FCT Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR

Formação Avançada de Recursos Humanos
Formulário de candidatura

Terminar sessão
Logout

Página principal
Main page

Página do orientador
Supervisor's page

Orientador / responsável pela formação
Supervisor

Formulário de candidatura do orientando
Proponent's application form

Referência da candidatura do orientando
Reference of the proponent's application
SFRH/BD/80596/2011
Nome e email do orientando
Proponent's name and email
Pedro Emanuel Marques Dias Pinto
pemdpinto@gmail.com

Esta candidatura foi lacrada a 27-06-2011 (10:32)
This application has been submitted at 27-06-2011 (10:32)

●

●

●

●

●

●

● ●

●

●

● ●

●

●

●

Engenharia Electrotécnica e Informática
Electrotechnical and Computer Engineering

Local de realização da Bolsa
Location of fellowship activities

No País
In Portugal

3. Procurador do candidato
3. Candidate's representative

(vazio)
(void)

Facultativo para o caso de bolsas totalmente no país
Optional for fellowships totally in Portugal

4. Programa de trabalho
4. Working programme

4.1. Título do programa de trabalhos
4.1. Title of the working programme

Mecanismos automáticos de atenção para um sistema avançado de apoio à condução

Domínio Científico
Scientific Domain

Mecatrónica

Data de início do programa de trabalhos **Work programme starting date**	**Duração (meses)** **Duration (month)**
01-10-2011	36
Data de início pretendida para a bolsa **Fellowship starting date**	**Duração (meses)** **Duration (month)**
01-10-2011	36

4.2. Sumário
4.2. Abstract

O trabalho consistirá em desenvolver e implementar algoritmos de percepção baseados no conceito de foco de atenção para detectar alvos a seguir por um sistema avançado de ajuda à condução. A utilização de mecanismos de atenção tem inspiração biológica e permite optimizar os recursos de processamentos, direccionando-os para as zonas com maior probabilidade de conterem elementos de interesse. Deste modo é possível aumentar o desempenho dos algoritmos e aplicá-los a tarefas criticas com requisitos de tempo-real como o caso da detecção de peões ou de veículos.

Este trabalho será validado através da sua integração num veículo autónomo actualmente em desenvolvimento na Universidade de Aveiro.

4.3. Estado da Arte
4.3. State of the art

Segundo dados estatísticos, em 2010, em Portugal, mais de 24 mil acidentes com automóveis ligeiros de passageiros resultaram em colisão [1]. Acidentes deste tipo podem, em muitos casos, ser evitados caso o condutor reaja atempadamente ao perigo. O factor humano é decisivo, pois numa situação real, os condutores são obrigados a decidir e executar em fracções de segundo uma manobra evasiva por forma a evitar uma colisão iminente.

O sector aeronáutico, há muito que se deparou com problemas semelhantes. A solução passou por uma automatização gradual das aeronaves que revolucionou o próprio processo de pilotagem. Sistemas como o ACAS (Airborne Collision Avoidance System) previnem acidentes aéreos evitando tragédias e salvando vidas [2]. Quanto aos veículos terrestres, a investigação tem sido contínua no sentido de conceber veículos completamente autónomos. Entre 2004 e 2007 o Departamento de Defesa Americano promoveu o DARPA Challange, uma competição com o objectivo de desenvolver tecnologias que possibilitem veículos movidos sem condutor [3]. Em 2010, no âmbito do VisLab Intercontinental Autonomous Challenge quatro veículos autónomos percorreram 15 mil quilómetros entre Parma e Shangai [4]. Também nesse ano um veículo desenvolvido em colaboração com a Universidade de Stanford percorreu o famoso traçado de Pikes Peak sem condutor em cerca de 27 minutos [5]. Um valor que mesmo longe do recorde obtido por pilotos humanos, provou a viabilidade da tecnologia. Apesar do sucesso alcançado com os veículos autónomos, ainda é necessário um período de amadurecimento até estas tecnologias estarem aptas a serem lançadas para o mercado. Para além disso é também necessário uma adaptação do quadro legislativo de modo a enquadrar estas novas realidades.

Como passo intermédio para esta transição, a comunidade científica tem tentado melhorar a segurança rodoviária criando ajudas electrónicas que auxiliem de forma activa os condutores. Desta investigação têm surgido automóveis com sistemas cada vez mais sofisticados. Várias marcas propõem actualmente veículos capazes não só de manter o veículo a uma velocidade constante mas também de travar automaticamente em caso de necessidade. Recentemente a Volvo lançou no mercado um sistema composto por uma câmara e um radar que visa detectar peões e em caso de emergência actuar nos travões de modo a evitar atropelamentos [6]. Este sistema apesar de eficaz em cidade, tem a desvantagem de apenas funcionar a baixa velocidade.

Os sistemas de prevenção de colisões dependem de algoritmos capazes de reconhecerem situações de perigo de forma eficiente, pois só assim é possível actuar em tempo útil por forma a evitar uma possível colisão. Na natureza, também os seres vivos, ao longo da sua evolução foram desenvolvendo mecanismos de defesa que lhes permitem alertar para as ameaças que os rodeiam. O olho humano Humano é um exemplo disso. A retina contém dois tipos de foto-receptores, chamados cones e bastonetes, que transformam a luz em impulsos eléctricos. Os primeiros encontram-se concentrados na região fovea e permitem reconhecer detalhes como a cor, enquanto que os segundos são bastante mais sensíveis à variação de luminosidade, sendo responsáveis pela visão periférica. Esta sensibilidade fora da zona de atenção terá permitido ao homem primitivo reconhecer facilmente movimentos de animais predadores. Esta separação de funções foi uma resposta da natureza às limitações de processamento do sistema visual, já que o aumento da sensibilidade é obtido à custa da redução da resolução. Este mecanismo biológico tem servido como fonte de inspiração para algoritmos de percepção baseados no conceito de foco de atenção. Em vez de se efectuar um processamento único numa representação de alta resolução, com custos bastante altos do ponto de vista computacional, o processamento é feito por etapas. Inicialmente determinam-se zonas de atenção usando algoritmos eficientes e mais grosseiros usando representações de baixa resolução ou sensores auxiliares. Posteriormente essas zonas são processadas de forma mais detalhada.

Em [7] é apresentada uma arquitectura para o seguimento de movimento em tempo real baseada numa hierarquia de algoritmos para o processamento de níveis incrementais de atenção. Em [8] é apresentado um método estatístico para determinar pontos que representem zonas de interesse numa imagem.

4.4. Objectivos
4.4. Objectives

O objectivo principal do trabalho consiste em desenvolver algoritmos que permitam a um sistema avançado de ajuda à condução e detectar e evitar perigos usando mecanismos de atenção.

A relevância e actualidade deste trabalho vai de encontro ao crescente interesse da indústria automóvel em potenciar as diversas tecnologias presentes nos veículos modernos através da criação de novos sistemas de segurança que permitam torná-los mais seguros. As técnicas a desenvolver deverão permitir ganhos de desempenho e robustez que irão contribuir positivamente para o avanço não só da área concreta de aplicação, mas também das ciências da percepção e visão artificial em geral.

Esta trabalho será implementado no AtlasCar, um veículo autónomo em desenvolvimento na Universidade de Aveiro que funciona como um laboratório móvel para estudos de condução autónoma e em sistemas avançados de apoio à condução [9].

4.5. Descrição detalhada
4.5. Detailed description

Os mecanismos de atenção permitem o desenvolvimento de sistemas sofisticados de percepção segundo uma hierarquia de densidade espacial e temporal da própria percepção. A análise contínua de grandes fluxos de informação, e a necessidade de a processar, pode não ser possível por restrições computacionais, mas também por questões de eficácia no acto de descortinar a importância de eventos ou estímulos devido a um eventual um excesso de detalhe que pode retirar contexto global e levar a contusões erróneas. Assim, a abordagem baseada em mecanismos de atenção será despoletada por percepção grosseira, periférica, de baixa resolução ou pouco definida na representação de eventos ou acções de agentes e, uma vez identificados os eventos candidatos à percepção detalhada, restrita no espaço, como é o caso da visão foveada, segue-se a fase de eleger os candidatos, por um qualquer critério de prioridades a definir durante o estudo, a essa percepção detalhada e seu consequente processamento.

O trabalho será desenvolvido em diversas tarefas das quais se prevêem as seguintes:

Tarefa 1-Definição dos mecanismos de atenção a explorar
Dada a panóplia de sensores disponíveis, e eventualmente outros a instalar no AtlasCAR, será necessário definir os relevantes para explorar as questões dos mecanismos de atenção. Poderão ser manchas de cor, movimento genérico a partir do fluxo de percepção (óptico, de espaço desimpedido, etc.), deformações de geometria 2D ou 3D, entre outros a considerar. Um forte candidato a mecanismo de atenção preferencial assenta na visão de multi-resolução porque permite diversas implementações desde câmaras multi-resolução (periférica, foveada) devidamente registadas em cima de dispositivos de orientação (Pan&Tilt units) até arrays de câmaras fixas de alta resolução com estratégias dedicadas de agulhagem dinâmica de qual a câmara activa em cada instante para permitir o processamento fino e preciso.

Tarefa 2-Formalização do problema e desenvolvimento de uma representação hierárquica de percepção
O problema da percepção e da sua gestão, em virtude da diversidade de dados, sensores e representações obrigará a uma clara hierarquização da informação. Essa informação variará desde os dados em bruto, às diversas integrações e fusões levadas a cabo por outros módulos cujo objectivo é dar significado e operar data reduction na abundância sensorial disponível, sem prejuízo de inclusivamente tirar partido do contexto e sequência temporal dos eventos.

