternatives available, being the Brute Forced Matcher out of the scope of this paper, due to its lack of "flexibility" present on a previous publication. Both methods used the same index parameters, with the chosen algorithm being the Locality-Sensitive Hashing (LSH), due to the choice of using non-patent binary descriptors, the number of tables used were only 1, with a key size of 12, and only 1 multiprobe level. The addition of a multiprobe to the LSH allowed for the reduction of the number of hash tables, which allow for a better performance while maintain the same obtained results. We observed an average reduction of 76.56% of processing time across different binary features detectors and descriptors (AKAZE, BRISK, ORB) (Tareen and Saleem (2018)), with the default and tweaked specifications, while using only 1 hash table versus the 6 hash tables originally recommended, with the corresponding images' indexes returned with equal accuracy.

Regarding the choice of having two bundles of similar matchers, although the FBM is build upon the FI, we performed search tests with the same query image on both and obtained a better result retrieving the matching image index by and average of 60.66% less processing time while using the FI, as can be seen at table 3.1. This justifies the creation of an FI bundle, although while matching using the FI only the original image index is retrieved, accompanied by the KNN's distances. This way, it is only possible to know what image was matched but it is impossible to find the homography

of said image with the queried one, which prevents the possibility of user's localisation. In order to contour this limitation, a second bundle was created. With the Flann Based Matcher, the matches obtained are correlated between the trained index and the queried image. Furthermore, in our tests the Flann Based Matcher, while using a single image, matched with an average processing time of 5.5 ms. This allows for an initial faster and broad user's localisation within the museum environment, followed by a more specific approach once the localisation is found. It is important to notice that, with our method, even when adding the FI and the posterior FBM processing time, it is still faster comparing to the only use of the FBM.

An additional method was also analysed based on ASIFT. Due to the nature of the application, it is expected that the users explore the superimposed content not only frontal-facing to the walls, but also shifting the smartphone to the side, which creates an image perspective more difficult to match. With the ASIFT algorithm we expected to explore the additional affine matching while using the FLANN index matcher to maintain an acceptable performance with the new additional descriptors. Unfortunately, the obtained results, while successful, returned a large reduction of matched indexes, in some cases more than 13 times less. For this reason, further tests and analyses will be performed and presented on a future publication.

Regarding the templates used to train the FLANN indexes, it was observed while advancing the presented algorithm that the wooden frames of the museum's artworks represented a large part of the retrieved features from the images, as can be seen on Fig. 3.2. When implemented, it was verified that the wooden frames' descriptors matched vastly between themselves across different artworks, which introduced plenty of false positive matches. To prevent this results we applied masks over the templates, as can also be seen on Fig. 3.2, allowing only the features present on the artworks to be computed as descriptors. This improved the performance and reduced the observed false positives. The nature of the shape and form of the templates images will be further explained at the finding homography step. In continuity of our previous work, the height of the templates was limited to 480 pixels, which is the obtained

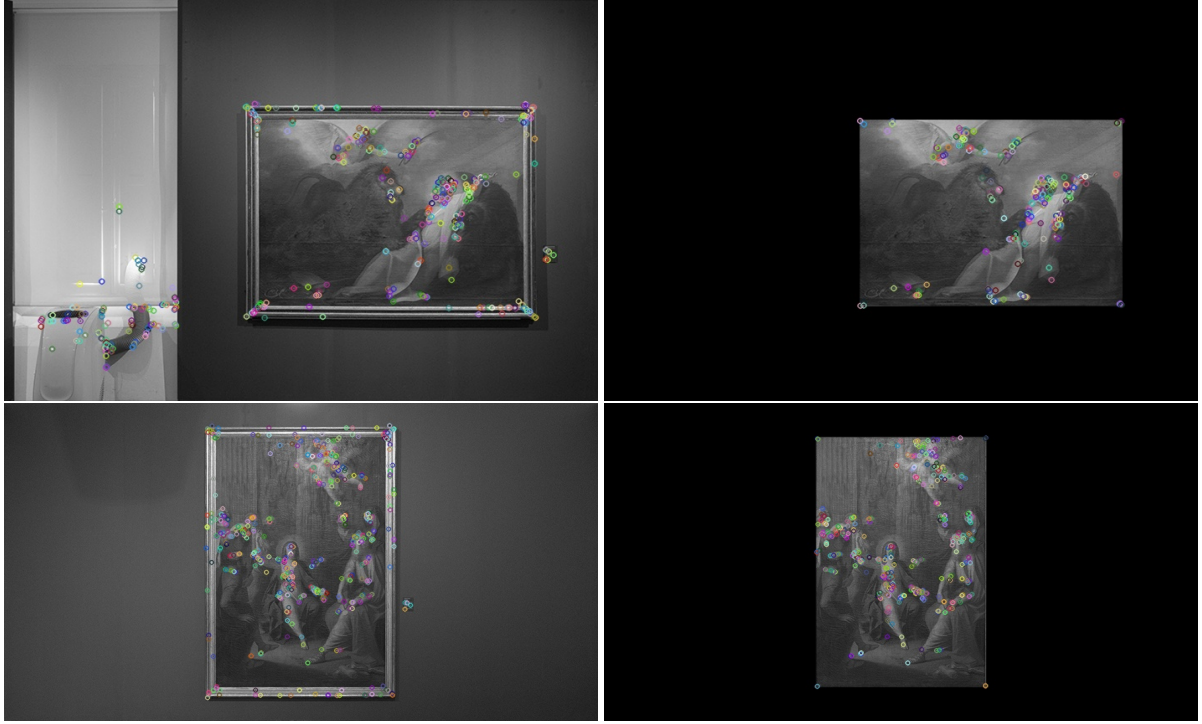


Figure 3.2: Top and Bottom: Example of the different amount of keypoints once the painting's frame is removed through the use of masks.

height of the smartphones' camera frame, and the detector and descriptor remains the BRISK, which allows for some image scaling in the recognition, which is expected to occur while the users navigate the museum.

Advancing to the recognition and localization stage (b) of the algorithm, while retrieving the frames from the user's smartphone camera, if there isn't a previous matched frame, the FLANN Index is used to find the corresponding image's index. As the FI usually returns the more approached image to the frame's descriptors, it is always necessary to perform at least the corresponding FLANN Based Matcher of the obtained image index to discard the insufficient matches. It was observed that the amount of returned matches from the FI is not correlated to the certainty of the retrieved match. Nevertheless, this method continues to be faster than a plain FBM use. All the matches obtained from the FBM are subjected to the Lowe's ratio test, where only the matches with distances inferior to a relation of 0.65 are considered good matches. When a match is found with at least 10 good matches, then we proceed to find the homography. Normally only 4 matches are needed for the homography calculation, but as mentioned on previous work, for this AR application it is mandatory

the computation of a good homography, and therefore the number of minimal good matches was increased. The following homography refinement method can be found on our previous publication, with the addition of a symmetry test and also a verification if the return matrix isn't transposed, being this way possible to salvage some bad outputs (Tolias and Avrithis (2011)). Having in mind the necessity of a considerable amount of good matches but also a smooth performance, the amount of descriptors detected from the frame is directly associated with the previous frame processing time, with all being firstly sorted by their response parameter.

As referred before, the templates' shape form was made with a purpose. When calculating the homography we found the perspective relation between two different planes: an image in the 2D world, and an object in the 3D world. Due to the similar construction structure between different smartphones, we were able to observe a limited variation in the intrinsic camera matrix, which allow us to assume a acceptable outcome within error, if needed we might implement an auto-calibration method as future work (Mendonça and Cipolla (1999)). In order to reduce additional computing calculations, when the template's images where obtained, the corresponding wall height was included, which allows for a direct relation between the artworks full of features and the plain walls that lack them. It is important to refer that a panoramic of all the walls of a museum room was also created, but giving the vastness of the information, the panoramic reduced the performance and increased the false positives matches, therefore increasing the amount of bad homographies computed. The current arrangement of templates cover completely the walls of the museum room with at least two artworks always present, with the exception of large artwork pieces. Using these limits, and with each template preceding the other and never overlaying, it is possible to retrieve the already known shape of the room without the need of advanced 3D calculations.

With the homography known, the next steps are the fusion of the previous two methods presented on former publications. A Gaussian Blur is applied to the obtained frame. A dynamically adjusted Canny (1986) edge detection is applied to the camera's



Figure 3.3: Example of the lines found in the environment through different perspectives.

frame, using the Otsu (1979) threshold to replace the high Canny's threshold while the lower varies with a direct proportion of 10% to the higher. From there, the Probabilistic Hough (1962) Transform (Kiryati et al. (1991)) is applied in order to retrieve the presented lines in the environment, as can be seen in Fig. 3.3.

The Line Segment Detector was also considered, but it presented a performance 3 times worse for the same amount of lines retrieved. The obtained lines are then filtered with the vertical and horizontal lines being discarded relatively to the horizon line. Afterwards the similar lines are removed, remaining only the unique lines, expected to be the environment vanishing lines. The intersecting points between these lines are calculate following the Cramer's Rule. The obtained intersecting points are added to a k-means clustering, where the most dense cluster is chosen and its centroid is considered as vanishing point. The already found lines from the homography are then adjusted to the obtained lines corresponding to the walls' horizontal delimitations, improving the already refined homography.

Following the last steps comes the stage of tracking (c). As referred in previous publications, it is not expected the possibility of always achieving a valid match with the templates. In order to be able to continue tracking the user's navigation it is necessary to deploy different methods for confirming the user's actions. The direct method is to continue tracking the matching image and the ones surrounding it to the left and right. We also used the retrieved homography perspective to generate a mask which is used while the same image is matched, which allows discarding unnecessary descriptors from the frame. With the fusion of both methods previously presented we are able



Figure 3.4: Left to right: the desired segmentation of the environment’s walls, two examples of superimposing results.

to use a novel approach, where we apply Kalman Filters to the vanishing point and corresponding points of the found and adjusted homography through vanishing lines. This method allows for a better perception of the user’s movement and smooths the transitions of the superimposed content.

Finally, we reach the last stage, the superimposition (d). Although it was already possible after the second stage, we decided it was a higher priority to first start tracking so we could evaluate the initial tracking frames, and after a small amount of good tracked frames, initialising the projection of content over the walls. Considering the purpose of the users’ visit being the museum’s artwork, to be able to superimpose contents while maintaining the artwork visible, the templates’ masks already generated and used while building the bundles, are used here. With all the templates following each other, an example can be seen of Fig. 3.1; we are able to find where the corresponding vanishing line end and another perpendicular wall commences, allowing us to generate a perspective matrix corresponding to the projected wall(s). With this information, we can project specific content on different walls throughout the museum’s

rooms. A desired result is presented on Fig. 3.4.

3.3 Conclusions

The current state of this module shows promising results, presenting the fusion of two different methods previously introduced that allow for a better filtering and also recovery of bad homographies, while introducing an additional geometric tracking method. With the possibility of acquiring mainly good homographies, it is possible to consider the calculations of the user's camera pose on the real world (Elqursh and Elgammal (2011); Bartoli and Sturm (2005); Vincent and Laganière (2001)), which in future work is being considered to be reprojected into a 2D map of the museum and the localisation and direction of the users being computed using Kalman Filters to reject the remaining bad homographies.

The presented form of the templates are considered the final version, with the complete wall height and shape in the templates being used to retrieve the walls' horizontal limits and localisation while also using masks to discard the unwanted features retrieved from the paintings' frames, and being continuous to each other in the real world, allowing the calculation of an accurate perspective matrix in order to superimpose content.

Regarding the search and matching between templates and the camera frame, this was also addressed with the introduction of a mixed FLANN indexes search engine, which has shown excellent time results and allows for a faster and broader localisation while remaining with a more specific matching intended for the AR superimposition method.

For future work, the unexpected occurrence of few returned indexes while adding and training using the ASIFT method to the current algorithm could be further explored and evaluated. Although, with the environment's vanishing lines, it is also possible to continue the tracking into more obtuse view perspectives, as can be seen on Fig. 3.3, which could disprove the necessity of properly implementing the ASIFT

method.

With the expected algorithm fully implemented, a battery of tests shall be produced to evaluate the performance and quality of this module in real-time and introducing additional rooms with different configurations.

4

AR Contents Superimposition on Walls and Persons

Abstract

When it comes to visitors' experiences at museums and heritage attractions, objects speak for themselves. With the aim of enhancing a traditional museum visit, a mobile Augmented Reality (AR) framework was developed during the M5SAR project. This paper presents two modules, the wall and human shape segmentation with AR content superimposition. The first, wall segmentation, is achieved by using a BRISK descriptor and geometric information, having the wall delimited, and the AR contents superposed over the detected wall contours. The second module, person segmenta-

tion, is achieved by using an OpenPose model, which computes the body joints. These joints are then combined with volumes to achieve AR clothes content superimposition. This paper shows the usage of both methods in a real museum environment.

4.1 Introduction

Augmented Reality (AR) (Azuma et al. (2001)) is no longer an emergent technology, thanks mainly to the mobile devices increasing hardware capabilities and new algorithms. As cornerstone, AR empowers a higher level of interaction between the user and real world objects, extending the experience on how the user sees and feels those objects, by creating a new level of edutainment that was not available before. While many mobile applications (App) already regard museums (InformationWeek (2017); TWSJ (2017)), the use of AR in those spaces is much less common, albeit not new, see e.g. Vainstein et al. (2016); Rodrigues et al. (2017); Portales et al. (2010); Gimeno et al. (2017).