Tarefa 3-Integração dos algoritmos de navegação na arquitectura do AtlasCAR.
O ambiente de desenvolvimento do AtlasCAR proporciona a coexistência de uma equipa multidisciplinar de

investigadores e para isso foi necessário adoptar uma arquitectura de software que o permita, mormente uma adequada intercomunicação entre processos e sistemas. Actualmente está em uso uma arquitectura CARMEN/IPC mas já começaram desenvolvimentos para a migração a breve prazo para o ROS. Seja como for, a tarefa de integração no sistema existente forçará a uma comunhão de métodos e estruturas.

Tarefa 4 – Implementação dos mecanismos de atenção
Escrita de software e seu teste no ambiente real para efectivamente implementar os mecanismos de atenção decididos nas tarefas anteriores.

Tarefa 5 - Algoritmos de percepção activa na sequência dos mecanismos de atenção
Escrita de software para tornar activa a percepção no sentido de dinamicamente se escolher os candidatos espaço-temporais de percepção mais adequados e canalizar para eles o processamento preferencial para a análise e confirmação da situação de risco. Esta tarefa completa a tarefa anterior e, em conjunto, implicar-se-ão desenvolvimentos na gestão de informação complexa e multi-dimensional. Também distribuída por estas duas tarefas estará sob atenção a questão da gestão de conflitos na concorrência ou simultâneidade de direcções de atenção que será preciso gerir de forma a assegurar o fluxo de dados e a assistência ao condutor para minimizar riscos de acidente. Por exemplo, em ambientes muito dinâmicos e com excesso de triggers de atenção, o sistema poderia recomendar redução de velocidade ou outras manobras que envolveriam algum planeamento.

Tarefa 6-Política de interacção com o condutor
Os resultados da análise do risco e perigos iminentes devem ser transmitidos ao destinatário (em geral o condutor) apenas quando o sistema tiver uma grande confiança sobre o estado da não consciência ou da desatenção do destinatário em relação a esse perigo. Por exemplo, se o campo de visão instantâneo do condutor estiver a cobrir a zona de risco iminente, a necessidade de o avisar será menor. Isso pode ser feito se se mantiver um sistema de observação do campo visual ou outros sinais biométricos do condutor que possam estar disponíveis. Portanto, o sistema de interacção com o condutor deve estar preparado para condicionar as suas saídas (ou eventualmente as suas acções num futuro próximo) em função da necessidade junto do destinatário.

Tarefa 7 – Escrita de relatórios, artigos e tese
Tarefa distribuída ao longo do período de trabalho. Esperam-se relatórios anuais, artigos em conferências e revistas internacionais.

Planeamento proposto para 36 meses.
Tarefa 0 – Programa curricular: ferramentas, estado da arte e projecto de tese – 12 meses
Tarefa 1 – Definição dos mecanismos de atenção - 6 meses
Tarefa 2 – Formalização do problema e estruturas de representação - 3 meses
Tarefa 3 – Integração no ambiente AtlasCAR - 3 meses
Tarefa 4 – Implementação e teste dos mecanismos de atenção - 3 meses
Tarefa 5 – Algoritmos de percepção activa - 3 meses
Tarefa 6 – Algoritmos de gestão e interacção com o condutor - 3 meses
Tarefa 7 – Escrita de artigos, relatórios e tese – 3 meses.

4.6. Anexos
4.6. Attachments

Nome Name	Tamanho Size
atlascar.jpg	141,77Kb

4.7. Referências
4.7. References

[1] Autoridade Nacional de Segurança Rodoviária, "Relatório Anual 2010"

[2] Ed Williams. "Airborne Collision Avoidance System", Australian Workshop on Safety Critical Systems and Software. pp. 97-110, 2004

[3] Guna Seetharaman, Arun Lakhotia, Erik Philip Blasch, "Unmanned Vehicles Come of Age: The DARPA Grand Challenge," Computer, pp. 26-29, December, 2006

[4] Alberto Broggi, Pietro Cerri, Mirko Felisa, and Maria Chiara Laghi, "The VisLab Intercontinental Autonomous Challenge: an Extensive Test for a Platoon of Intelligent Vehicles", Intl. Journal of Vehicle Autonomous Systems, special issue for 10th Anniversary, 2011, ISSN:1471-0226

[5] K. L. R. Talvala, K. Kritayakirana, and J. Christian Gerdes, 'Pushing the Limits: From Lanekeeping to Autonomous Racing', Annual Reviews in Control 35(1), pp. 137-148, 2011

[6] Distner M, Bengtsson MQ, Broberg T, Jakobsson L, "City Safety – A System Addressing Rear-End Collisions at Low Speeds", International Technical Conference on the Enhanced Safety of Vehicles 2009, 2009

[7] Kentaro Toyama, Gregory D. Hager, "Incremental Focus of Attentionfor Robust Vision-Based Tracking", International Journal of Computer Vision 35, 1999, pp. 45-63

[8] Niall Winters, José Santos-Victor, "Visual Attention-based Robot Navigation using Information Sampling", International Conference on Intelligent Robots and Systems

[9] Vitor Santos, Jorge Almeida, Emanuel Avila, et al, "ATLASCAR –Technologies for a Computer Assisted Driving System on board a Common Automobile". 13th International IEEE, Annual Conference on Intelligent Transportation Systems, Madeira Island, Portugal, September 19-22, 2010, pp. 1421-1427

Programa Doutoral em Engenharia Mecânica

Mechanisms of visual attention

Trabalho submetido no âmbito da disciplina Projecto de Tese I

Pedro Emanuel Marques Dias Pinto
Número Mecanográfico: 66200

Resumo

Os mecanismos de atenção são fundamentais para a percepção, pois permitem filtrar informação, seleccionando aquela que é mais relevante num dado contexto. Em tarefas com elevada exigencia computacional, como o processamento de visão em tempo real, a inspiração em mecanismos biológicos surge como uma alternativa ao varrimento de cada frame da imagem.

Neste trabalho tem como objectivo apresentar alguns dos principais e mais influentes modelos visuais de atenção. São ainda descritas as ferramentas disponíveis para a implementação e desenvolvimento destes algoritmos. Finalmente descrevem-se alguns conjuntos de dados adequados à avaliação do desempenho dos modelos.

Conclui-se neste trabalho que o desenvolvimento de mecanismos visuais de atenção admite abordagens bastante distintas e que apesar do vasto trabalho de investigação já existente, ainda existe bastante caminho a percorrer no desenvolvimento destes modelos.

Mechanisms of visual attention

Pedro Pinto

September 14, 2014

Contents

Chapter 1

Introduction

The human failure is a source for many traffic accidents. The performance of the driving task is greatly influenced by external factors such as distractions and fadigue. The aerospace industry has relied for many years on technology that assists pilots during flights. In recent years, road vehicles have benefited from the development of Advanced Driver Assistance Systems (ADAS) that are able to sense the environment and react preventively in the imminence of an accident.

For the human driver, sight is one of the most important senses and a rich source of information. The eyes capture the morphology of the road and detect any obstacle that appears in the way. Likewise, vision is also one of the most promising sensory modalities for ADAS. For several years, researchers have been developing intelligent object recognition systems that aim to locate and recognize hazards. The first commercial systems are already available in some top-of-the-line models (Figure 1.1), and it is expected that more models get equipped with these technologies in the near future.

Machine vision systems typically decompose this problem in two steps, hypotesis generation and verification. In the former, an image is segmented and a Region of Interest (ROI) with a potential target is determined. In a second step, a classification algorithm is applyed to validate or reject the hypotesis.

For the first step, a *brute force* approach is often enforced, by applying a sliding-window technique. This algorithm has high computational cost, as it implies an exhaustive search in the image space. This problem is severely aggravated with the usage of high definition cameras.

The nature has encountered this problem before during species evolution. Brains cannot process in real time all the sensory data they receive. The solution was the development of attention mechanisms that can quickly filter non-important data and focus the brain's limited processing power in potential interesting information.

Biology has been a source of inspiration for many technological solutions. Understanding how the brain works can be helpful to the design of efficient algorithms that can detect the most interesting regions of the image, a task that the animals appear to do effortlessly. The details of the cognitive processes are still largly unknown and are a topic of active research by different scientific fields. Attentional mechanisms have been addressed from different perspectives by both psychologists, neurologists and engineers. While psychologists and neurologists are focused in understanding and explaining nature, engineers are more interested in gaining insights that help them find solutions to existing problems.

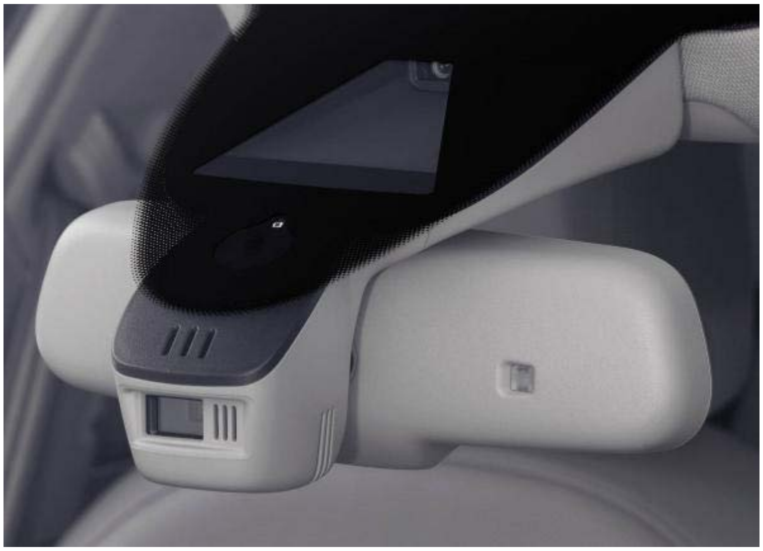

Figure 1.1: A modern ADAS with a camera integrated on the rear view mirror

Chapter 2

Attention

In 1890, William James defined attention in his seminal work *Principles of Psychology* [2] as ”taking possession by the mind, in clear and vivid form, of one out of what seem several simultaneously possible objects or trains of thought”.

Studies suggest that attention can be exogenous, if influenced by external stimuli, or endogenous, if guided by the person’s goals and context. In figure 2.1 the explorer is salient in the image thus he can be easily spotted. His red garments drive attention because of the contrasting color with the white of the landscape. Saliency is the property that enables one individual when free viewing a scene to focus on a specific part of the visual field. This stimulus driven perception is also called *bottom-up* attention.

On the other hand, when performing search tasks, the target characteristic features become more salient. The attention is also naturally shifted towards areas where the individual believes a target may be located. Goal and context therefore complement visual stimuli, in a *top-down* approach.

In 2007, Most and Astur studied how voluntary attention afected the ability to respond to unexpected events [3]. They used a driving simulator where participants were asked to search for specific road signs. During the test, at an intersection, a motorcycle would veer into the driver’s path. The test concluded that the collision rates were much higher when the motorcycle’s color was different from the color of the signs that were being searched.

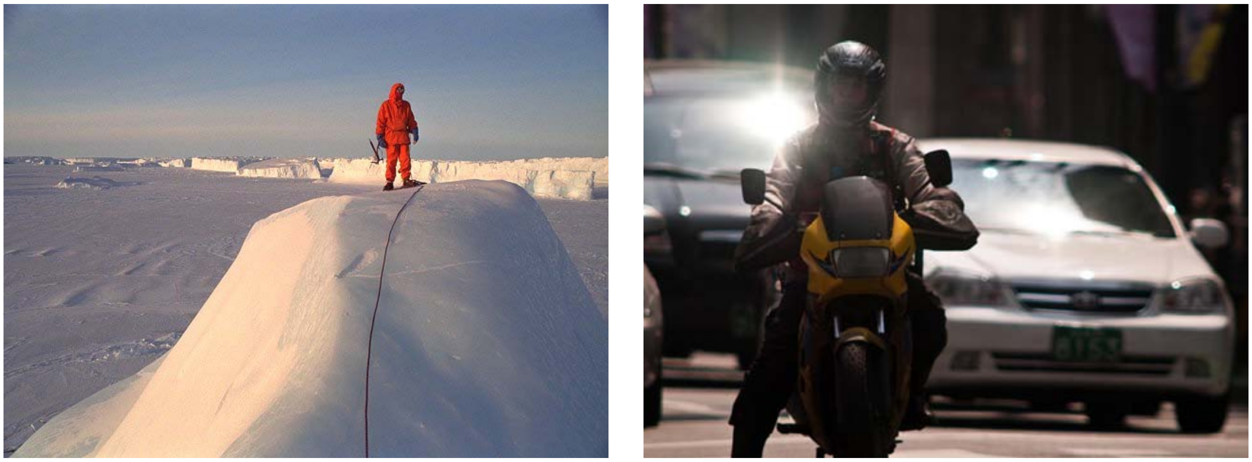

Figure 2.1: Left: The bright colored garments make the explorer stand out from the background. Right: When looking for blue signs, the yellow motorcycle is ignored.

Studies have also shown that performance varies greatly between feature search and conjunction search (Figure 2.2). In the first case the target differs from the distractors by just a single feature. One example is seaching for a red circle in a set of green circles. In the latter case, the target differs by more than one feature. For instance, searching for a red circle in a set of red rectangles and green circles. The fact that feature search is much faster then conjuction search suggests that there is a *pre-attentive* stage where basic features like color, orientation, motion and spatial frequency are first processed.

Vision is an active process in which perceived images are combined with stored knowledge to create an internal reconstruction of the visual world. Features are rearranged according the person's expectations and

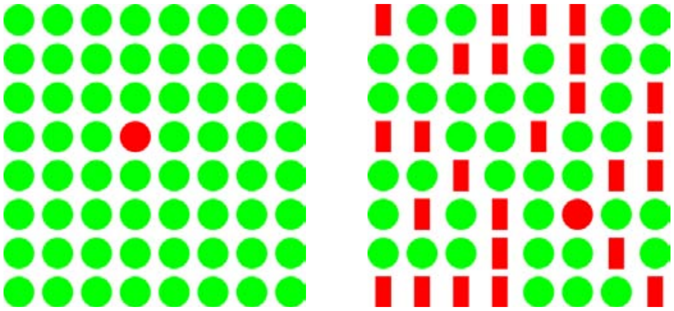

Figure 2.2: Left: Feature search. Right: Conjunction search.

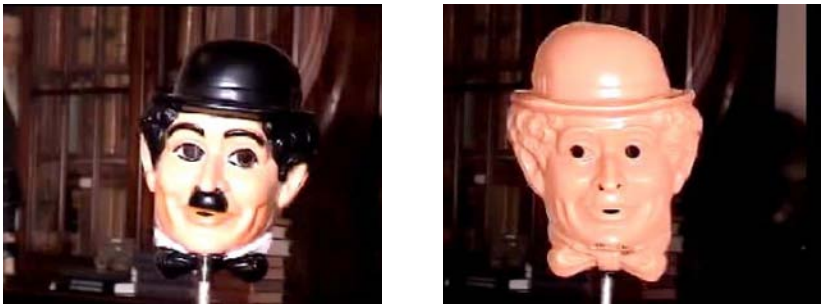

Figure 2.3: Hollow-face illusion. Charlie Chaplin mask on the left is convex. The inner side of the mask appears as convex but in reality is hollow.

past experiences. This is one of the causes for some kinds of optical illusions like the *Hollow-Face illusion* (Figure 2.3). In this optical illusion a concave surface appears as a convex face. This shows how top-down knowledge of faces interferes with bottom-up signalled information.

In 1980, Anne Treisman and Garry Gelade formulated the Feature Integration Theory (FIT) [4] which strongly influenced subsquent research on human visual attention. They proposed a model with two stages called pre-attentive and focused attention. In the first, basic features are registered early and in parallel across the visual field, resulting in multiple feature maps. At the second stage, these are bound into a master map and objects are recognised in a sequentional cognitive process.

A similar model called *Guided Search* was presented by Jeremy Wolfe in 1989. It had the same two stages as FIT but introduced the idea that the top-down information would boost basic features at the pre-attentive stage.

Chapter 3

Computational models of Attention

For computer vision applications, researchers are not simply interested in understanding the neurological mechanisms, but also how those principles can these be applied in order to create more efficient machines. They create computational attention systems which are biologicaly inspired computer implementations of atentional models. Their output can be either a saliency map or a trajectory of focused regions. Saliency maps are usually represented by intensity maps that show the amount of saliency or the probability of each pixel being salient.

There are different approaches to address saliency calculation. Some algorithms make use of local information where a pixel or patch is compared with its surroundings, while others calculate saliency over the global image.

Based on Treisman and Gelade ideas [4], Koch and Ullman developed a computational framework [5] that given an image would sequentially select foci of attention. In this model, a number of elementary features like color, orientation and direction of movement are initially represented in parallel in separated topographical maps also known as conspicuity maps. These early representation maps are usually calculated at multiple scales. The objective of selective attention is to fuse these conspicuity maps into a coherent master map called saliency map. The authors also mention the possibility of modulating the relative gain of the features according with higher order objectives. In Koch algorithm [5], the Focus of Attention is selected by a Winner

Takes All network. This is neuronal maximum finder that singles out the most conspicuous locations. The process will influence subsquent selections by reinforcing the neighbourhood (proximity) or characteristic features (similariy). This framework has originated several algorithms that share its principles.

One of such algorithms, was developed by Itty et al. who developed the first complete implementation of the framework and set a benchmark for which future works would be compared against [6]. Itti's model computes three basic features (color, intensity and orientation). The saliency map is created using a map normalization operator, which promotes maps with small number of strong peaks and suppresses maps with numerous identical peaks. Furthermore, innibition of return is achieved by iteralively eliminating the active salient region. In this model, saliency is the result of local feature contrast as inspired by the interaction among cells within primate visual cortex. Itti's model has extended over time; one such enhancement is the addition of top-down modulation [7].

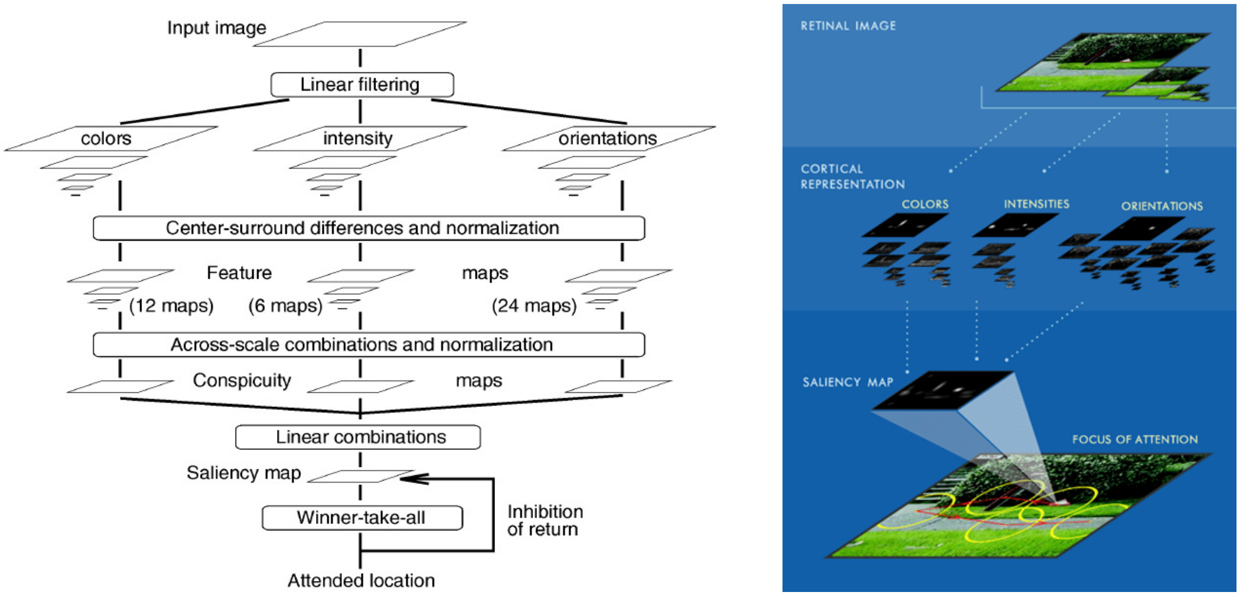

Figure 3.1: Original Itti and Koch model

A different approach to visual perception modeling emerged from information theory research and the usage of statistical tools. Fred Attneave realised that the natural visual stimuli is highly redundant and that the brain can predict missing information from the scene structure [8]. Later, Bruce and Tsotsus defined

Attention based on Information Maximization (AIM) [9]. They presented a bottom-up model where a salient region is the one that provides the most information (Figure 3.2).