The Mobile Image Recognition based Augmented Reality (MIRAR) framework (Pereira et al. (2017)) (developed under M5SAR project (Rodrigues et al. (2017))) focuses on the development of mobile multi-platform AR systems. One of the MIRAR's requirements is to only use the mobile devices RGB cameras to achieve its goals. A framework that integrates our presented goals is completely different from the existing AR software development kits – SDK, frameworks, content management systems, etc.(Artoolkit (2017); Catchoom (2017); Layar (2017)).

This chapter focuses on two particular modules of MIRAR, namely: (a) the recognition of walls, and (b) the segmentation of human shapes. While the first module intends to project AR contents onto the walls (e.g., to project text or media), the second contemplates the overlap of clothes onto persons. The wall detection and recognition is supported upon the same principles of the object's recognition (BRISK descriptor) but uses images from the environment to achieve it. On the other hand, the human detection and segmentation uses Convolutional Neural Networks (CNN) for the detection

(namely, the OpenPose model (Cao et al. (2018))). The overlapping of contents in the museum environment is done over the area limited by the wall or using the body joints along with clothes volumes to put contents over the persons. The main contribution of this paper is the integration of AR contents in walls and persons in real environments.

The chapter is structured as follows. The contextualization and a brief state of the art is presented in Sec. 4.2, followed by the wall segmentation and content overlapping sub-module in Sec. 4.3, and the human shape segmentation and content overlapping in Sec. 4.4. The paper concludes with a final discussion and future work, Sec. 4.5.

4.2 Contextualization and State of the Art

AR image-based markers (Cheng and Tsai (2013)) allow adding in any environment easily detectable pre-set signals (e.g. paintings and statues), and then use computer vision techniques to sense them. In the AR context, there are some image-based commercial and open source SDK and content management systems, such as Catchoom (2017), Artoolkit (2017) or Layar (2017). Each of the above solutions has pros and cons and, to the best of your knowledge, none has implemented wall and person segmentation with information overlapping.

The ability of segmenting the planar surfaces of any environment continues to be a challenge in computer vision, mainly if only a monocular camera is used. One of the directest approach to an environment's scanning is the use of RGB-D cameras (Gupta et al. (2015)) or LiDaR devices (Hulik et al. (2014)) to directly acquire a 3D scan of the cameras' reach. A more indirect approach – more based on computation than hardware – is the Simultaneous Localization and Asynchronous Mapping (SLAM) (Durrant-Whyte and Bailey (2006)). SLAM's methods for indoor and outdoor navigation has shown new advances either by using the Direct Sparse Odometry (Engel et al. (2018)), or with a feature matching method like the ORB SLAM (Mur-Artal et al. (2015)) or even a Semi-Dense (Engel et al. (2013)) or Large-Scale Direct Monocular SLAM (Engel et al. (2014)).

Another usual approach is the cloud of points method or the structure from motion, which is part of the SLAM's universe, relying on multiple frames to be able to calculate a relation in-between the features – 3D points – and the camera's position. There have been developments in the outdoor, or landmark, recognition (Babahajiani et al. (2014)), an also simple objects detection and its layout prediction using the cloud of oriented gradients (Ren and Sudderth (2016)). Another example, proving the possibilities of a proper environment's layout analysis, is the use of a structure from motion algorithm using the natural straight lines in an environment, through representation, triangulation and bundle adjustment (Bartoli and Sturm (2005)).

One of the main novelties is the use of CNN to solve any complex computer vision challenge, including environment's layout prediction (Tateno et al. (2017)), although the current state is not useful in runtime. On the other hand, in every common human-based construction there can be found the presence of lines or edges in its geometric perspective. These vanishing lines allows us to predict the orientation and position of planes (Haines and Calway (2012)). It is even possible to compute a relative pose estimation using the present lines in the environment (Elqursh and Elgammal (2011)). These techniques, applied to the indoor layouts' prediction, allows us to compute the existence of natural planar surfaces (Serrão et al. (2015)), even by using the edges of maps available on any indoor layout (Mallya and Lazebnik (2015)). One major advance in the outdoor camera localization is the PoseNet (Kendall et al. (2015)), which also uses a CNN. It is important to stress that none of those methods presents the superimposing of contents over an environment know *a priori*, on a monocular mobile device and in runtime.

The second module to be presented in this work focuses on human segmentation and pose estimation, which is also a challenging problem due to several factors, such as body parts occlusions, different viewpoints, or human motion (Fang et al. (2017)). In the majority of models based on monocular cameras, the estimation of occluded limbs is not reliable. Nevertheless, good results for a single person's pose estimation can be achieved (Fang et al. (2017)). Conversely, pose estimation for multiple people

is a more difficult task because humans occlude and interact between them. To deal with this task, two types of approaches are commonly used: (a) top-down approach (He et al. (2017)), where a human detector is used to find each person and then running the pose estimation on every detection. However, top-down approach does not work if the detector fails to detect a person, or if a limb from other people appears in a single person's bounding box. Moreover, the runtime needed for these approaches is affected by the number of people in the image, i.e., more people means greater computational cost. (b) The bottom-up approach (Cao et al. (2017); Fang et al. (2017)) estimates human poses individually using pixel information. The bottom-up approach can solve both problems cited above: the information from the entire picture can distinguish between the people's body parts, and the efficiency is maintained even as the number of persons in the image increases.

As in the wall detection, the best results for pose estimation are achieved using R-CNN (Regions - CNN) (Girshick et al. (2014)) or evolutions, such as the Fast R-CNN (Girshick (2015)), Faster R-CNN (Ren et al. (2015)) or the Single Shot MultiBox Detector (SSD) (Huang et al. (2017)). A comparison between those methods can be found in Huang et al. (2017). The results show that SSD has the highest mAP (mean average precision) and speed. With good results, OpenPose (Cao et al. (2017)) can also be used for pose estimation, being based on Part Affinity Fields (PAFs) and confidence maps (or heatmaps). The method's overall process can be divided in two steps: estimate the body parts (ankles, shoulders, etc.) and connect body parts to form limbs that result in a pose. In more detail, the method takes an input image and then it simultaneously infers heatmaps and PAFs. Next, a bipartite matching algorithm is used to associate body parts and, at last, the body parts are grouped to form poses. The OpenPose can be used with a monocular camera and run in "real-time" on mobile devices. Additionally, the estimated 2D poses can be used to predict 3D poses using a "lifting" system, that does not need additional cameras (Tome et al. (2017)).

Several methods exist for clothes overlapping. A popular one is Virtual Fitting Room (VFR) (Erra et al. (2018)), which combines AR technologies with depth and

colour data in order to provide strong body recognition functionality and effectively address the clothes overlapping process. Most of these VFR applications overlap 3D models or pictures of a clothing within the live video feed and then track the movements of the user. In the past, markers were used to capture the person (Araki and Muraoka (2008)). In that case, specific joints are used to place the markers, which differ in colours according to the actual position on the body. From a consumer's point of view, a general disadvantage is the time consumed placing the markers and the discomfort of using them. Isikdogan and Kara (2012) use the distance between the Kinect sensor and the user to scale a 2D model over the detected person, only depicting the treatment of t-shirts. Another similar approach, presented in Erra et al. (2018), uses 3D clothing with skeleton animation. Two examples of the several commercial applications are Facecake (2016) and Fitnect (Kft. (2016)).

4.3 Wall Detection and Information Overlapping

Previously, the authors followed two distinct approaches to solve the environments' surfaces detection (Pereira et al. (2017); Rodrigues et al. (2018); Veiga et al. (2017, 2018)). A first approach assumes that the vanishing lines present in the environment follow an expected geometric shape; and a second approach focuses on retrieving the walls' proportions using the features extraction and matching method, followed by the homographies' computation. The methods were then combined in order to achieve a harmonious detection, recognition and localization of the environment, allowing to dynamically superimpose different types of content over the walls, such as images, video, animations, or 3D objects.

As detailed next, the present algorithm is designed to work over regular plane walls, which are known *a priori* through a previously bundle creation phase. Being the purpose of this AR application the ability to run seamlessly on any current monocular smartphones, from which only a RGB image is provided by the camera (i.e., without any additional depth information), it is important to assure an ideal performance using

less computational' eager algorithms.

Our current algorithm divides itself in five different stages: (a) the bundle creation, (b) the recognition and localization, (c) corners' adjustment, (d) tracking, and (e) superimposition.

The first stage of the algorithm – the bundle creation (a) – is pre-executed, i.e., not performed during runtime. For this task two distinct types of bundles are generated: a FLANN (Fast Library for Approximate Nearest Neighbours) (Muja and Lowe (2012)) Index (FI) bundle, and a FLANN Based Matcher (FBM) bundle. This odd combination is due to a better performance being obtained by a hybrid version of both FLANN matchers instead of only one, as presented in Veiga et al. (2018). Reasons for this choices will be better detailed during the recognition and localization (b) phase.

Museums' environments are full of detail and some of its areas gather enough significant information to be considered keypoints, which can be detected and define by computing its descriptors. In this approach, the BRISK keypoint detector and descriptor extractor (Leutenegger et al. (2011)) is used, due to its capabilities of performing well with image scaling. Images of continuous walls, as can be seen in Fig. 4.1 top two rows, allow not only to project content, but also retrieve the users' localization through the sparse unique keypoints inside the artworks. The retrieved features are stored during the bundle creation, allowing the comparison during runtime with the ones obtained from the smartphones' cameras.

As observed in Veiga et al. (2018), the paintings' wooden frames are rich in similar features, which often would lead to cross-matched in between them. To prevent this false matches, the templates are pre-processed before training the FLANN indexes, defining masks where only the features from the artworks could be obtained, as it can be seen on Fig. 4.1 bottom row. Additional final templates examples can be observe on Fig. 4.2. The motive behind the shape and form of the templates will be explained in detail during the next phases.

Although FBM is built upon FI, previous performance tests showed that the bare FI returns results similar to the ones obtained with FBM, but with an average of 60.66%

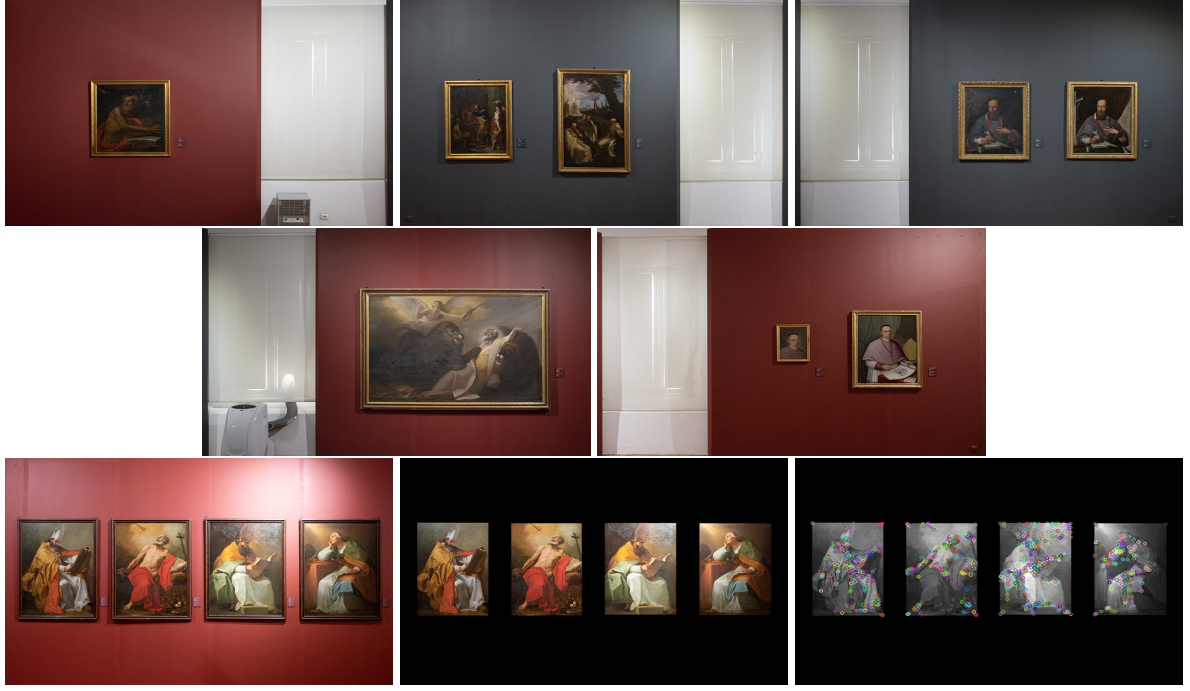


Figure 4.1: Top two rows, from top left to bottom right: Example of five templates following of the same wall. Bottom row, pre-processing of the templates. Left to right, input image with the complete desired height of the wall, mask applied over removing the wooden frames, features retrieved and computed.

less processing time (Veiga et al. (2018)), which justifies the choice of building an FI bundle. While both methods retrieve the same template index, the FBM also retrieves the matching between features, which is essential for the computation of the homography. Following this necessity, a bundle is created for each matching method, which allows to generate a hybrid FLANN matching method. This method, starts by searching across our templates with the FI bundle and then only process the top retrieved results with FBM, which was proved to be a faster matching method, when compared to using exclusively FBM (Veiga et al. (2018)).