They calculate Shannons self-information over 7x7 RGB images patches. In order to reduce the dimensionality problem, the authors perform ICA (Independent Component Analysis) on a dataset of natural images. Shannons self-information is given by the following equation where p(x) is the probability of a given pixel:

$$I(x) = -log(p(x)) \tag{3.1}$$

A biologically inspired neural circuitry is then used to compute the saliency based on this self information measure.

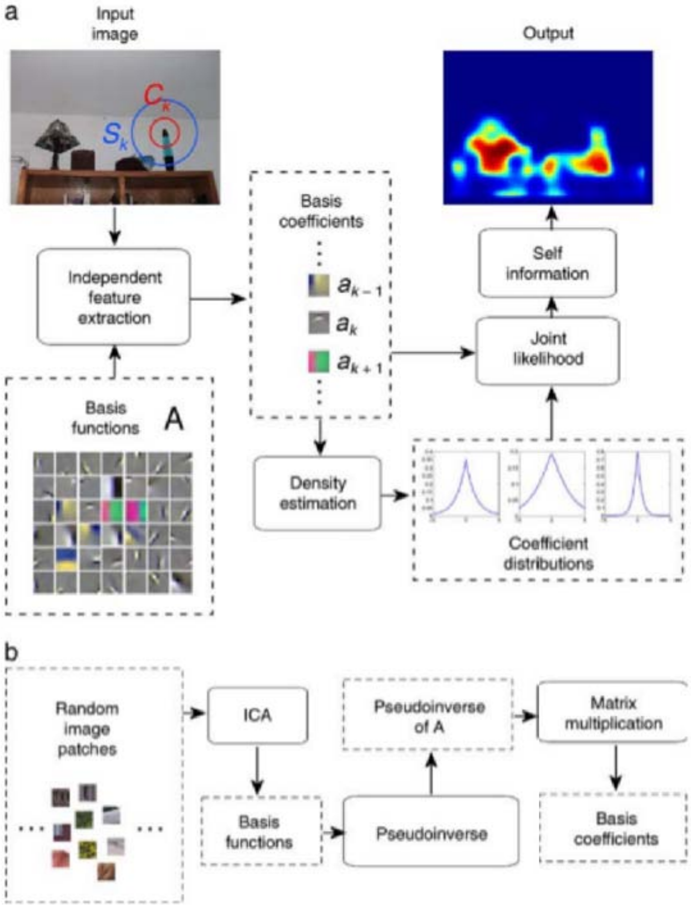

Figure 3.2: The Attention based on Information Maximization model

A similar saliency definition is given by Zhang's SUN bayesian framework [10]. This algorithm defines saliency as the probability of the target at each location, given a visual feature. The framework is able to incorporate top-down information with bottom-up saliency (equation 3.2). Top-down information is modeled as likelihood and location prior. The former is the probability of a feature given a certain target while the latter reflects the probability of the target given a location [11]. Bottom-up saliency is inversely proportional to the probability of the feature for a specific location. When there is no top-down information available (*free-viewing*), the saliency results from the self information at that point.

$$log\, s_z = \underbrace{-log\, p(F = f_z)}_{\text{Bottom-up saliency}} + \underbrace{log\, p(F = f_z|C = 1)}_{\text{Log likelihood}} + \underbrace{log\, p(C = 1|L = l_z)}_{\text{Location prior}} \tag{3.2}$$

In SUN, saliency is derived from natural image statistics, which is obtained by calculation the probability of a feature over a set of natural scenes. The probability distribution is estimated by fitting a gaussian curve for the observed values of each feature.

Rare 2012 [12] is the latest of a family of algorithms originated from Matei Mancas's work [13]. It is a pure bottom up attentional model based on multi-scale spatial rarity. The algorithm has three steps: feature extraction, multi-scale rarity and channels fusion.

In the first step RGB color is first decomposed using a PCA based color transformation. This results in separated luminance and chrominance channels. Each channel is then splitted in two pathways: low-level features and medium-level features. In the second pathway a gabor filters are applied with different orientations. In the second stage, rarity is calculated by calculating the cross-scale occurrence probability of each pixel. In the final stage the maps are fused into a saliency map.

A rather different approach is to process the image in the frequency domain as shown by Hou and Zhang in [14] . The basic ideia behing their method is that the visual system should be sensitive to features that deviate from the norm. They determined a recurrent pattern on the log-spectrum of natural images that can be aproximated by convoluting the log amplitude spectrum with an averaging kernel. In this algorithm, saliency results from the spectral residual of an image.

In [15], the authors show that the spectral residual of the amplitude spectrum is not essential to obtain

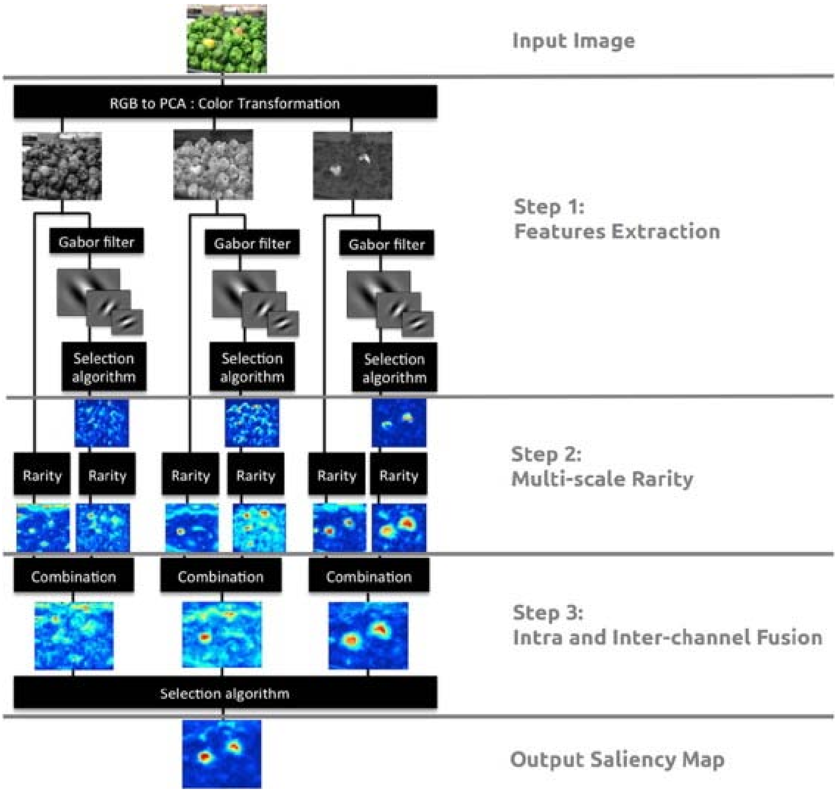

Figure 3.3: Rare 2012 algorithm

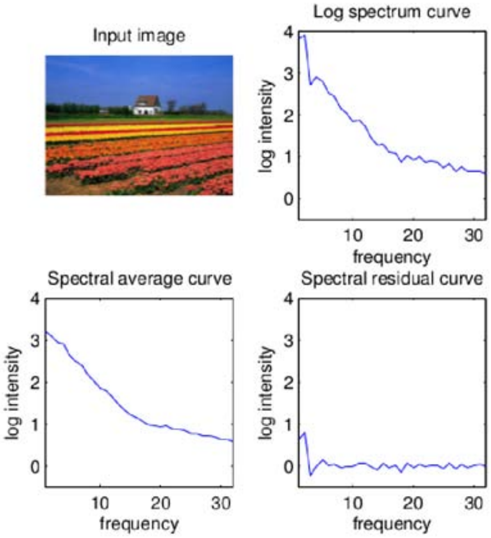

Figure 3.4: Spectral Residual algorithm

the saliency map and that this can be calculated using only the images Phase spectrum of Fourier Transform. More recently, Jian Li demostrated [16] that saliency could be determined by the convolution of the amplitude spectrum with a Gaussian kernel of an appropriate scale. He argued that spectral residual was just a special case and developed an improved generalized method that is able to detect both small and large salient regions. Furthermore, he used the Hypercomplex Fourier Transform to account for multiple features.

Chapter 4

Tools

This chapter describes some useful libraries and frameworks for implementing computational models of attention.

4.1 iNVT: iLab Neuromorphic Vision C++ Toolkit

The iNVT (pronounced *invent*) [17] is an open source toolkit developed at the University of Southern California that originated on the research work of Laurent Itti. It provides a modular framework for the development of new biologically inspired vision algorithms by mimicking the neurobiology of the primate brain. The term Neuromorphic refers to the fusion of computational neuroscience with more pragmatic engineering which aims to solve real-world problems. It also includes complete models such as Itti's model of bottom-up visual attention and of Bayesian surprise. Figure 4.1 shows a diagram of iNVT main modules. This toolkit has been successfully applied to robotics. In [18] the authors developed a robot localization system based on the scene gist refined by landmark points located using saliency.

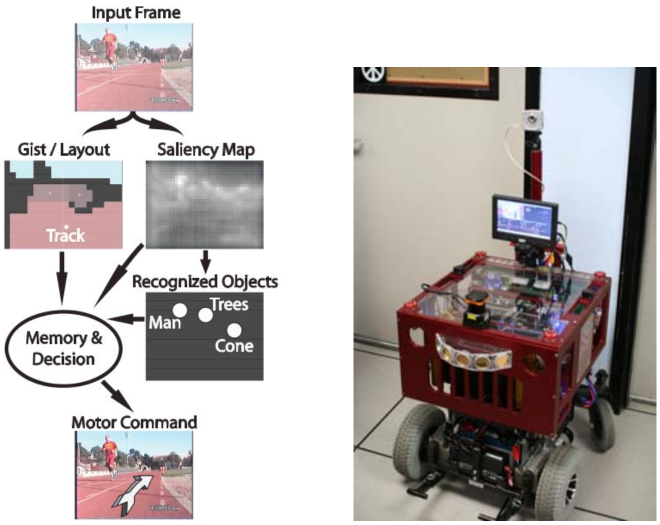

Figure 4.1: Left: iNVT toolkit components. Right: The Beobot 2.0, a robot using iNVT for localizing salient landmarks.

4.2 Saliency Toolbox

The Saliency toolbox [19] is an attempt to reimplement the iNVT core functionality in Matlab. The authors also extended Itti's algorithm to attend to proto-objects. These are defined as volatile units of visual information that can be bound into a coherent and stable object when accessed by focused attention. In other words, proto-objects are early representations of candidade objects.

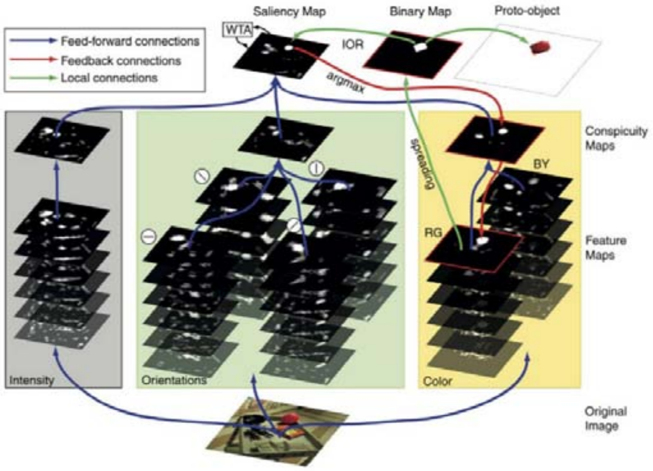

Figure 4.2: Extension of Itti's algorithm including proto-objects.

4.3 OpenCV

OpenCV [20] is a widely used open source computer vision library which targets real-time processing. It is written in C++, but also has bindings for other languages like Python and Java. It also supports several platforms like Windows, Linux, Android and Mac. OpenCV does not include any saliency algorithm although it contains several basic algorithms that can be combined in order to implement them. The library also contains a very useful Machine Learning module with several learning algorithms implemented.

4.4 Nick's Machine Perception Toolbox

The NMPT package [21] was developed by Nicholas Butko as a set of libraries for Machine Perception. It is build upon OpenCV and contains an implementation of FastSUN, an efficient algorithm based on SUN's, targeting real-time processing.

4.5 Spectral / Phase-based Visual Saliency

Boris Schauerte has made available several state-of-the-art spectral saliency algorithms [22]. These are Matlab implementations that can be found at the *File Exchange* service of *Matlab Central* website.

4.6 Authors implementations

Many researchers make their code available online in their own websites. The Attention Group of the NumediArt Institute [23] mantains in their website a fairly complete set of pointers to several saliency algorithms.

Chapter 5

Data sets

Evaluation of the saliency algorithms is typically done by comparing the results with other known algorithms from the literature like the popular Itti's model. Different papers have been using different images in varying conditions which makes it difficult to assess their true performance. With the increasing number of saliency algorithms it became necessary to find common benchmarks to compare and evaluate them. This is not a trivial task, as not all saliency algorithms are of the same kind and can be compared side by side. For instance, some are point based, aiming to predict image locations where people may fixate, while others are region based, addressing salient object detection and segmentation problems. Furthermore, top-down approaches may not perform well in *free-viewing* tasks.

Tilke Judd et all, from MIT collected a data set [24] containing 300 natural images with eye tracking data from 39 observers. These observers had been asked to *free-view* each image for 3 seconds. The motivation behind this data set is to provide a common platform that can be shared among researchers in visual attention modeling. Besides the images data set, the website enables the submission of algorithms to be evaluated online. Currently it has the performance figures of 18 saliency algorithms.

In [25], Borji et al. evaluate 5 datasets and compare 35 saliency detection models. The authors conclude that salient object detection models usually perform better than fixation prediction models. Also, scenes containing objects in textured and cluttered backgrounds revealed to be more challanging.

The PASCAL (Pattern Analysis, Statistical modeling, and Computational Learning) Visual Object Classes Challenge [26] is a widely used data set for object for benchmarking classification and recognition algorithms. It contains natural images with several object classes. Since region based saliency is similar to a segmentation problem, this data set can also be used to evaluate algorithms which search for salient objects. For each image it contains information about which pixels are part of objects and wich are part of the background.

Chapter 6

Conclusions and Future directions

This work presented an overview of current research on methods for driving attention to regions of interest in images. The classical Itty and Koch model continues to inspire current models, with over 2000 citations just in the last three years according with the Google Scholar statistics.

Biologically inspired methods try to measure saliency in the form of saliency maps. This is a challenging task because saliency is an ill defined concept without a precise definition. The motivations behind these methods also differ greatly among researchers. Some authors try to predict eye fixation locations and use as ground truth datasets of eye tracking data. Others use saliency as a way to efficiently detect objects in the scene. Yet others use saliency to improve object segmentation.

Furthermore, advances in Machine Learning has increased the interest for algorithms that model saliency by learning from datasets. In this area, Deep Learning algorithms promise the ability to learn hierarchical features from raw pixels. Methods like Convolutional Networks have been applied successfully to recognition tasks and are also starting to be applied to Saliency prediction. [27].

Bibliography

[1] E. Bruce Goldstein, editor. *Encyclopedia of Perception.* SAGE Publications, Inc., 2010.

[2] William James. *The principles of psychology.* H. Holt and company New York, 1890.

[3] Steven B. Most and Robert S. Astur. Feature-based attentional set as a cause of traffic accidents. *Visual Cognition*, 15(2):125–132, 2007.

[4] Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, January 1980.

[5] C. Koch and S Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. 1985.

[6] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(11):1254–1259, November 1998.

[7] V. Navalpakkam and L. Itti. Modeling the influence of task on attention. *Vision Research*, 45(2):205–231, Jan 2005.

[8] F. Attneave. Some informational aspects of visual perception. *Psychol Rev*, 61(3):183193, 1954.

[9] Neil D. B. Bruce and John K. Tsotsos. Saliency, attention, and visual search: An information theoretic approach. *Journal of Vision*, 9(3):1–24, 2009.

[10] Lingyun Zhang, Matthew H. Tong, Tim K. Marks, Honghao Shan, and Garrison W. Cottrell. Sun: A bayesian framework for saliency using natural statistics. *Journal of Vision*, 8(7), December 2008.

[11] R. P. N. Rao, B. A. Olshausen, and M. S. Lewicki. *Probabilistic models of the brain: Perception and neural function*. MIT Press, Cambridge, MA, 2002.

[12] Nicolas Riche, Matei Mancas, Matthieu Duvinage, Makiese Mibulumukini, Bernard Gosselin, and Thierry Dutoit. Rare2012: A multi-scale rarity-based saliency detection with its comparative statistical analysis. *Sig. Proc.: Image Comm.*, 28(6):642–658, 2013.

[13] Matei Mancas. *Computational Attention: Modelisation & Application to Audio and Image Processing*. PhD thesis, University of Mons, 2007.

[14] Xiaodi Hou and Liqing Zhang. Saliency detection: A spectral residual approach. In *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR07). IEEE Computer Society*, pages 1–8, 2007.

[15] Chenlei Guo, Qi Ma, and LiMing Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*. IEEE Computer Society, 2008.

[16] Jian Li, Martin D. Levine, Xiangjing An, Xin Xu, and Hangen He. Visual saliency based on scale-space analysis in the frequency domain. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(4):996–1010, 2013.

[17] L. Itti. The ilab neuromorphic vision c++ toolkit: Free tools for the next generation of vision algorithms. *The Neuromorphic Engineer*, 1(1):10, Mar 2004.

[18] C. Siagian and L. Itti. Biologically inspired mobile robot vision localization. *IEEE Transactions on Robotics*, 25(4):861–873, July 2009.

[19] Dirk Walther and Christof Koch. 2006 special issue: Modeling attention to salient proto-objects. *Neural Netw.*, 19(9):1395–1407, November 2006.

[20] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 1st edition, October 2008.

[21] Nicholas J. Butko, Lingyun Zhang, Garrison W. Cottrell, and Javier R. Movellan. Visual saliency model for robot cameras. In *ICRA*, pages 2398–2403. IEEE, 2008.

[22] Boris Schauerte and Rainer Stiefelhagen. Quaternion-based spectral saliency detection for eye fixation prediction. In *Proceedings of the 12th European Conference on Computer Vision (ECCV)*, Firenze, Italy, October 7-13 2012.