Both methods – FI and FBM – used the same index parameters, and the same searching algorithm, the Locality-Sensitive Hashing (LSH), which performs extremely well with non-patent binary descriptors. The LSH used a single hash table with a key size of 12, and only 1 multiprobe level. The addition of a multiprobe to the LSH, allows to reduce the number of hash tables, obtaining a better computational performance without affecting precision. As presented in Veiga et al. (2018), it was noted an average reduction of 76.56% of processing time across different binary features de-

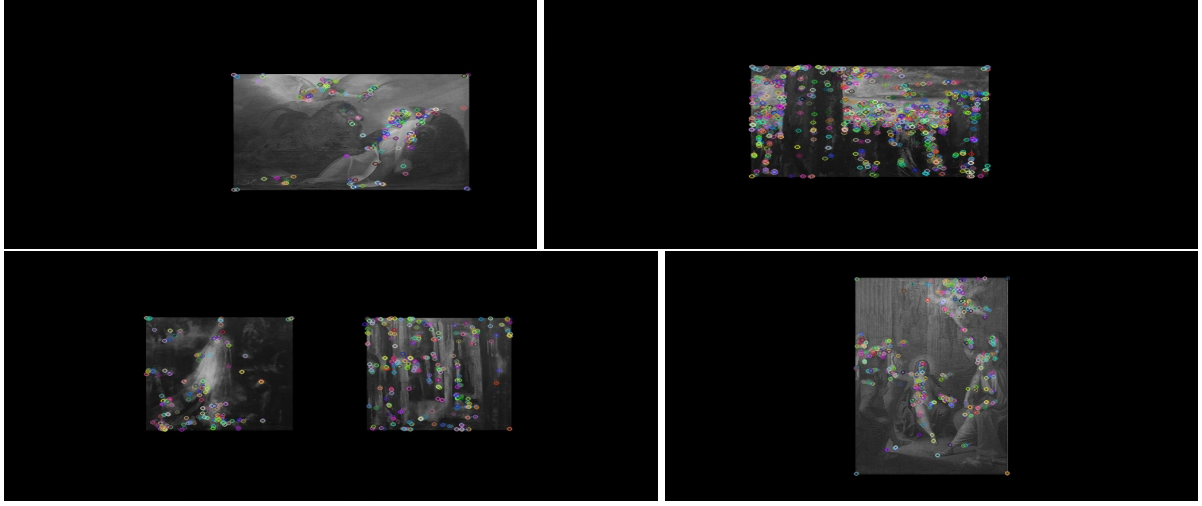


Figure 4.2: Example of some of the templates used during the bundle creation stage.

tectors and descriptors (AKAZE, BRISK, ORB) (Tareen and Saleem (2018)) while using only 1 hash table, versus the 6 hash tables originally recommended.

The runtime computation starts with the recognition and localization stage (b). While no localization information or previous match is available, the retrieved frame from the camera is resized to a resolution of 640×480 pixels (px), and processed with the BRISK feature detector through the FI feature matcher, returning a list of probabilities for the index of each template, as can be seen on the top-left and top-centre of Fig. 4.3. Similar to the top-5 rank in CNN, the image with highest probability is not occasionally matched, although one within the top-5 is used. Then the FBM is applied through the top-5 indexes and the results are subjected to the Lowe's ratio test, where only the matches with distances to each other with a relation between 55% and 80% are considered. If at least 20 of these matches are obtained, then the algorithm continues, otherwise it skips this frame's processing. It is also important to stress that in order to achieve a near real-time performance, the previous frame's processing time is correlated with the total amount of descriptors for the current frame, with all being firstly sorted by their response parameter, which correlates the level of similarity between the templates and frames' descriptors.

With the computation of the homography's matrix between the correlated matches of the template and the camera's frame, the perspective transformation of the 2D template can be computed as an object within the 3D world, which can be observe in

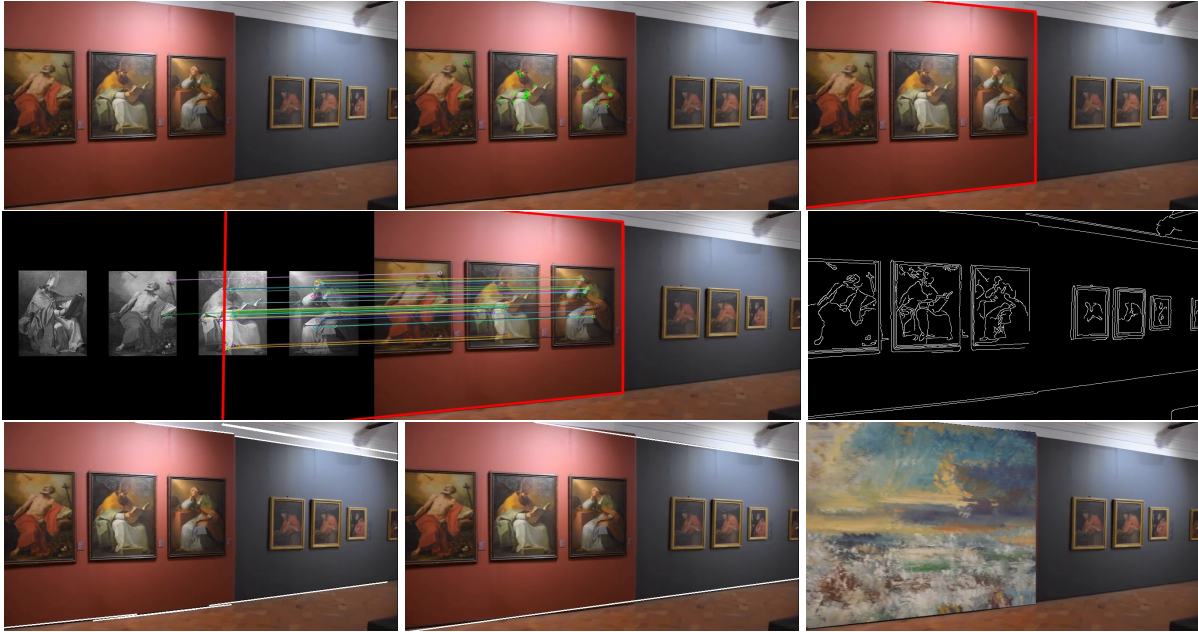


Figure 4.3: Pipeline of the environments' superimposition algorithm. Left to right, top to bottom: input frame, keypoints and descriptor computation, homography's calculation, demonstration of the relation between matches and the homography, Canny edge detection, Probabilistic Hough Transform, vanishing lines post-processing, content superimposed.

Fig. 4.3 – top-right. Normally, the homography only requires 4 matches to be able to calculate but, with the user navigating through out the museum, steady results for this AR application were obtained only when the minimum limit of matches was increased to 20 points. We also discard the bad homographies verifying if the computed matrix presents a possible solution which could match our desired output: direction, proportion, and perspective. A demonstration of this process can be seen in Fig. 4.3 – centre-left and right.

During the bundle creation stage (a) the templates' shape form where made for a specific purpose: the ability to find the upper and bottom margins of any specific wall, as well the left and right limits when necessary. The current arrangement of templates is divided between two rooms, one regular – cuboid – and one irregular. The aim for the regular room is to be able to localize the exact position and angle that the user is pointing. Furthermore, using the continuous templates from the same wall, as shown in Fig. 4.1 – top two rows, an automated mixed 3D layout of the museum's room is being designed, with the objetive of further exploring the AR applications without the

need for advanced 3D calculations. In the irregular room, the walls are used to project any desired content, e.g., a video-documentary related to the artwork exposed on that specific wall without the ability to project over the entire environment's layout.

With the homography already known, the next step is the corner's adjustment stage (c), which is the result of the combination of several methods (Pereira et al. (2017); Rodrigues et al. (2018); Veiga et al. (2017, 2018)). The frame's edges are computed by applying a Gaussian filter to blur the frame, followed by a dynamic Canny (1986) edge detection using the Otsu (1979) threshold to replace the high Canny's threshold, which decides if a pixel is accepted as an edge, while the lower threshold, which decides if a pixel is rejected, varies with a direct proportion of 10% to the higher. The computed edges can be seen on the centre-right of Fig. 4.3. Afterwards, the Probabilistic Hough Transform (Kiryati et al. (1991)) is applied in order to retrieve the lines present in the frame, as seen in the bottom-left of Fig. 4.3.

Next, the obtained lines are filtered by discarding the extremely uneven lines in relation to the horizon line, followed by the calculation of the similar ones, resulting only in the expected environment's vanishing lines. The lines' intersecting points were clustered using a *K*-means clustering method, where the densest cluster is chosen, and its centroid is considered as the vanishing point of said lines. Considering the original location of the homography's corner points, with the known vanishing point, these corners can be adjusted to existing lines in the environment – upper and lower limit of the wall –, as observed in the bottom-centre of the Fig. 4.3.

Previously, the application of Kalman filters to the vanishing point and its corresponding corner's coordinates was introduced in Veiga et al. (2018), allowing for a better perception of the user's movement, and consequently smoothing the transitions of the superimposed content. Although the current state of the present algorithm retains this step, Kalman filters are no longer used for tracking, with its main function being the validation of a proper template's perspective found on the processed frames. More precisely, the Kalman filtering of the coordinates allows to predict their next position and estimate if the ones retrieved behaved as noise or valid inputs. This favors

the obtention of more precise coordinates in time with more harmonious trajectories – it is important to refer that the obtained homographies are not perfect and their perspective fluctuates significantly, which leads to noisy coordinates. This probably is due to the recursiveness of the Kalman filters but, there was only the need to adjust the uncertainty matrix to our specific application and no additional past information is required to be able to process in real time. Before advancing to the last two stages of the algorithm, the previous steps are computed again using a mask retrieved from the calculated coordinates. When the Kalman filters stabilizes, the process proceeds to the next stage.

Regarding the tracking stage (d), with the corresponding template's coordinates found, the good features to track within our current frame's mask are computed, using the Shi and Tomasi (1993) method. Afterwards, the optical flow between the previous and the current frame is calculated using the iterative Lucas-Kanade method with pyramids (Bouguet (2001)). Using this method, a more accurate homography between frames can be computed, which results in a more fluid and smooth tracking using even less computation than our previous approach. It should be noticed some important aspect of this approach such as the fact that the smartphones' cameras are different between brands and models, sometimes even between the operating system versions, which results in different features match across the devices. Through this method, a lighter computational tracking in any device and in multiple conditions was possible. The Shi-Tomasi corners continues to be obtained through the tracking, which enables the visitor to continue walking through the museum without the AR experience – which enables the visitor to explore the content in a higher detail.

Following the previous stage, the superimposition stage (e) can finally be processed. With the improved tracking stage, the overlay of content over the environments' previous known walls, allowing the visitors' movement, is possible, without affecting the projected content. The result can be seen in the bottom-right of Fig. 4.3. Although, it is only presented the projection of content over the corresponding template's shape, it is also possible to use the template's mask and re-purpose the artwork's surrounding

empty walls with content without covering the artwork. With the different templates, specific content can be projected on different walls throughout the museum's divisions.

4.4 Person Detection and Clothes Overlapping

As mention, the goal of the Person Detection and Clothes Overlapping module is to use a mobile device to project AR content (clothes) over persons that are in a museum. On other words, the goal is "to dress" museums' users with clothes from the epoch of the museums' objects. The module has two main steps: (i) the person detection and pose estimation, and the (ii) clothes overlapping. Those steps will be explained in detail in the following sections.

The implementation was done in Unity (2018) using the OpenCV library (Asset for Unity). In order to verify the implementation's reliability, computational tests were done in a desktop computer and in a mobile device, namely using a Windows 10 desktop with an Intel i7-6700 running at 3.40 GHz and an ASUS Zenpad 3S 10" tablet.

The method used for pose estimation was OpenPose (see Sec. 4.2 and Cao et al. (2017); Kim (2018)). OpenPose was implemented on TensorFlow (Google (2018)) and the CNN architecture for feature extraction is MobileNets (Howard et al. (2017b)). The extracted features serve as input for the OpenPose algorithm, that produces confidence maps (or heatmaps) and PAFs maps which are concatenated. The concatenation consists of 57 parts: 18 keypoint confidence maps plus 1 background, and 38 ($= 19 \times 2$) PAFs. The component *joint/body part* of the body, e.g., the right knee, the right hip, or the left shoulder, are shown in Fig. 4.4, where red and blue circles indicate the person's left and right body parts. A pair of connected parts, *limb*, e.g., the right shoulder connection with the neck are shown in the same figure, the green line segments.

A total amount of 90 frames of expected user navigation were the input to the CNN. Furthermore, two input sizes images for the CNN were tested: 368×368 and 192×192 px. Depending on the size of the input, the average process time for each frame was 236ms/2031ms (milliseconds) and 70ms/588ms, respectively in the desktop and tablet.