[23] Attention Group of NumediArt Insitute. Computational attention - saliency modeling and applications. `http://tcts.fpms.ac.be/~rocca/attention/`, 2013. [Online; accessed August-2013].

[24] Tilkea Judd, Fredo Durand, and Antonio Torralba. A benchmark of computational models of saliency to predict human fixations. Technical Report MIT-CSAIL-TR-2012-001, Massachusetts Institute of Technology, 2012.

[25] Ali Borji, Dicky N. Sihite, and Laurent Itti. Salient object detection: A benchmark. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV (2)*, volume 7573 of *Lecture Notes in Computer Science*, pages 414–429. Springer, 2012.

[26] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, jun 2010.

[27] M. Dorr E. Vig and D. Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. *Computer Vision and Pattern Recognition*, 2014.

[28] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

Efficient visual obstacle detection for ADAS

Pedro Pinto

September 29, 2015

1 Introduction

Autonomous Vehicles has become a hot research topic. As technology transitions from research labs to the production lines, the idea of self-driving cars is becoming more attainable each day. There are already in the market vehicles with sophisticated ADAS that provide some forms of semi-autonomous driving. For instance, the 2016 BWM 7 series provides a fully autonomous parking feature activated by the key fob. The new Mercedes E Class will be able to keep in its lane, drive itself through dark tunnels and make lane changes, at speeds up to 130 kilometers per hour. Tesla has also recently announced a software update for its current vehicles that will enable self steering and automatic lane change maneuvers at the press of the turn signal.

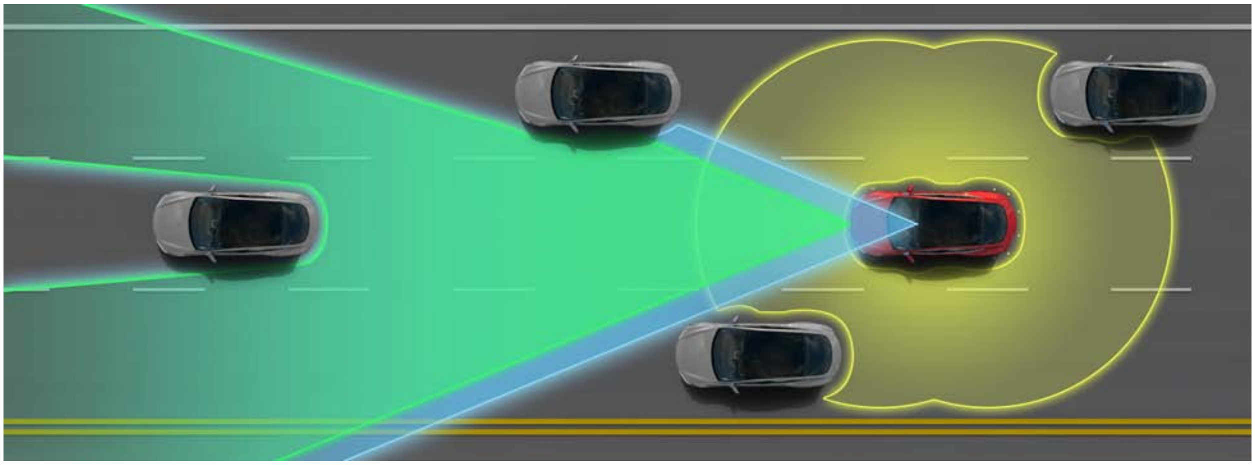

Figure 1: Tesla

On January 2015, Audi showcased an autonomous concept car that performed a long-distance test drive of 550 miles from Silicon Valley to Las Vegas. All the autonomous functions were managed by a compact central driver assistance control unit, called zFAS.

All these recent developments pave the way to the ultimate goal of a self-driving car. However, while the challenge of driving in structured environments like highways is considered solved, there is a lot of research to be made to reliably handle real world situations on unstructured environments.

2 Problem

Perception is a key aspect of any autonomous driving system. Many solutions use cameras and computer vision techniques to detect lanes, road signals and obstacles like pedestrians or other vehicles. New image

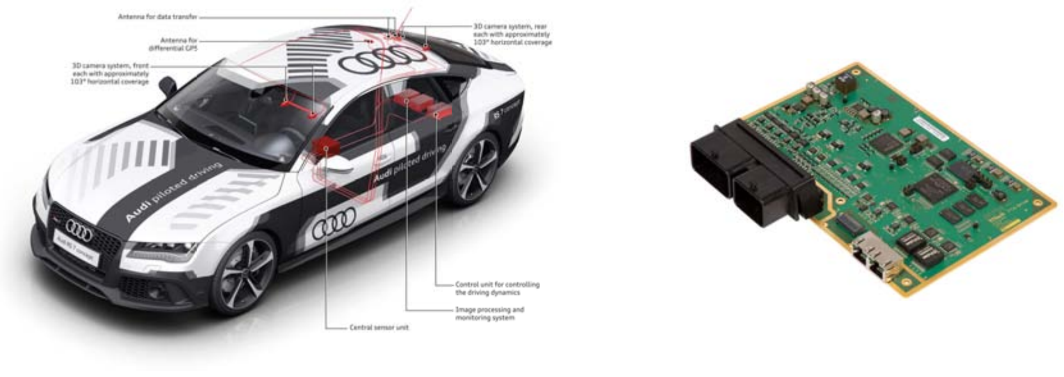
Antenna for data transfer
Antenna for differential GPS
3D camera system, front each with approximately 103° horizontal coverage
3D camera system, rear each with approximately 103° horizontal coverage
Control unit for controlling the driving dynamics
Image processing and monitoring system
Central sensor unit
Audi piloted driving

Figure 2: Left: Audi Piloted Driving concept car. Right: ADAS ECU

sensors have higher resolutions which allows to capture finer details. However, this comes at the cost of more data to be processed and therefore more computational power required. This is particularly relevant for ADAS and mobile robots because they need to operate in real-time. If an obstacle is detected too late, it may cause the vehicle to crash.

The most naïve form of object localization is called sliding window. The algorithm consists on an exhaustive search by matching or classifying a window in all possible image locations. This process can generate a huge number of candidate locations, specially if several scales are tested. This is very inefficient because most of these candidates usually do not contain any object.

For real-time applications, the algorithm speed is crucial. A significant performance speed-up can be achieved by optimizing the object search process.

3 From Saliency to Object detection

Nature through the evolutionary process has solved this problem using attention mechanisms. These select the most relevant perception data to process in order to cope with the brain limited processing resources.

The biology has inspired research and the development of models like the one proposed by Itty [1]. That visual attention system is a bottom-up approach where color, intensity and orientation features are combined into a single topographical saliency (conspicuity) map. A dynamical neural network then selects attended locations in order of decreasing saliency.

A lot of research has been done regarding saliency with different approaches and purposes. Some saliency models try to predict human eye fixations while others try to segment the most salient objects. What distinguishes the last approach from traditional segmentation research, is that the objective is not to partition the complete image in coherent regions, but only the most relevant parts.

Saliency is also not the same as object detection. Object detection consists in detecting instances of objects of a specific class in an image. Saliency, on the other hand, is agnostic to a particular class, but can guide the search process as part of an object detection algorithm.

4 Deep learning and Convolution Neural Networks

Deep learning is the application of hierarchical learning algorithms with multiple layers to represent increasingly complex concepts [2]. An image, for example, can be fed to the algorithm as an array of raw pixel values, then processed through several layers of learned feature detectors that can detect edges, object parts and even complete objects. In traditional computer vision, the features are usually hand crafted by a domain expert, sometimes by trial and error.

Deep learning is an old idea that was popularized by the work of Yann LeCun et al. [3] when he developed a Convolution Neural Network (CNNs) for handwritten zip codes recognition. LeCun showed that stochastic gradient descent was effective for training CNNs through back-propagation. After some years forgotten by the scientific community, in 2006, Geoffrey Hinton group developed pre-training techniques based on auto-encoders that attracted the interest of researchers to this type of algorithms [4]. In 2012, Krizhevsky et al. [5], obtained record breaking results on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) using a CNN. The dataset used consisted on 1.2 million high-resolution images with 1000 different classes. For efficiency, the algorithm was implemented on GPUs. These devices surpass CPU performance due to their massively parallel architectures that fit very well to CNNs. With the continuous development of GPU computing, it has become possible to train larger networks in less time. [6]

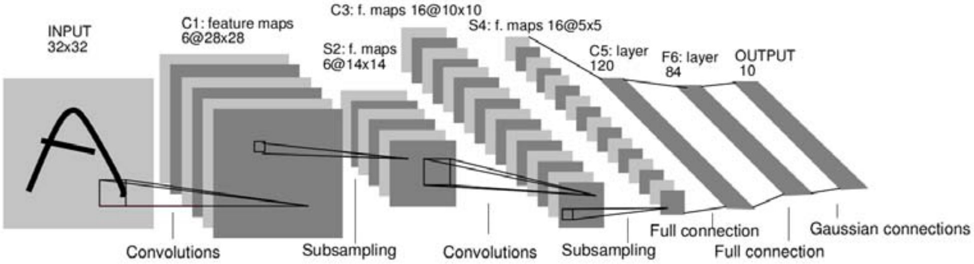

Figure 3: Convolution Neural Network

5 Convolutional Neural Networks and object detection

Krizhevsky results with ImageNet have shown the power of CNNs for visual recognition problems where a classifier assigns a label corresponding to the main object in an image. From classification problems, Deep Networks have started to be applied to object localization and detection. The former requires a bounding box around the predicted objected to be returned. Detection, on the other hand, is an even harder problem as an image may not contain any target object or it may contain several.