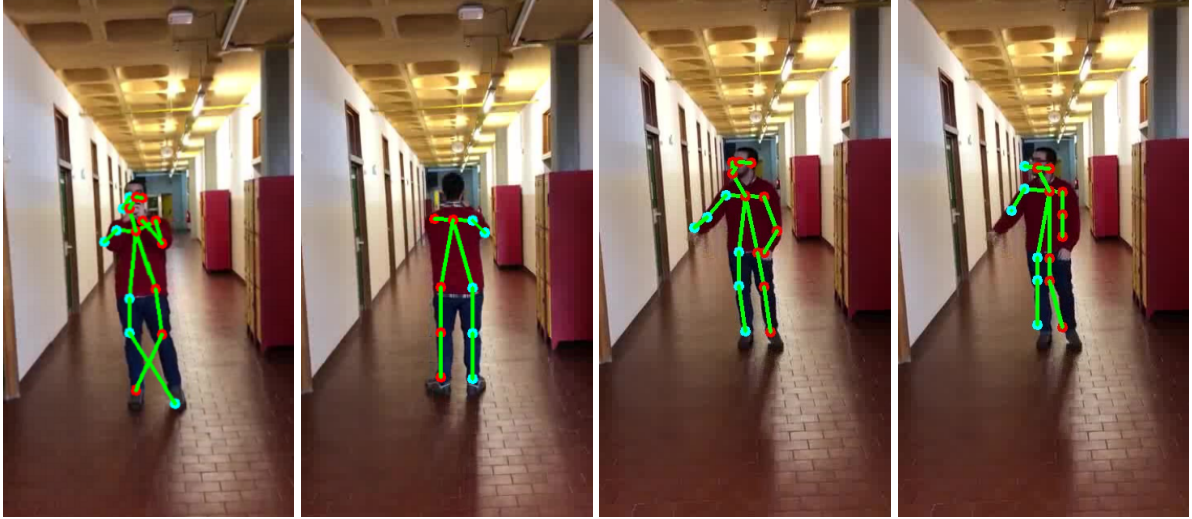


Figure 4.4: Left to right, example of confusion between left and right ankle, the correct detected pose, and the pose estimation with spatial size of the CNN equal to 368×368 px and 192×192 px.

As expected, reducing the input size images of the CNN allow attaining improvements on the execution time, but the accuracy of the results dropped. The pose is always estimated, but the confidence map for a body part to be considered valid must be above 25% of the maximum value estimated in the confidence map (this value was empirically chosen), otherwise is not considered. A missing body part example for a 192×192 px image which was detected in the 368×368 px image is shown in Fig. 4.4, right most image. The same figure also shows an example of error that sometimes occurs in the identification of the right and left hands/legs (left most image).

Besides the presented cases, a stabilization method was needed because pose estimated (body part) can wrongly “change” position, for instance due to light changes. The stabilization is done using groups of body parts from the estimated pose. The body parts selection for each group is based on the change that body parts do when any single one moves, see Fig. 4.5.

The stabilization algorithm is as follows: (a) for each one of the 5 groups present in Fig. 4.5, a group of RoIs (one for each body part), with 2% of the width and height of the frame (value chosen empirically), is used to validate if all the body parts from the group have changed position or not. (b) To allow a body part to change position, all the other group body parts must change, i.e., they must have a position change bigger

Groups	1st	2nd	3rd	4th	5th
Parts	Neck	Right Hip	Left Hip	Right Elbow	Left Elbow
	Right Shoulder	Right Ankle	Left Ankle	Right Wrist	Left Wrist
	Left Shoulder	Right Knee	Left Knee		

Figure 4.5: Pose estimation stabilization groups.



Figure 4.6: Examples of volume 2D views.

than the RoIs mentioned before. (c) Depending of the group, if one or two body part(s) have a value bigger than the predefined RoIs, this wrong body part(s) is/are replaced by the correct ones, that was/were estimated in a previous frame.

To solve the incorrect detection of the body parts problem, the estimated pose view is used, i.e., to distinguish between right and left body parts it is necessary to validate if the body is in a front or in a back view. (d) This is done by observing that in a front view, the x coordinates of the right side body parts should be smaller than the ones from the left side. To replace a missing body part from a pose is used the previously estimated pose.

In the second phase, the clothes overlapping methods has as input the estimated body parts. For clothes overlapping, three methods were tested: (i) segments, (ii) textures, and (iii) volumes. The first two methods were presented in Bajireanu et al. (2018), showing some lack precision and the limitation of only working in frontal view.

For the third method (volumes), the two main steps are: (a) rotate and resize the volume, (b) project the (clothes) volume over the person.

In the first step, (a.1) a clothe volume was developed in 3DS MAX (2018) and (a.2) imported to Unity. Then, (a.3) the volume was rotated horizontally accordingly to four

		Body Parts				
		Nose	Right eye	Left eye	Right Ear	Left Ear
Views	Front	1	1	1	1	1
		1	1	1	0	0
		1	1	1	0	1
		1	1	1	1	0
	Back	0	0	0	0	0
		0	0	0	1	1
		0	0	0	1	0
		0	0	0	0	1
	Right Side	1	1	0	1	0
	Left Side	1	0	1	0	1

Figure 4.7: Created views conditions represented horizontally. A detected part is represented by 1 and not detected by 0.

pose views, as presented in Fig. 4.6 where frontal, back, side right, and side left views of the volume can be seen. (a.4) The views were then associated to the OpenPose detected and non detected body parts (namely: nose, right eye, left eye, right ear and left ear) according with the conditions presented in Fig. 4.7, where 1 represents a detected body part, and 0 a non detected body part. Additionally, (a.5) to strengthen the assurance of front or back view, the x coordinates distance between right and left hips and shoulders coordinates should be more than 5% of the frame width (this value was empirically chosen). (a.6) A previous view is used if none of the above conditions are met. Finally, (a.7) the volume is resized using the distance between ankles and neck which is an approximation to the person's height.

The resized volume is now project over the detected person (b). To achieve the referred projection, the volume body parts keypoints (see Fig. 4.8 left) are (b.1) overlapped over the estimated OpenPose pose body part keypoints, and (b.2) rotated accordingly to the angle (α_i) between a vertical alignment and each OpenPose's i -limb, see Fig. 4.8 right.

Figure 4.9 shown results of the overlapped volume in a museum environment. The overlapping volumes over a person takes an average processing time of 6.1ms/31.4ms for the desktop and mobile respectively. In general, the overall process takes a mean time of 76.1ms (70ms + 6.1ms) and 590.4ms (559ms + 31.4ms) for the desktop and mobile.

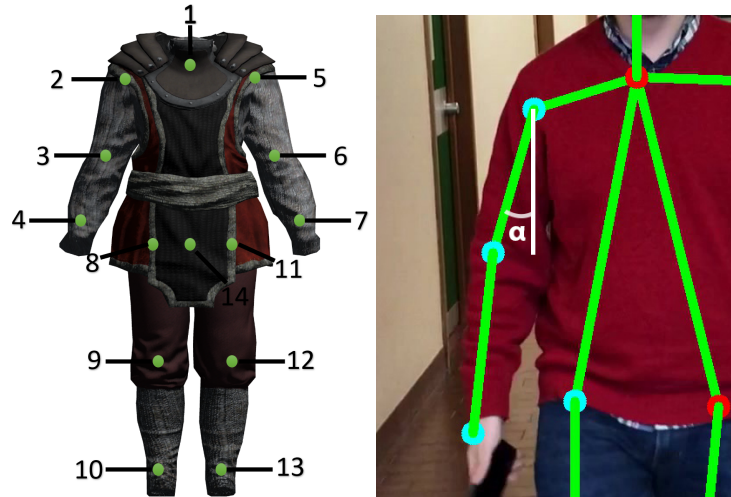


Figure 4.8: Left, volume keypoints. Right, example of a limb's angle.



Figure 4.9: Examples of human shape superimposition using "volumes".

4.5 Conclusions

This chapter presents two modules to be integrated in MIRAR framework (Pereira et al. (2017)), namely: the wall segmentation with overlapping of information and the hu-

man shapes segmentation with clothes overlapping. Furthermore, the modules were integrated as it can be seen in the examples in Fig. 4.10.

Regarding the walls' detection and information overlapping, the current results present a functional and fluid experience of content superimposition even with visitors' movements or with acute angles between the camera's position and the superimposed walls. Nevertheless, further tests in different conditions and new environments' implementations are required to improve and evolve the present algorithm into a more broad and stable performance.

For human clothes overlapping in real involvements (museum in this case), the proposed method combines OpenPose body parts detection with volumes overlapping. For better pose estimation accuracy in mobile devices, a stabilization method and the pose views were created. For real-time performances on mobile devices an OpenPose model with a MobileNet architecture was used and two input image sizes were tested (namely, 368×368 and 192×192 px). The smallest size is the best option for mobile devices in term of execution time, but it is worse in term of accuracy, nevertheless is a good trade-off for the application.

For future work, a faster and more accurate performance with OpenPose could be achieved by testing new network architectures, new training strategies and other datasets. Another way to get better pose estimation results could be achieved by testing models like PersonLab (Papandreou et al. (2018)) or others. For this specific module, other way to do pose view estimation is to train a model to do body / foot keypoints estimation and use the foot keypoints position to know the pose view. Additionally, to predict 3D poses by using the estimated 2D poses, the "lifting" system implementation could be done. In the case of the indoor localization through only computer vision is still not resolved, with the necessity of creating a new compatible method to our present tracking system. There is also a need to develop a mixed 3D layout of the regular museums' rooms in order to be able to totally replace the environment if needed. This would also allow, especially with the seamless tracking, the possibility of superimposing advanced 3D models contents that could offer better information,



Figure 4.10: Examples of both modules working together.

orientation or navigation through the user's visit, fully immersing the visitor in this new era museums' experience.

5

Efficient Small-Scale Network for Room Spatial Layout Estimation

Abstract

In this paper, we focus on the challenging task of retrieving the spatial layout of different cluttered indoor scenes from monocular images, using a smaller deep neural network than the existing proposals. Older geometric solutions are prone to failure in the presence of cluttered scenes because they depend strongly on hand-engineering features and the expectation of the possibility of the vanishing points' calculation. With the growth of neural networks, the geometric methods were either replaced or fused within the emerging area of deep learning. The more recent solutions rely on dense

neural networks with additional adjustments, either by the calculation of the vanishing points, position based, or by layout ranking. All these methods presented valid solutions to this challenge with the flaw of being computationally demanding. We present a more lightweight solution, running the segmentation on a smaller neural network and introducing a discriminative classifier for the posterior layout ranking and optimization. Our proposed method is evaluated by two standard dataset benchmarks, achieving near state of the art results even with a fraction of the required parameters than the available state of the art methods.

5.1 Introduction

The conventional geometrical form present on the vast majority of our day-to-day indoor environments is similar to a 3D box or cuboid. Our main task, presented in this paper, is the delimitation and segmentation of an indoor environment’s limits: walls, floor and ceiling, into corners and edges using only a single monocular image, as shown in Fig. 5.1.

When we are inside any room, our viewpoint orientation and position defines the captured spatial layout, which carries three-dimensional information about the indoor scene. This is a challenge for monocular images due to the lack of a depth channel. These parametric rooms, where we consider the planes’ boundaries to be perpendicular between each other, is normally referred as the Manhattan assumption (Coughlan and Yuille (2001)). Although it seems trivial for our eyes to find any common room’s layout, it is a challenging task in the field of computer vision to estimate its boundaries. The complexity of this case increases exponentially when the indoor scene is cluttered with furniture and/or other objects.

Most of the challenges for monocular indoor layout estimation are not related to the rooms’ architecture but to the additional amount of information distributed through the image. Although, in a cuboid room its boundaries converge to three mutually orthogonal vanishing points (Rother (2002)), which can be found by finding the ‘Manhat-

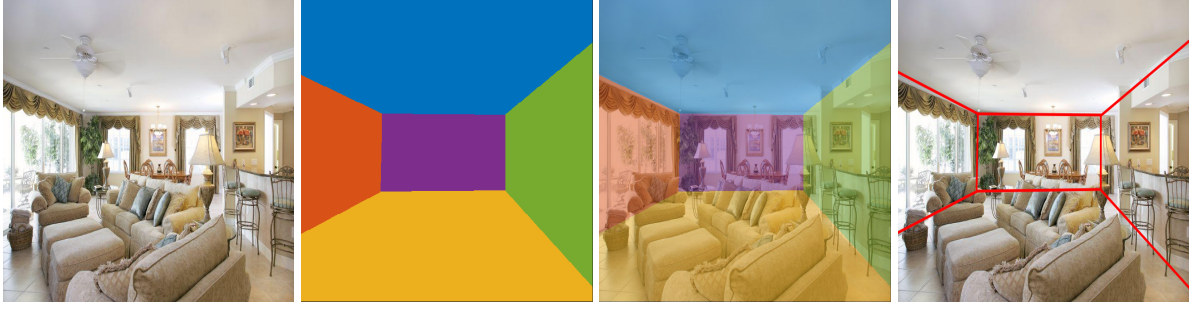


Figure 5.1: LSUN dataset example. Left to right: input image, segmentation map, superimposed segmentation map, superimposed edge map.

tan lines’ (Ramalingam and Brand (2013)) and their intersections points, indoor scenes are usually filled with additional expected or unexpected objects, whose nature and shape can influence the retrieval of the present environment’s lines and corners. Due to the aleatory distribution of the clustered objects, we know that in most rooms, their boundary could be partially or completely occluded. Nonetheless, the same tridimensional shapes are present on most man-made constructions, which allows us to anticipate the presence of edges across the planes’ intersections. Prior work, until the introduction of fully convolutional networks (FCN) to achieve pixelwise labelling (Mallya and Lazebnik (2015)), focused on obtaining the room’s geometry through similar methods based on the location of the vanishing points.

Recent methods focus on the use of FCN or deep convolutional neural networks (DCNN), with encoder-decoder, iterative or double refinement architectures, either for the retrieval of edges, or segmentation of the present layout, room’s points and type, or points and edges. There is normally present a posteriori refinement processing over the convolutional network result, which adjusts the layout reconstruction based on layout ranking, or position based. All the existing state-of-the-art methods already present significant results comparing with the ground-truth, either by corner or pixel error. Our presented approach introduces a novel implementation of the room spatial layout estimation, which replaces the previous heavy backbones of VGG16, ResNet50 ResNet101 for a more small-scale network aimed for mobile use, the MobileNetv2 for backbone network plus DeepLabV3 for semantic segmentation model, followed by a discriminative classifier and a sliding window for layout ranking and refinement. We

used an input image resolution of 224×224 , which is the default for MobileNetV2 and also, coincidentally, the same size used on Zhang et al. (2019), one of the top three state-of-the-art methods.