Saliency-based object proposal methods followed by post-classification using deep convolutional features is a common approach shared by most high quality object detection methods [7]. In addition to allowing for use of more sophisticated classifiers, the use of detection proposals alters the data distribution that the classifiers handle. This may also improve detection quality by reducing spurious false positives. One example of such algorithm is R-CNN [8].

Another approach is to pose the object localization problem as a regression problem like in [7].

5.1 OverFeat

OverFeat is an integrated framework that uses CNNs for recognition, localization and detection tasks [9]. Uses a classification scheme similar to Krizhevsky [5], extending it for multi-scale classification. After extracting features from the image, a sliding-window approach assures a fixed size input to the classifier. The sliding window is implemented in an optimized way to prevent redundant calculations. The classifier outputs a class and the respective confidence for each location. In addition, the algorithm includes a bounding box regressor for each class which predicts the location scale of the object in respect with each window.

Overlaping bounding boxes are merged and the final prediction is given by taking the ones with maximum class scores.

5.2 R-CNN

The R-CNN (Regions with CNN features) algorithm [8] combines a bottom-up category independent region proposal method with a CNN and a linear classifier. The region proposal methods, like selective seach [10], generate potential bounding boxes in an image. Each proposed box is then fed to CNN to extract a 4096-dimensional feature vector that is classified by a linear SVM. The CNN requires a fixed input size, it is therefore to wrap the region first. After classification, post-processing is used to refine the bounding box, eliminate duplicate detections, and re-score the box based on other objects in the scene. R-CNNs are computationally expensive in terms of space and time. An image can take 47 seconds to process [11], which is not suitable for real-time applications.

To overcome these limitations new algorithms have been published like SPPnet, and Fast R-CNNs [11].

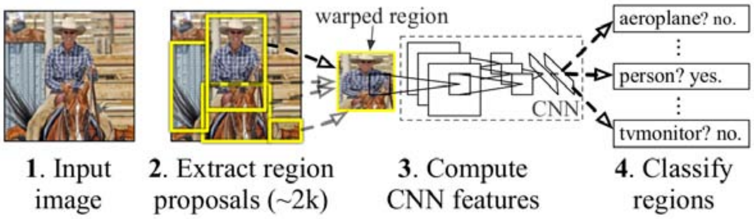

Figure 4: R-CNN algorithm

5.3 SPPnet

The R-CNN computations are very time-consuming as the algorithm requires that the deep convolutional networks be applied to thousands of warped regions per image. He et al. proposed SPPnet [12], a method that only requires one computation of the feature maps. By applying Spatial Pyramid Matching between the convolutional and the fully-connected layers, it is possible to generate a fixed-length output for training the classifiers that does not depend on the input size. Therefore it allows training with images of varying sizes and scales. Test results have shown that SPP-net can perform 24–102x faster then the R-CNN described previously.

When adapted for object detection, the algorithm also uses selective search, that provides about 2000 candidate windows per image.

Besides the spatial pyramid pooling, the SPPnet has many similarities with R-CNN. Its multi-stage pipeline consists on extracting features, fine-tuning a network, training an SVM classifier and fitting bounding-box regressors. However, unlinke R-CNN, it restricts the back-progapation of errors through the network during training, which can be considered a drawback.

6 Fast R-CNN

The previous approaches implemented multi-stage pipelines where classification an location refinement were sequential processes. The Fast R-CCN [11] is a simgle stage algorithm that trains networks in one fine-tuning stage that jointly optimizes a softmax classifier and bounding-box regressors, rather than training a softmax classifier, SVMs, and regressors in three separate stages.

Fast R-CNN takes as input a single-scale image (or an image pyramid) and a list of pre-computed object proposals to score. It outputs two output vectors per RoI, one soft-max probability estimation over the set of classes and per-class bounding box regression. The algorithm uses multi-task loss for training.

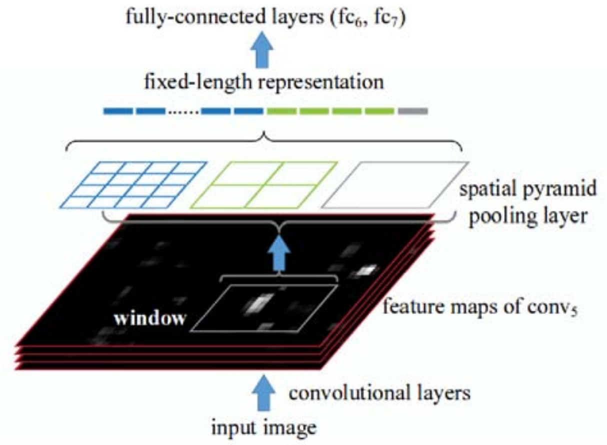
fully-connected layers (fc_6, fc_7)
fixed-length representation
spatial pyramid pooling layer
window
feature maps of $conv_5$
convolutional layers
input image

Figure 5: SPP-net

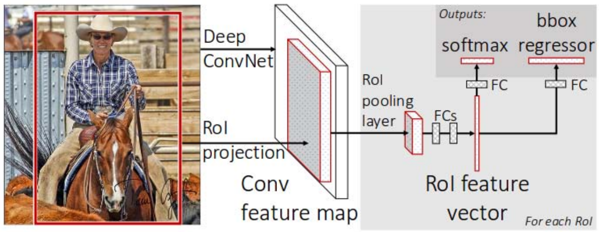

Figure 6: Fast R-CNN algorithm

7 Region Proposal methods

While the improvements on image recognition systems have been impressive, not much consideration was given to the real-time detection performance required for applications like ADAS.

Exhaustive search is computationally expensive, therefore many object detection metods use proposal generators algorithms to create a small set of candidates bonding boxes in order to allow efficient object detection by more complex classifiers. Some sliding window techniques use a coarse search grid and fixed aspect ratios where they apply fast weak classifiers as a pre-selection step. The number of proposals is usually a few thousands regions, which is a fraction of the total possible candidates of a typical sliding-window algorithm (100,000–1,000,000 windows).

This kind of algorithms needs to have high recall and efficiency. In other words, the method to be faster that the detector itself and all relevant areas must be proposed.

An interesting ideia for specific applications like ADAS, is to use context information, like in [13], to reduce the search space.

Most of current approaches can be classified as grouping methods or window scoring methods. [14]

7.1 Objectness

Objectness [15] is a measure that quantifies the likelihood of an image containing an object. It combines in a Bayesian framework several image cues, namely multi-scale saliency (spectral residual approach), color contrast, edge density and straddleness. Superpixels Straddling cue is higher for regions fitting tightly around an object.

The Objectness can be used for enhancing sliding window approaches, discarding not relevant bounding boxes.

7.2 Selective Search

Selective search [10] is the most common method in use. It takes advantage of the underlying image structure to generate class independent object locations. It relies on a graph-cut segmentation algorithm [16] to create initial regions from oversegmentation. Then it iteratively groups similar neighboring regions together. The method applies several merging strategies in order to be robust to different image conditions. It uses multiple colour spaces, starting regions and similarity measures like texture (gradient orientations).

Selective Search is relatively fast and was able to obtain better recall results then the previous described Objectness approach.

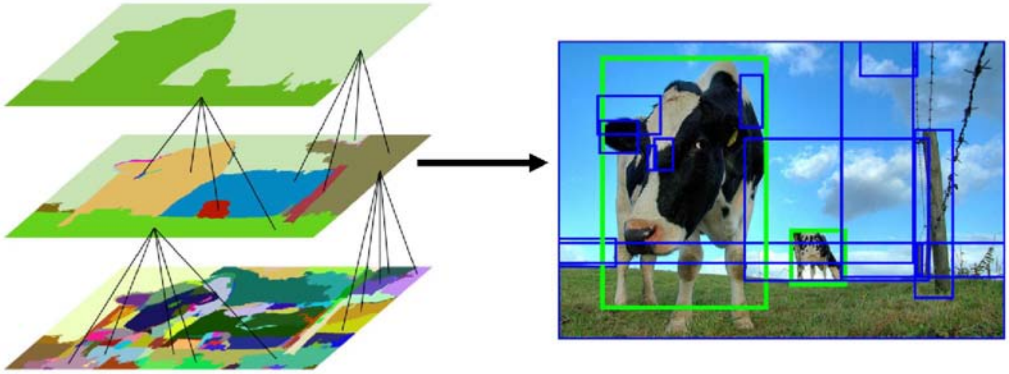

Figure 7: Selective Search algorithm

7.3 Edge Boxes

In [17], it is described a method to generate object bound boxes proposals using edges. It is based on the idea that the number of contours in a bounding box is indicative of the likelihood of the box containing an object. It uses a Structured Edge detector [18] to obtain the initial edge map. An "objectness" score measures the number of edges that exist in the box minus those that are members of contours that overlap the box's boundary. This algorithm is very fast, a near real-time variant can process an image in a tenth of a second.

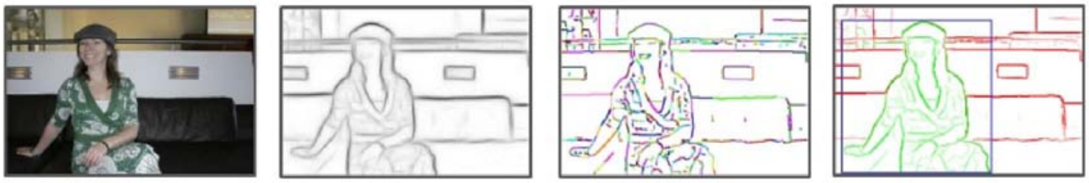

Figure 8: Edge Boxes algorithm

7.4 MultiBox

MultiBox [19] is a learning-based proposal generation method that defines object detection as a regression problem. The algorithms uses a Deep Neural Network to directly estimate a fixed number of bounding boxes coordinates. In addition, for each candidate proposal, the network also outputs a confidence score that represents the likelihood of containing an object. The objective function combines the confidence score with location similarity between the bounding box and ground truth. The performance closely matches Selective Search with less computational cost. For single class detection, the algorithm can be used as a fast, monolithic detector. However, for increased accuracy an additional post-classifier is required. On [7], MultiBox is combined with a context model and a post-classifier to provide better higher-quality results.