The spatial layout estimation is an important task which allows us to predict an indoor environment’s geometric limits, which as applications in the fields of augmented reality, indoor modelling (Xiao and Furukawa (2014); Martin-Brualla et al. (2014); Liu et al. (2015)), indoor navigation (Boniardi et al. (2019a); Xu et al. (2014)), robotics (Boniardi et al. (2019b)), virtual reality, and visual cognition (Hedau et al. (2010); Qiao et al. (2015)). The proposed method is an important step towards a cloudless indoor layout estimation due to its lightweight performance, aimed to edge or mobile devices implementations, which would benefit the human and robot interaction with our surroundings.

The chapter is structured as follows. Related works and state of the art are reviewed in Sec. 5.2, followed by the description of the proposed method in Sec. 5.3. Experimental results are presented in Sec. 5.4, and its applications in Sec. 5.5. The concluding remarks and future work are drawn in Sec. 5.6.

5.2 Related Work

The main turning point for the indoor layout estimation was its reformulation as a structuring learning problem, firstly introduced by Hedau et al. (2009), which also presented the first benchmark dataset for this kind of estimation. The idea to approximate a cuboid to a three-dimensional indoor scene was also presented and is derived from the Manhattan assumption (Ramalingam et al. (2013)). Pixelwise geometric labels were also introduced: *left wall*, *middle wall*, *right wall*, *floor*, *ceiling*, *object*. The authors adapted techniques from Hoiem et al. (2007) and divided their methodology in two stages. First, a large number of layout hypotheses are generated by multiple rays from three vanishing points (Rother (2002)), which were obtained by the Manhattan Lines (Ramalingam and Brand (2013)). Some of these hypothesis presented low accuracy

due to the amount of clutter and were impossible to be recovered on the second stage. Afterwards, every hypothesis was scored and ranked using a structured regressor and the features from the labels to find the fitting cuboid with the highest ranking.

A similar layout hypothesis ranking was presented on Lee et al. (2009), which associated the layouts to orientation maps generated by line segments that represented the different regions orientation. Gupta et al. (2010) and Hedau et al. (2010) introduced 3D object reasoning and estimation to refine the structured predictions. Wang et al. (2010) used the indoor clutter to model the room layout. In Del Pero et al. (2012) and Del Pero et al. (2013) Markov Chain Monte Carlo (Gilks et al. (1995)) were used to search for the generative model parameters, while considering both the indoor spatial layout and the 3D cluttered objects. Similarly, Schwing et al. (2012) Schwing et al. (2013), also used the Wang et al. (2010) method, and applied a dense sampling and introduced the integral geometry decomposition method for a efficient structure estimation. Chao et al. (2013) presented a different concept which uses human detections to estimate the vanishing points more accurately, improving highly cluttered indoor scenes 3D interpretation.

The evolution of convolutional neural networks (CNN) and the birth of the FCNs (Long et al. (2015)) started a new age of state of the art achievements in multiple computer vision topics, including semantic segmentation, scene classification and object detection. Mallya and Lazebnik (2015) was the first proposed method exploring the FCNs to predict the informative edge maps, obtaining rough contours that were afterwards added as new features to the layout hypothesis ranking, together with the line membership (Hedau et al. (2009)) and geometric context (Hoiem et al. (2005)). Zhang et al. (2016b) continued this path exploring the deconvolutional networks, with and without fully connected layers. In Dasgupta et al. (2016), instead of learning edges, the authors used FCNs to predict semantic labels. Although the post-processing still relied on the vanishing lines to refine the results, the used previous edge map was replaced by a heat map of semantic surfaces. Ren et al. (2016) proposed a refinement from the coarse result obtained by the a multi-task convolutional neural network (MFCN) (Dai et al. (2016)) combining the layout contours and the surfaces properties.

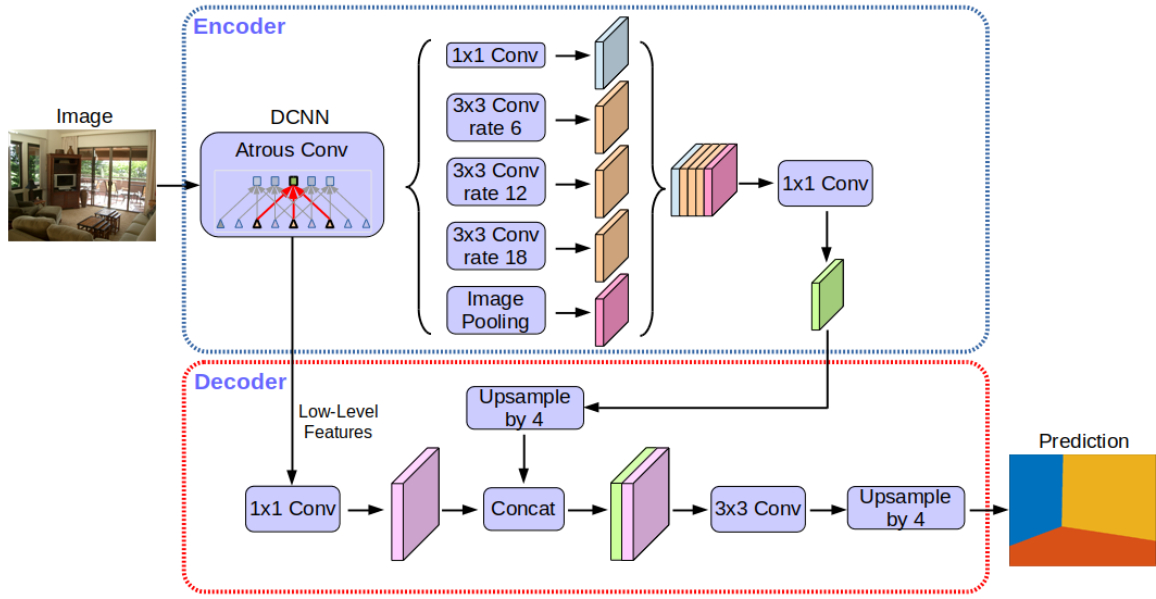


Figure 5.2: DeepLabV3 Encoder-Decoder Architecture (Chen et al. (2018)).

Lee et al. (2017) presented a novel formulation to the indoor layout problem, with an end-to-end network that estimates the locations of the room’s layout keypoints instead of its edges. Zhao et al. (2017) also introduced an unique solution estimating the edges through semantic segmentation and proposing a physics inspired optimization scheme. Zou et al. (2018) and Sun et al. (2019) explored retrieving the room spatial layout from single panoramic images, using similar methods to Lee et al. (2017) and also recurrent neural networks (RNN), like the long short-term memory (LSTM) architecture. Lin et al. (2018) proposed the use of a deep fully convolutional neural network with a layout-degeneration method to remove the need for post-processing refinements. The Manhattan assumption (Ramalingam et al. (2013)) is also used in Hsiao et al. (2019), combining a pre-processing of vanishing points to a Resnet50 (He et al. (2016)) to post-process into a flat room layout representation. Zhang et al. (2019) presented a dual decoder network that is fed by the same encoder, obtaining simultaneously and edge and segmentation maps, which are then combined into the scoring function for ranking and refinement of the layout hypothesis estimation.

5.3 Method

5.3.1 Overview

In the context of indoor spatial layout estimation, most research methods proposed are based on the ‘Manhattan World’ assumption (Ramalingam et al. (2013)), where any room in an image contains three orthogonal directions coinciding on three vanishing points (Rother (2002)). Hedau et al. (2009) represented its layout model using the rays from the outside vanishing points and the inside frame vanishing point. Our proposed method doesn’t rely on the vanishing points for training or refinement, but still follows the Manhattan assumption indirectly. An associated application is presented further in Sec. 5.5.

The estimation of a room layout can be divided in three type of maps: edges, keypoints, and segmentation. Each of these heat maps have their advantages and disadvantages, according to the complexity of the input image: amount of clutter, type of room, occlusion of important spatial edges or corners, randomness of unusual objects. All the obtained results are pixelwise, or pixel-independent, and so each pixel of these maps contains the probability of belonging to a class. Edges maps consist of three layers to distinguish between the boundaries edges: wall-wall, wall-floor, wall-ceiling; or they only contain a single layer without differentiating the labels. The keypoints map consists of multiple layers for each keypoint that are scored afterwards. The segmentation heat map is formed by five layers, one for each label: left wall, center wall, right wall, floor, and ceiling.

Our proposed method adopts a semantic segmentation approach in conjugation with a discriminative classifier and layout refinement by a sliding window. The pipeline follows as described on Fig. 5.2, with the input image being fed into the MobileNetV2 network inside the DeepLabV3 semantic segmentation model, the coarsed layout result type is then classified by a support vectoring machine (SVM), and a type corresponding layout is slided vertically and horizontally over the previous obtained coarsed layout.

5.3.2 Network Architecture

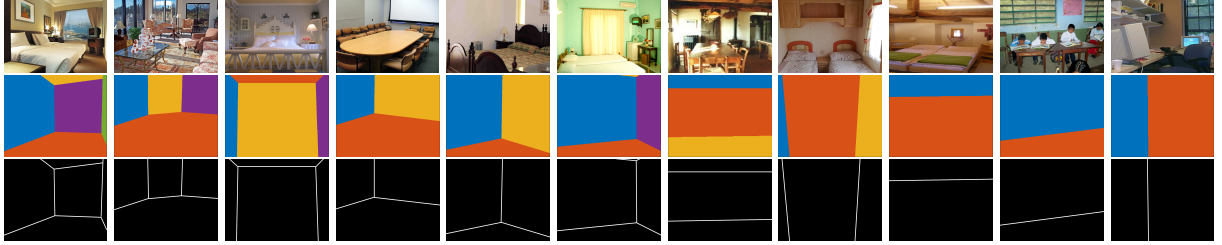
An indoor scene can be captured from different viewpoints; the geometric layout present in that view can be approximated to one of the eleven types of layout present on the LSUN dataset (Zhang et al. (2015)), as can be seen in Fig. 5.3. The ground truth of any room layout, either in a cluttered room or a open indoor scene, will coincide with some of the features that can be extracted from an image, such as the Manhattan lines. With the introduction of deep convolutional neural networks (DCNN) to tackle the room layout estimation, most of these low-level features and context clues were merged in end-to-end pipelines.

Similar to some previously proposed methods, we also used a encoder-decoder for semantic segmentation of the labels: *wall left*, *wall center*, *wall right*, *floor*, *ceiling*. With most of the state of the art results on the PASCAL VOC segmentation challenge (Everingham et al. (2011)) being achieved through the use of DCNN, it was a natural transition from the prior methods to the age of deep learning.

Our proposed method was developed aimed to edge and mobile devices. Therefore, we chose a more lightweight implementation, replacing the common VGG16, ResNet101, and ResNet50 previously used with the MobileNetV2 as backbone and DeepLabV3 as the semantic segmentation model of our network. In Fig. 5.2 is possible to observe the DeepLabV3 architecture, as introduced in Chen et al. (2017). The first block, the DCNN, is where the MobileNetV2 lies. As was proposed in Sandler et al. (2018), the use of shortcut connections between the bottlenecked layers allowed for a better preservation of relevant information throughout the network. Where the lightweight depthwise convolutions on the intermediate expansion layers allowed the filtering of non-linear features, which in this precise problem, is a benefit. With this framework as backbone, we are able to use only 4.52 million parameters, compared to the 138 millions of the VGG16. The DeepLabV3 model, with the proposed atrous spatial pyramid pooling module, which is a peculiar case of dilated residual networks, allows for a better probe of the convolutional features at multiple scales.

We fine-tuned a DeepLabV3 (Chen et al. (2017)) model pre-trained on the PAS-

CAL VOC 2012 dataset (Everingham et al. (2011)) with the network backbone of a MobileNetV2 (Sandler et al. (2018)) pre-trained on the MS-COCO dataset (Lin et al. (2014)). The information fed during the training to the network was the image and its semantic segmentation map, as can be seen on top and middle in Fig. 5.3.



Type 0 Type 1 Type 2 Type 3 Type 4 Type 5 Type 6 Type 7 Type 8 Type 9 Type 10

Figure 5.3: Different type of room layouts available on LSUN. From left to right, each room type is indexed from 0 to 10 as in Zhang et al. (2015). On top we see the images, on the middle the segmentation maps, and on the bottom the edge maps.

5.3.3 Layout Refinement

During the training of the network, we also trained a discriminative classifier. Inspired by the famous MNIST digits handwritten recognition (Deng (2012)), we trained a supporting vector machine (SVM) with the edges layouts, as the ones present on the bottom of Fig. 5.3.

After we obtained a coarse semantic segmentation from the DeepLabV3 model, we isolated its edges to become a binary image and fed it to the SVM to classify the room type, performing a layout hypothesis ranking. With the obtained result, we use randomly the original corresponding layout types. We called this stage the sliding window, as inspired by the single shot detectors (SSD) method (Liu et al. (2016)). Starting in the middle, with an additional margin of 5 pixels to each side, we move the image left and right, up and down, creating a heat map based on the coinciding edges between the neural network and the multiple layout templates. Afterwards, when the heat map coincides, or we pass of a maximum of 100 layout estimations, we assume the pixelwise maximum values across the heat map as the estimated layout. Is important to note that the LSUN dataset has unbalanced room layout types, which can be

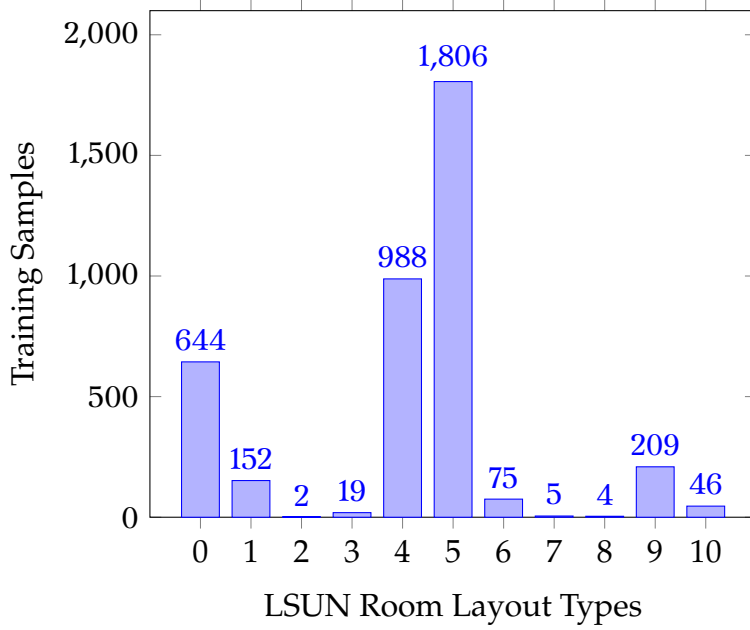


Figure 5.4: Distribution of the amount of samples per type of the training images from the LSUN dataset.

further analysed on table 5.2. Therefore, when a room layout type, limited to a lower sampling, if the heat map obtained by the sliding window does not coincide, the next in the ranking obtained by the SVM is considered as a possibility.

5.4 Experimental Results

5.4.1 Datasets

Our network was trained on the Large-scale Understanding Challenge (LSUN) room layout dataset (Zhang et al. (2015)), which contains a diverse collection of indoor scenes organized in: *bedroom, classroom, conference room, dinette home, dining room, hotel room, living room, and office*. All the provided layouts can be approximated to cuboids and are also divided in 11 different types of layouts, as can be seen in Fig. 5.3. The dataset is composed of 4000 training images, 394 validation images, and 1000 testing images. This dataset is unbalanced in terms of type distribution, as is shown in Fig. 5.4, which influences its ability to properly generalize and predict the under-sampled room layout types. We also performed tests on the Hedau et al. (2009) dataset, which is consisted of

Method	Pixel Error (%)	Training Dataset
Hedau et al. (2009)	21.20	Hedau
Del Pero et al. (2012)	16.30	Hedau
Gupta et al. (2010)	16.20	Hedau
Zhang et al. (2016b)	14.50	Hedau
Ramalingam et al. (2013)	13.34	Hedau
Mallya and Lazebnik (2015)	12.83	Hedau+
Schwing et al. (2012)	12.80	Hedau
Del Pero et al. (2013)	12.70	Hedau
Dasgupta et al. (2016)	9.73	Hedau
Zou et al. (2018)	9.69	LSUN
Ren et al. (2016)	8.67	Hedau
Lee et al. (2017)	8.36	Hedau+LSUN
Zhang et al. (2019)	7.94	Hedau
Ours	7.63	LSUN
Lin et al. (2018)	7.41	LSUN
Zhang et al. (2019)	7.36	LSUN
Zhao et al. (2017)	6.60	SUNRGBD+LSUN
Hsiao et al. (2019)	5.01	LSUN

Table 5.1: Room layout estimation performance on Hedau et al. (2009) dataset.

209 training images and 104 testing images, using our LSUN pretrained model. Mallya and Lazebnik (2015) also introduced an augmentation of the Hedau et al. (2009) called Hedau+, but we didn’t use it in our benchmarks.

5.4.2 Accuracy

The performance evaluation is measured by two standard metrics: pixel error, and corner error. On the Hedau et al. (2009) dataset, only the pixel error was measured.

The pixel error consists on measuring the pixelwise accuracy of the obtained layout with the ground truth, across all images, and average it. The corner error is relative to the Euclidean distances between the obtained corners and their associated ground truth, averaged across all images. The LSUN room layout challenge dataset (Zhang et al. (2015)) provides a toolkit to measure this evaluations.

Method	Pixel Error (%)	Corner Error (%)
Hedau et al. (2009)	24.23	15.48
Mallya and Lazebnik (2015)	16.71	11.02
Dasgupta et al. (2016)	10.63	8.20
Lee et al. (2017)	9.86	6.30
Ren et al. (2016)	9.31	7.95
Ours	6.94	5.46
Hsiao et al. (2019)	6.68	4.92
Zhang et al. (2019)	6.58	5.17
Lin et al. (2018)	6.25	—
Zhao et al. (2017)	5.29	3.84

Table 5.2: Room layout estimation performance on LSUN (Zhang et al. (2015)) dataset.

5.4.3 Experimental Results

We trained the network only on the LSUN dataset (Zhang et al. (2015)), but also performed the evaluation measuring on the Hedau et al. (2009) dataset.

The LSUN dataset presents a wide range of resolutions, therefore, we rescaled to 224x224 using bi-cubic interpolation prior to training. We also performed image augmentation while training by colour shifts, cropping, horizontal flipping, and jittering. Vertical flipping wasn’t considered due to the nature of the room’s orientation.

Table 5.1 compares the efficiency of the different available methods on the Hedau dataset, since the Hedau et al. (2009) publication. This benchmark only has pixelwise error evaluation. Note that the state of the art present in this benchmark was achieved using a different dataset for training. Even though our method was trained only on the LSUN dataset, we were still able to obtain good results and generalization.

On Table 5.2, we compare the obtained results using the LSUN toolkit. Here we have a pixelwise error rate and also a corner error. Although our results seem average, it is important to notice that we are achieving results over the average with a more smaller neural network. Some of the obtained room layout results are demonstrated in Fig. 5.5.

The proposed algorithm was implemented using tensorflow on a PC with an Intel



Figure 5.5: Examples of room layout estimations using our method on the LSUN dataset. From left to right: input image, semantic segmentation ground truth, our network prediction, final estimation after refinement.

i9-9900K CPU and a Nvidia GTX 2070 Super, with the semantic segmentation being obtained in $12ms$, and the posterior refinement in $9ms$. While these results are achieved on a PC, the authors are already developing an implementation on a mobile device.

5.5 Applications

5.5.1 Camera Pose Estimation

An interesting application over the retrieved indoor spatial layout, following the Manhattan assumption (Ramalingam et al. (2013)), is estimating a relative camera's pose. This application is suitable only when we have a good 2D approximation of the complete center wall with all the corners present, which only occurs in LSUN room layout type 0, as can be seen in Fig. 5.3.

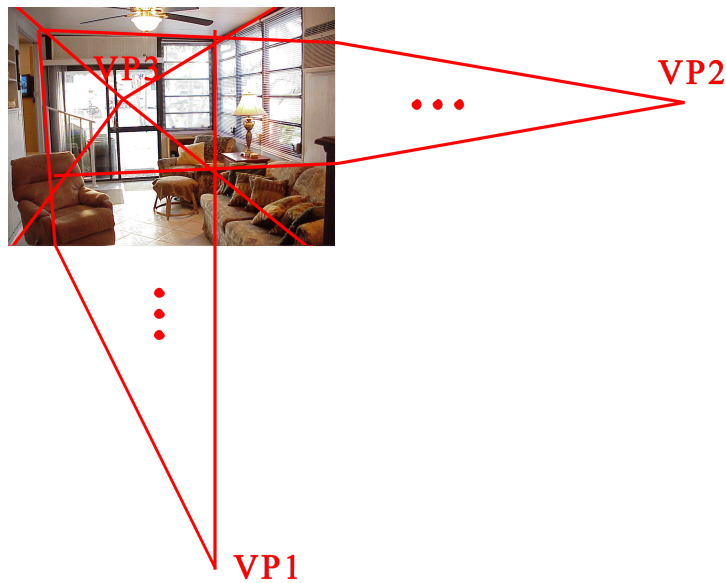


Figure 5.6: Example of the camera's pose relation to the vanishing points.

After using the proposed method, we estimated the outer vanishing points by the borders of the center wall, without the need for Manhattan lines (Ramalingam and Brand (2013)) calculation, and the inner vanishing point through the edges between walls-floor and walls-ceiling. Similar to what was used in Gakne and O'Keefe (2017) and using the opposite route of what was proposed in Wilczkowiak et al. (2001), we are able to obtain a prediction of the camera's pose estimation through the use of the estimated vanishing points. The inner vanishing point will be aligned with the center of the camera and will be used to find the camera's translation, while the outer vanishing points will be used to retrieve the camera's rotation matrix, as can be seen in Fig. 5.6. Without the camera's intrinsic matrix, we are unable to predict the scale.

Using our proposed method, some of the demanding computational operations, for example line segments ou Hough transform are replaced by estimations, which result in a lightweight application to retrieve a relative camera’s pose estimation.

5.6 Conclusions and Future Work

We presented an efficient room layout estimation method based on a smaller neural network, with MobileNetV2 as backbone and DeepLabv3 as the semantic segmentation model, with a posterior process of layout ranking based on a discriminative classifier of the edges and a sliding window method to refine the layout estimation. Our method is the first to implement a MobileNetV2 plus DeepLabv3 network for room layout estimation, with only a fraction of the parameters used on previous methods: VGG16 has 138 million, ResNet101 has 44.5 million, ResNet50 has 25.6 million, MobileNetV2 has 4.52 million. Even that our proposed method doesn’t outperform the current state of the art, we demonstrated that a network with a fraction of the parameters can achieve near state of the art results in either the Hedau, as the LSUN, datasets.

Future work will focus on implementing our method on edge and mobile devices, improving the sliding window method, and also transitioning and evaluating our method with a backbone of MobileNetV3, small and large, plus DeepLabV3+

6

Conclusions

This thesis presented the progress into the development of a monocular camera's pose estimation solution with the main goal of superimposing dynamic content over a rectangular cuboid room's vertical planes - walls. The developments and innovations were divided across the previous three chapters, with the different stages of the pipeline being further detailed in each publication. Although each chapter contains its own conclusion and discussion of the work done, it follows below a final remark over the work done.

Since the main goal only requires at the end the replacement of the corresponding pixels in a frame to achieve the desired superimposition, it is necessary to explore what was done to achieve that result. Environment recognition is an arduous task, either computational, or in terms of algorithms. Normally it involves the use of depth

cameras to easily retrieve the 3D shape of a scene. As one of the necessities for this thesis is the use of a monocular camera associated with the limits of the mobile devices' performance, such widespread solutions weren't viable at the time of this thesis creation. In order to achieve this goal it was necessary to explore the limits of markerless image recognition and to find a way to initially calculate the camera's location.

With the developed solution, there's a scalable possibility of using the initial recognition across an entire museum, and not only a specific room without using any additional localization method besides computer vision. Within this research, enhanced hybrid searching methods were developed and a further comprehensive homography refinement was created, increasing the frame's validation, which allowed for a better camera pose estimation calculation, with less sparse results.

Another innovation presented was the fusion of the previous developed method for retrieving the environments' geometric shapes with the homography calculation of the first localization, allowing for the recalculation of an almost perfect planar homography even with uncalibrated cameras, allowing for the estimation of the camera's matrix. This method can filter almost all the sparse and outlier estimations of the camera's pose in the real-world environment.

The progressive tracking introduced the possibility of a continuous superimposition regardless of the monocular camera's quality, only requiring the initial localization. This plane tracking method is extremely effective performance-wise, and achieves exceedingly obtuse angles which are not possible using only markerless based planar recognition and homography. The incorporation of Kalman filters, not only to the outside world, but also to the camera's 6DoF, allowed for a more smooth and seamless content superimposition.

Furthermore, the room spatial layout estimation method proposed offers a novel solution to the problem using smaller networks with less parameters and a discriminative classifier and sliding windows for refinement of the layout hypothesis. This lightweight method offers the possibility of an edge or mobile device implementation, being another step to a cloudless interaction with our surroundings. The application

for camera orientation introduced using this method presents one of its possibilities.

6.1 Future Work

The presented work was done in only one museum, the Archaeological and Lapidary Museum Prince Henry the Navigator in Faro. In terms of future work, it is necessary to expand into different environments, either other museums, layouts or places. Although the algorithm is currently heavily dependent on the initialization of the localization, there is a possibility of isolating part of the algorithm and removing the initial scan, allowing it to function over any distinct vertical planes, which would allow for a more diverse application.

It is also imperative the creation or adaptation of this work to a proper dataset so that a feasible benchmark can be produced. This is specially complicated due to the fact that almost all of the available datasets are aimed at depth cameras, or to obtain the flow estimation while driving a car. This work focuses primarily in localizing and tracking the camera's direction using only the walls available in the surrounding environment in a bi-dimensional correspondence to the tridimensional geometric layout.

The current implementation also lacks a proper performance evaluation on a mobile device across the environments. It is necessary to implement an additional method so when the tracking is lost the pose estimation can continue to be estimated, i.e., mobile devices magnetometers, gyroscopes or accelerometers. An analysis using multiple mobile devices with distinct camera specifications would also be beneficial to test the behaviour of the uncalibrated homography reconstruction and the progressive tracking performance.

The hybrid searching method could evolve into a more complete localization if it would allow for additional information to be associated while scanning the existent rooms, i.e., GPS, beacons, magnetometer readings, wi-fi ssids. With the current evolution in the artificial intelligence field, there should be room for major improvements and restructuring in the artificial neural network methods used in conjugation with the

binary feature descriptors.

Nevertheless, the proposed objectives were reached, with even additional innovation in-between them, including several publications. Even though improvements could be made, this thesis proved that it is possible to continuously obtain the camera's pose estimation across a room, even with uncalibrated monocular cameras, and only processing 2D information.

Regarding the room layout estimation using machine and deep learning, which is the latest method presented on this thesis, there is still room for improvements and additional benchmarks and applications development. A deep analysis can also be performed using different models for backbone and other semantic segmentations models. Nonetheless, the implementation and benchmark of this proposed method is crucial on edge and mobile devices.

6.2 Publications

While pursuing my master's degree, several peer review works were published, being a total of seven articles for international conferences, one submission for a journal and three other publications for local conferences. Amidst the following list there are present the same four documents presented through the main chapters of this thesis.

- **Veiga, R.**, Bajireanu, R., Pereira, J., Sardo, J., Cardoso, P.J.S., and Rodrigues, J.M.F. (2017). **Indoor environment and human shape detection for augmented reality: an initial study**. In Procs 23rd edition of the Portuguese Conference on Pattern Recognition, Amadora, Portugal, 28 Oct., pp. 67-68
- Pereira, J.A.R., Sardo, J.D.P., Freitas, M.A.G., **Veiga R.**, Cardoso, P.J.S., Rodrigues, J.M.F. (2017) **MIRAR: Mobile Image Recognition based Augmented Reality Framework**, accepted for Int. Congress on Engineering and Sustainability in the XXI Century, 11 - 13 October, Faro, Portugal
- João D. P. Sardo, João A. R. Pereira, **Ricardo J. M. Veiga**, Jorge Semião, Pedro J. S.

- Cardoso, João M. F. Rodrigues. (2018) **Multisensorial Portable Device for Augmented Reality Experiences in Museums**. International Journal of Education and Learning Systems, 3, 60-69
- Pedro J.S. Cardoso, Pedro Guerreiro, João A. R. Pereira, **Ricardo J.M. Veiga** (2018) **A Route Planner Supported on Recommender Systems Suggestions: Enhancing Visits to Cultural Heritage Places**. In Procs 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, Thessaloniki, Greece, 20-22 June
 - Rodrigues, J.M.F., **Veiga, R.**, Bajireanu, R., Lam, R., Pereira, J., Sardo, J., Cardoso, P.J.S., and Bica, P. (2018) **Mobile augmented reality framework - MIRAR**. In 12th International Conference on Universal Access in Human-Computer Interaction, integrated in the 20th HCII, Las Vegas, USA, pp. 102–121
 - Bajireanu, R., Pereira, J., **Veiga, R.**, Sardo, J., Cardoso, P.J.S., Lam, R., and Rodrigues, J.M.F. (2018) **Mobile human shape superimposition: an initial approach using OpenPose**. In Procs 18th International Conference on Applied Computer Science, Dubrovnik, Croatia, 26-28 Sep.
 - **Ricardo J. M. Veiga**, João A. R. Pereira, João D. P. Sardo, Roman Bajireanu, Pedro J. S. Cardoso, João M. F. Rodrigues (2019). **Augmented Reality Indoor Environment Detection: Proof-of-Concept**. In WSEAS Transactions on Mathematics, ISSN / E-ISSN: 1109-2769 / 2224-2880, Volume 18, 2019, Art. 28, pp. 203-210
 - Rodrigues J.M.F., **Veiga R.J.M.**, Bajireanu R., Lam R., Cardoso P.J.S., Bica P. (2019) **AR Contents Superimposition on Walls and Persons**. In: Antona M., Stephanidis C. (eds) Universal Access in Human-Computer Interaction. Theory, Methods and Tools. HCII 2019. Lecture Notes in Computer Science, vol 11572, pp. 638-645, Springer, Cham. DOI: 10.1007/978-3-030-23560-4_46
 - **Veiga, Ricardo J.M.**, Rodrigues, João M.F. (2019) **Indoor Wall Detection, Tracking and Superimposition**. In Procs 25th edition of the Portuguese Conference on

Pattern Recognition, Porto, Portugal, 31 Oct, pp. 125-128.

- **Veiga, Ricardo J.M., Cardoso, Pedro J.S., Rodrigues, João M.F. (2020) Efficient Small-Scale Network for Room Spatial Layout Estimation** In Submission to 14th International conference on Universal Access in Human-Computer Interaction, integrated in the 22nd HCII, Copenhagen, Denmark, 19-24 July

References

- Araki, N. and Muraoka, Y. (2008). Follow-the-trial-fitter: real-time dressing without undressing. In *Procs IEEE Conf. on Digital Information Management*, pages 33–38, London, UK.
- Artoolkit (2017). ARtoolKit, the world’s most widely used tracking library for augmented reality. <http://artoolkit.org/>. Retrieved: Nov. 16, 2017.
- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. (2001). Recent advances in Augmented Reality. *IEEE Computer Graphics and Applications*, 21(6):34–47.
- Babahajiani, P., Fan, L., and Gabbouj, M. (2014). Object recognition in 3D point cloud of urban street scene. In *Procs Asian Conf. on Computer Vision*, pages 177–190. Springer.
- Baggio, D. L. (2012). *Mastering OpenCV with practical computer vision projects*. Packt Publishing Ltd.
- Bailey, T. and Durrant-Whyte, H. (2006). Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics & Automation Magazine*, 13(3):108–117.
- Bajireanu, R., Pereira, J. A., Veiga, R. J., Sardo, J. D., Cardoso, P. J., Lam, R., and Rodrigues, J. M. (2018). Mobile human shape superimposition: an initial approach using OpenPose. *Procs 18th Int. Conf. on Applied Computer Science*.
- Bartoli, A. and Sturm, P. (2005). Structure-from-motion using lines: Representation, triangulation, and bundle adjustment. *Computer vision and image understanding*, 100(3):416–441.
- Bhole, C. and Pal, C. (2012). Automated person segmentation in videos. In *21st International Conference on Pattern Recognition (ICPR)*, pages 3672–3675. IEEE.
- Boniardi, F., Caselitz, T., Kümmerle, R., and Burgard, W. (2019a). A pose graph-based localization system for long-term navigation in cad floor plans. *Robotics and Autonomous Systems*, 112:84–97.
- Boniardi, F., Valada, A., Mohan, R., Caselitz, T., and Burgard, W. (2019b). Robot localization in floor plans using a room layout edge extraction network. *arXiv preprint arXiv:1903.01804*.
- Bouguet, J.-Y. (2001). Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm. *Intel Corporation*, 5(1-10):4.

- Buch, A. G., Kraft, D., Kamarainen, J.-K., Petersen, H. G., and Krüger, N. (2013). Pose estimation using local structure-specific shape and appearance context. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2080–2087. IEEE.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 8(6):679–698.
- Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., and Sheikh, Y. (2018). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*.
- Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In *CVPR*, volume 1(2), page 7.
- Catchoom (2017). Catchoom. <http://catchoom.com/>. Retrieved: Nov. 16, 2017.
- Chao, Y.-W., Choi, W., Pantofaru, C., and Savarese, S. (2013). Layout estimation of highly cluttered indoor scenes using geometric and semantic cues. In *International Conference on Image Analysis and Processing*, pages 489–499. Springer.
- Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818.
- Cheng, K.-H. and Tsai, C.-C. (2013). Affordances of augmented reality in science learning: Suggestions for future research. *J. Science Education and Technology*, 22(4):449–462.
- Coughlan, J. M. and Yuille, A. L. (2001). The manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In *Advances in Neural Information Processing Systems*, pages 845–851.
- Dai, J., He, K., and Sun, J. (2016). Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3150–3158.
- Dasgupta, S., Fang, K., Chen, K., and Savarese, S. (2016). Delay: Robust spatial layout estimation for cluttered indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 616–624.
- Del Pero, L., Bowdish, J., Fried, D., Kermgard, B., Hartley, E., and Barnard, K. (2012). Bayesian geometric modeling of indoor scenes. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2719–2726. IEEE.
- Del Pero, L., Bowdish, J., Kermgard, B., Hartley, E., and Barnard, K. (2013). Understanding bayesian rooms using composite 3d object models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 153–160.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142.

- Duan, W. (2011). *Vanishing points detection and camera calibration*. PhD thesis, University of Sheffield.
- Durrant-Whyte, H. and Bailey, T. (2006). Simultaneous localization and mapping: part I. *IEEE robotics & automation magazine*, 13(2):99–110.
- Elqursh, A. and Elgammal, A. (2011). Line-based relative pose estimation. In *Procs IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3049–3056. IEEE.
- Engel, J., Koltun, V., and Cremers, D. (2018). Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):611–625.
- Engel, J., Schöps, T., and Cremers, D. (2014). LSD-SLAM: Large-scale direct monocular SLAM. In *Procs European Conf. on Computer Vision*, pages 834–849. Springer.
- Engel, J., Sturm, J., and Cremers, D. (2013). Semi-dense visual odometry for a monocular camera. In *Procs IEEE Int. Conf. on Computer Vision*, pages 1449–1456.
- Erhan, D., Szegedy, C., Toshev, A., and Anguelov, D. (2014). Scalable object detection using deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2154.
- Erra, U., Scanniello, G., and Colonnese, V. (2018). Exploring the effectiveness of an augmented reality dressing room. *Multimedia Tools and Applications*, pages 1–31.
- Estimate (2017). Create magical experiences in the physical world. <https://goo.gl/OHW04y>. Retrieved: April 04, 2017.
- Everingham, M., Van Gool, L., Williams, C., Winn, J., and Zisserman, A. (2011). The pascal visual object classes challenge 2012 (voc2012) results (2012). In URL <http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html>.
- Facecake (2016). Facecake. <http://www.facecake.com/>. Retrieved: September. 17, 2018.
- Fang, H., Xie, S., Tai, Y.-W., and Lu, C. (2017). RMPE: Regional multi-person pose estimation. In *Procs IEEE Int. Conf. on Computer Vision*, volume 2.
- Farneback, G. (2003). Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer.
- Figat, J., Kornuta, T., and Kasprzak, W. (2014). Performance evaluation of binary descriptors of local features. In *Proc. International Conference on Computer Vision and Graphics*, pages 187–194. Springer.
- Fleet, D. and Weiss, Y. (2006). Optical flow estimation. In *Handbook of mathematical models in computer vision*, pages 237–257. Springer.
- Gakne, P. V. and O’Keefe, K. (2017). Monocular-based pose estimation using vanishing points for indoor image correction. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7. IEEE.
- Gilks, W. R., Richardson, S., and Spiegelhalter, D. (1995). *Markov chain Monte Carlo in practice*. Chapman and Hall/CRC.

- Jimeno, J., Portales, C., Coma, I., Fernandez, M., and Martinez, B. (2017). Combining traditional and indirect augmented reality for indoor crowded environments. a case study on the casa batlló museum. *Computers & Graphics*, 69:92–103.
- Girshick, R. (2015). Fast R-CNN. In *Procs IEEE Conf. on Computer Vision*, pages 1440–1448.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Procs IEEE Conf. on Computer Vision and Pattern Recognition*, pages 580–587.
- Google (2018). TensorFlow - an open-source machine learning framework for everyone. <https://www.tensorflow.org/>. Retrived: January 14, 2018.
- Gupta, A., Hebert, M., Kanade, T., and Blei, D. M. (2010). Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *Advances in neural information processing systems*, pages 1288–1296.
- Gupta, S., Arbeláez, P., Girshick, R., and Malik, J. (2015). Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. *International Journal of Computer Vision*, 112(2):133–149.
- Haines, O. and Calway, A. (2012). Detecting planes and estimating their orientation from a single image. In *BMVC*, pages 1–11.
- Hallquist, A. and Zakhor, A. (2013). Single view pose estimation of mobile devices in urban environments. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, pages 347–354. IEEE.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask R-CNN. In *Procs IEEE Int. Conf. on Computer Vision*, pages 2980–2988.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Hedau, V., Hoiem, D., and Forsyth, D. (2009). Recovering the spatial layout of cluttered rooms. In *2009 IEEE 12th international conference on computer vision*, pages 1849–1856. IEEE.
- Hedau, V., Hoiem, D., and Forsyth, D. (2010). Thinking inside the box: Using appearance models and context based on room geometry. In *European Conference on Computer Vision*, pages 224–237. Springer.
- Hernández-Vela, A., Reyes, M., Ponce, V., and Escalera, S. (2012). Grabcut-based human segmentation in video sequences. *Sensors*, 12(11):15376–15393.
- HMS (2017). Srbija 1914 / augmented reality exhibition at historical museum of Serbia, Belgrade. <https://vimeo.com/126699550>. Retrieved: April 04, 2017.
- Hoiem, D., Efros, A. A., and Hebert, M. (2005). Geometric context from a single image. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 654–661. IEEE.

- Hoiem, D., Efros, A. A., and Hebert, M. (2007). Recovering surface layout from an image. *International Journal of Computer Vision*, 75(1):151–172.
- Hough, P. V. (1962). Method and means for recognizing complex patterns. US Patent 3,069,654.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017a). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017b). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Hsiao, C.-W., Sun, C., Sun, M., and Chen, H.-T. (2019). Flat2layout: Flat representation for estimating layout of general room types. *arXiv preprint arXiv:1905.12571*.
- Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al. (2016). Speed/accuracy trade-offs for modern convolutional object detectors. *arXiv preprint arXiv:1611.10012*.
- Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. In *Procs IEEE Conf. on Computer Vision and Pattern Recognition*, volume 4, pages 3296 – 3297, Honolulu, HI, USA.
- Hulik, R., Spänzel, M., Smrz, P., and Materna, Z. (2014). Continuous plane detection in point-cloud data based on 3D Hough transform. *J. of Visual Communication and Image Representation*, 25(1):86–97.
- InformationWeek (2017). Informationweek: 10 fantastic iPhone, Android Apps for museum visits. <https://goo.gl/XF3rj4>. Retrieved: April 04, 2017.
- Isikdogan, F. and Kara, G. (2012). A real time virtual dressing room application using Kinect. *Computer Vision Course Project*.
- Kendall, A., Grimes, M., and Cipolla, R. (2015). PoseNet: A convolutional network for real-time 6-dof camera relocalization. In *Procs of the IEEE Int. Conf. on Computer Vision*, pages 2938–2946.
- Kft., F. I. (2016). Fitnect. <http://www.fitnect.hu/>. Retrieved: September. 17, 2018.
- Kim, I. (2018). tf-pose-estimation. <https://bit.ly/2HJxxcq>. Retrieved: Apr. 10, 2018.
- Kiryati, N., Eldar, Y., and Bruckstein, A. M. (1991). A probabilistic Hough transform. *Pattern recognition*, 24(4):303–316.
- Kudan (2017). Kudan computer vision. <https://www.kudan.eu/>. Retrieved: Nov. 16, 2017.
- Layar (2017). Layar. <https://www.layar.com/>. Retrieved: Nov. 16, 2017.

- Lee, C.-Y., Badrinarayanan, V., Malisiewicz, T., and Rabinovich, A. (2017). Roomnet: End-to-end room layout estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4865–4874.
- Lee, D. C., Hebert, M., and Kanade, T. (2009). Geometric reasoning for single image structure recovery. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2136–2143. IEEE.
- Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). BRISK: Binary robust invariant scalable keypoints. In *Procs IEEE Int. Conf. on Computer Vision*, pages 2548–2555. IEEE.
- Lin, H. J., Huang, S.-W., Lai, S.-H., and Chiang, C.-K. (2018). Indoor scene layout estimation from a single image. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 842–847. IEEE.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.
- Liu, C., Schwing, A. G., Kundu, K., Urtasun, R., and Fidler, S. (2015). Rent3d: Floor-plan priors for monocular layout estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3413–3421.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.
- Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.
- Lv, Q., Josephson, W., Wang, Z., Charikar, M., and Li, K. (2007). Multi-probe lsh: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases*, pages 950–961. VLDB Endowment.
- Mallya, A. and Lazebnik, S. (2015). Learning informative edge maps for indoor scene layout prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 936–944.
- Martin-Brualla, R., He, Y., Russell, B. C., and Seitz, S. M. (2014). The 3d jigsaw puzzle: Mapping large indoor spaces. In *European Conference on Computer Vision*, pages 1–16. Springer.
- MAX, D. (2018). 3DS MAX. <https://www.autodesk.com/products/3ds-max/overview>. Retrieved: Dezember 3, 2018.
- Mendonça, P. R. and Cipolla, R. (1999). A simple technique for self-calibration. In *cvpr*, page 1500. IEEE.
- Muja, M. and Lowe, D. G. (2012). Fast matching of binary features. In *Procs 9th Conf. Computer and Robot Vision*, pages 404–410. IEEE.

- Mur-Artal, R., Montiel, J. M. M., and Tardos, J. D. (2015). ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163.
- OpenCV (2017). OpenCV. <http://opencv.org/>. Retrieved: April 04, 2017.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66.
- Ouyang, W. and Wang, X. (2013). Joint deep learning for pedestrian detection. In *Procs Int. Conf. Computer Vision*, pages 2056–2063. IEEE.
- Pádua, L., Adão, T., Narciso, D., Cunha, A., Magalhães, L., and Peres, E. (2015). Towards modern cost-effective and lightweight augmented reality setups. *Int. J. of Web Portals*, 7(2):33–59.
- Papandreou, G., Zhu, T., Chen, L.-C., Gidaris, S., Tompson, J., and Murphy, K. (2018). Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. *arXiv preprint arXiv:1803.08225*.
- Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., and Murphy, K. (2017). Towards accurate multiperson pose estimation in the wild. *arXiv preprint arXiv:1701.01779*, 8.
- Park, S. and Yoo, J.-H. (2014). Human segmentation based on grabcut in real-time video sequences. In *IEEE International Conference on Consumer Electronics (ICCE)*, pages 111–112. IEEE.
- Pereira, J. A., Veiga, R. J., de Freitas, M. A., Sardo, J., Cardoso, P. J., and Rodrigues, J. M. (2017). MIRAR: Mobile image recognition based augmented reality framework. In *Procs Int. Congress on Engineering and Sustainability in the XXI Century*, pages 321–337. Springer.
- Portales, C., Vinals, M. J., and Alonso-Monasterio, P. (2010). Ar-immersive cinema at the aula natura visitors center. *IEEE MultiMedia*, 17(4):8–15.
- Qiao, H., Li, Y., Li, F., Xi, X., and Wu, W. (2015). Biologically inspired model for visual cognition achieving unsupervised episodic and semantic feature learning. *IEEE transactions on cybernetics*, 46(10):2335–2347.
- Qualcomm (2017). Invisible museum. <https://goo.gl/aS0NKh>. Retrieved: April 04, 2017.
- Ramalingam, S. and Brand, M. (2013). Lifting 3d manhattan lines from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 497–504.
- Ramalingam, S., Pillai, J. K., Jain, A., and Taguchi, Y. (2013). Manhattan junction catalogue for spatial reasoning of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3065–3072.
- Redmon, J. and Farhadi, A. (2016). Yolo9000: better, faster, stronger. *arXiv preprint arXiv:1612.08242*.

- Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.
- Ren, Y., Li, S., Chen, C., and Kuo, C.-C. J. (2016). A coarse-to-fine indoor layout estimation (cfile) method. In *Asian Conference on Computer Vision*, pages 36–51. Springer.
- Ren, Z. and Sudderth, E. B. (2016). Three-dimensional object detection and layout prediction using clouds of oriented gradients. In *Procs IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1525–1533.
- Riba Pi, E. (2015). Implementation of a 3d pose estimation algorithm. Master’s thesis, Universitat Politècnica de Catalunya.
- Ring, J. (1963). The laser in astronomy. *New Scientist*, 18(344):672–673.
- Rodrigues, J., Lessa, J., Gregório, M., Ramos, C., and Cardoso, P. (2016). An initial framework for a museum application for senior citizens. In *Procs 7th Int. Conf. on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion*.
- Rodrigues, J., Pereira, J., Sardo, J., Freitas, M., Cardoso, P., Gomes, M., and Bica, P. (2017). Adaptive card design UI implementation for an augmented reality museum application. In *Procs 11th Int. Conf. on Universal Access in Human-Computer Interaction*.
- Rodrigues, J. M., Veiga, R. J., Bajireanu, R., Lam, R., Pereira, J. A., Sardo, J. D., Cardoso, P. J., and Bica, P. (2018). Mobile Augmented Reality framework-MIRAR. In *Procs Int. Conf. on Universal Access in Human-Computer Interaction*, pages 102–121. Springer.
- Rother, C. (2002). A new approach to vanishing point detection in architectural environments. *Image and Vision Computing*, 20(9-10):647–655.
- Rother, C., Kolmogorov, V., and Blake, A. (2004). Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23(3), pages 309–314. ACM.
- Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In *Proc. Int. Conf. on Computer Vision*, pages 2564–2571. IEEE.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520.
- Sardo, J., Semião, J., Monteiro, J., Pereira, J. A., de Freitas, M. A., Esteves, E., and Rodrigues, J. M. (2017). Portable device for touch, taste and smell sensations in augmented reality experiences. In *Procs Int. Congress on Engineering and Sustainability in the XXI Century*, pages 305–320. Springer.
- Schwing, A. G., Fidler, S., Pollefeys, M., and Urtasun, R. (2013). Box in the box: Joint 3d layout and object reasoning from single images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 353–360.

- Schwing, A. G., Hazan, T., Pollefeys, M., and Urtasun, R. (2012). Efficient structured prediction for 3d indoor scene understanding. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2815–2822. IEEE.
- Sermanet, P., Kavukcuoglu, K., Chintala, S., and LeCun, Y. (2013). Pedestrian detection with unsupervised multi-stage feature learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3626–3633. IEEE.
- Serrão, M., Shahrabadi, S., Moreno, M., José, J. T., Rodrigues, J. I., Rodrigues, J. M. F., and du Buf, J. M. H. (2015). Computer vision and GIS for the navigation of blind persons in buildings. *Universal Access in the Information Society*, 14(1):67–80.
- Shi, J. and Tomasi, C. (1993). Good features to track. Technical report, Cornell University.
- SM (2017). Science museum - atmosphere gallery. <https://vimeo.com/20789653>. Retrieved: April 04, 2017.
- Sousa, L., Rodrigues, J., Monteiro, J., Cardoso, P., Semião, J., and Alves, R. (2014). A 3D gesture recognition interface for energy monitoring and control applications. *Procs of ACE'14*, pages 62–71.
- Sun, C., Hsiao, C.-W., Sun, M., and Chen, H.-T. (2019). Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1047–1056.
- Tareen, S. A. K. and Saleem, Z. (2018). A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In *Procs Int. Conf. on Computing, Mathematics and Engineering Technologies*, pages 1–10. IEEE.
- Tateno, K., Tombari, F., Laina, I., and Navab, N. (2017). Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Procs IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2.
- Tian, Y., Luo, P., Wang, X., and Tang, X. (2015). Deep learning strong parts for pedestrian detection. In *Procs of the IEEE international conference on computer vision*, pages 1904–1912.
- Tolias, G. and Avrithis, Y. (2011). Speeded-up, relaxed spatial matching. In *Procs IEEE Int. Conf. on Computer Vision*, pages 1653–1660. IEEE.
- Tome, D., Russell, C., and Agapito, L. (2017). Lifting from the deep: Convolutional 3D pose estimation from a single image. *Procs IEEE Conf. Computer Vision and Pattern Recognition*, pages 2500–2509.
- TWSJ (2017). The wall street journal: Best apps for visiting museums. <https://goo.gl/cPTyP9>. Retrieved: April 04, 2017.
- Unity (2018). Unity3D. <https://unity3d.com/pt>. Retrieved: Jan. 10, 2018.
- Vainstein, N., Kuflik, T., and Lanir, J. (2016). Towards using mobile, head-worn displays in cultural heritage: User requirements and a research agenda. In *Proc. 21st Int. Conf. on Intelligent User Interfaces*, pages 327–331. ACM.

- Veiga, R. J., Bajireanu, R., Pereira, J. A., Sardo, J. D., Cardoso, P. J., and Rodrigues, J. M. (2017). Indoor environment and human shape detection for augmented reality: an initial study. *Procs 23rd Portuguese Conf. Pattern Recognition*, page 21.
- Veiga, R. J., Pereira, J. A., Sardo, J. D., Bajireanu, R., Cardoso, P. J., and Rodrigues, J. M. (2018). Augmented Reality indoor environment detection: Proof-of-concept. *Procs Applied Mathematics And Computer Science*.
- Vincent, E. and Laganière, R. (2001). Detecting planar homographies in an image pair. In *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, pages 182–187. IEEE.
- Wang, H., Gould, S., and Koller, D. (2010). Discriminative learning with latent variables for cluttered indoor scene understanding. In *European Conference on Computer Vision*, pages 497–510. Springer.
- Wilczkowiak, M., Boyer, E., and Sturm, P. (2001). Camera calibration and 3d reconstruction from single images using parallelepipeds. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, volume 1, pages 142–148. IEEE.
- Xiao, J. and Furukawa, Y. (2014). Reconstructing the world’s museums. *International journal of computer vision*, 110(3):243–258.
- Xiao, J., Zhang, J., Adler, B., Zhang, H., and Zhang, J. (2013). Three-dimensional point cloud plane segmentation in both structured and unstructured environments. *Robotics and Autonomous Systems*, 61(12):1641–1652.
- Xu, Q., Li, L., Lim, J. H., Tan, C. Y. C., Mukawa, M., and Wang, G. (2014). A wearable virtual guide for context-aware cognitive indoor navigation. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, pages 111–120. ACM.
- Zhang, L., Lin, L., Liang, X., and He, K. (2016a). Is faster R-CNN doing well for pedestrian detection? In *European Conference on Computer Vision*, pages 443–457. Springer.
- Zhang, W., Zhang, W., and Gu, J. (2019). Edge-semantic learning strategy for layout estimation in indoor environment. *IEEE transactions on cybernetics*.
- Zhang, W., Zhang, W., Liu, K., and Gu, J. (2016b). Learning to predict high-quality edge maps for room layout estimation. *IEEE Transactions on Multimedia*, 19(5):935–943.
- Zhang, Y., Yu, F., Song, S., Xu, P., Seff, A., and Xiao, J. (2015). Large-scale scene understanding challenge: Room layout estimation. In *CVPR Workshop*.
- Zhao, H., Lu, M., Yao, A., Guo, Y., Chen, Y., and Zhang, L. (2017). Physics inspired optimization on semantic transfer features: An alternative method for room layout estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10–18.
- Zou, C., Colburn, A., Shan, Q., and Hoiem, D. (2018). Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2051–2059.