7.5 YOLO

YOLO [20] is also a learning-based proposal generation method that predicts bounding boxes and class probabilities directly from images, in an unified architecture. It uses a single Convolutional Neural Network that uses global image features, therefore taking context into account. An input image is divided in cells (7x7 grid). Each of this cells is responsible for estimating the probability of an object centered in that location, as well as its bounding box. This algorithm processes images in real-time at 45 frames per second, hundreds to thousands of times faster than existing detection systems. Its main limitation, however, is that it cannot predict multiple objects centered around the same cell. This poses problems when detecting small objects that appear in groups.

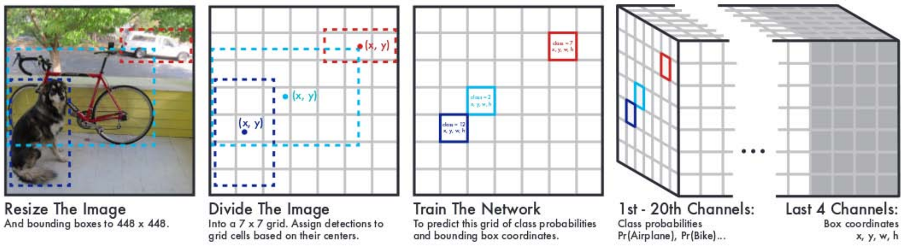

Figure 9: YOLO algorithm

8 Aplications to ADAS

Deep Learning has been successfully applied to some visual recognition tasks associated with ADAS like traffic sign recognition [21] and pedestrian detection [22].

8.1 NVIDIA Drive-PX

The NVIDIA DrivePX is a platform for developing Advanced Driver Assistance Systems. It is equipped with two Tegra X1 GPUs and 10 GB of DRAM memory. It also comes with libraries to assist in creating deep learning and computer vision algorithms.

On CES 2015, Mike Houston demonstrated an obstacle and signs detection algorithm running on real-time on the Drive-PX. It implemented a sliding-window approach with a simple network for candidate proposal and a modified version of the AlexNet [5] for recognition. It was able to detect several types of vehicles (i.e. SUV, truck, police car...), traffic signs, pedestrians, cyclists and crossroads. It was used a GoPro Hero 4 camera, however the algorithm only processed monochrome frames. In order to run in real-time the demo took advantage of the cuDNN, a library which implements a GPU optimized set of primitives for deep neural networks.

8.2 Mobileye EyeQ4

Mobileye is developing a new ASIC called EyeQ4 that consists of 14 computing cores and is capable of more than 2.5 teraflops. It is targeted for the next generation ADAS based on Deep Learning techniques. One

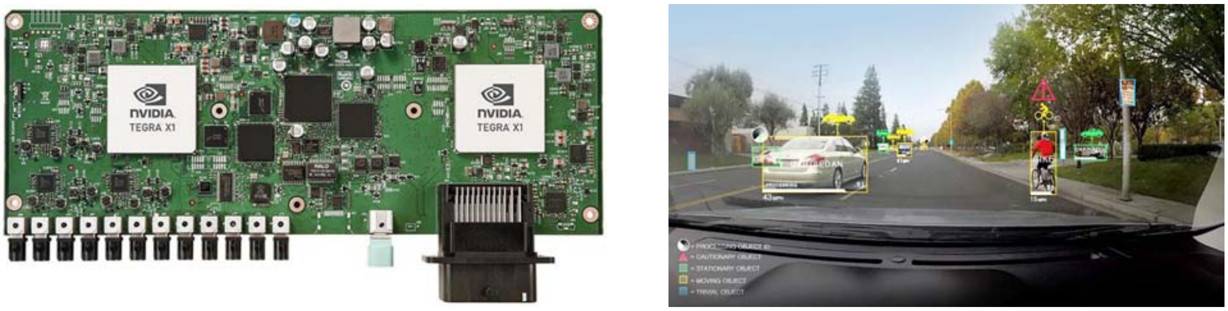

Figure 10: Left: NVIDIA Drive-PX. Right: Obstacle detection demo at CES 2015

example of such algorithms is real-time pixel level labeling for free space calculation.

9 Project

9.1 Objective

Inspired by NVIDIA Drive-PX demo at CES, the project consists on implementing an obstacle detection algorithm running in real-time on a NVIDIA Jetson. This kind of algorithms are normally used as part of Forward Collision Warning Systems.

9.2 Hardware

The NVIDIA Jetson TK1 development kit is a low cost platform for developing applications based on the embedded Tegra K1 processor. This System-on-a-chip has a Kepler GPU with 192 CUDA cores and a quad-core ARM Cortex A15. As with Drive-PX. NVIDIA also provides the cuDNN library to this platform, making it attractive for developing applications based on deep learning.

An Hungarian start-up called AdasWorks has demonstrated an autonomous vehicle with perception agorithms running on the Jetson board.

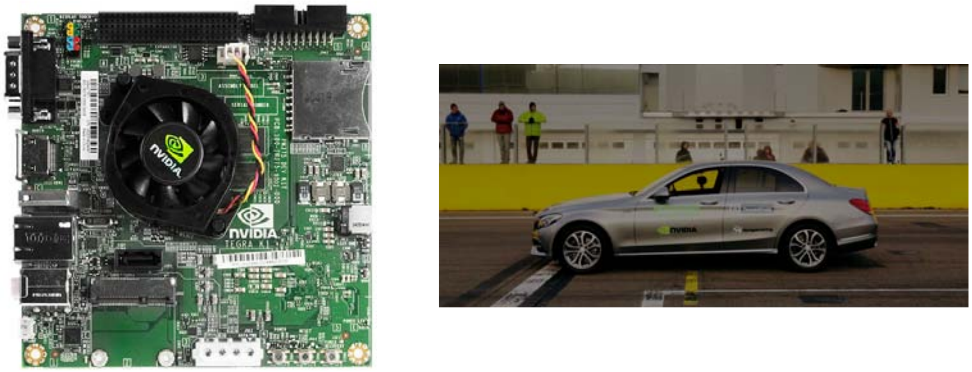

Figure 11: Left: NVIDIA Jetson TK1. Right: AdasWorks prototype

9.3 Tools

9.3.1 Berkeley Caffe

Caffe [23] is an open source deep learning framework created by Yangqing Jia during his PhD at UC Berkeley. It is curently developed by the Berkeley Vision and Learning Center (BVLC) and by an active community of contributors. The code is written in C++ with some optimizations written in CUDA to take advantage of the GPUs processing power. It includes bindings for Python and MATLAB, and integrates with NVIDIA cuDNN library. Caffe model definitions are written as configuration files that are later instantiated. The framework also includes reference models like the popular AlexNet [5]. Trained models can be downloaded from the model zoo, which is a community maintained repository for sharing models.

9.3.2 OpenCV

OpenCV [24] is an open source framework for developing real-time computer vision applications. It started in 1998 as a research project at Intel and it is currently supported by the non-profit OpenCV.org foundation. The framework implements a fairly complete set of classic and state-of-the-art computer vision algorithms. It is written in C++ but also has bindings for other languages. It has been ported for several different platforms, including the NVIDIA Jetson.

9.4 KITTI dataset

KITTI [25] is a benchmark for assessing computer vision algorithms. The dataset was collected with Annieway, an autonomous driving platform that has previously participated on the DARPA Grand Challange [26]. The vehicle collected data in different environments, including urban, rural areas and on highway.

The benchmark has different sets of data for different tasks, namely, stereo, optical flow, visual odometry, 3D object detection and 3D tracking. Having the data publicly available, enables anyone to perform research on autonomous vehicles perception without having to have access to a vehicle. On the other hand, the common dataset makes it possible to compare and rank different algorithms and solutions.

Of special interest for this work, is the object detection and object orientation estimation benchmark which consists of 7481 training images and 7518 test images. The dataset has a total of 80.256 labeled objects. There are different difficulty levels (easy, moderate and hard) with different minimum bounding boxes sizes and occlusion levels.

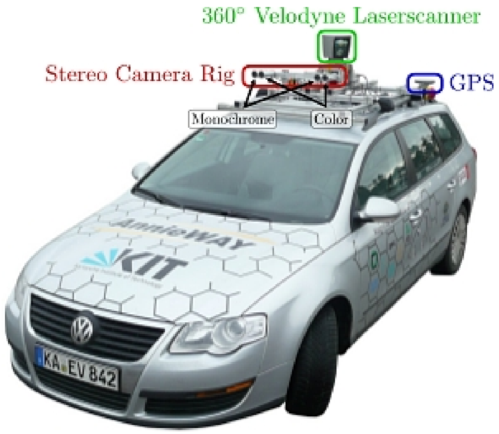

Figure 12: Vehicle used to collect KITTI dataset

References

[1] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(11):1254–1259, November 1998.

[2] Yoshua Bengio Yann LeCun and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, 2015.

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998.

[4] Geoffrey E. Hinton and Simon Osindero. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:2006, 2006.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[6] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014.

[7] Christian Szegedy, Scott Reed, Dumitru Erhan, and Dragomir Anguelov. Scalable, high-quality object detection. *CoRR*, abs/1412.1441, 2014.

[8] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524, 2013.

[9] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013.

[10] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.

[11] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *CoRR*, abs/1406.4729, 2014.

[13] David Held, Jesse Levinson, and Sebastian Thrun. A probabilistic framework for car detection in images using context and scale. In *IEEE International Conference on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA*, pages 1628–1634, 2012.

[14] Jan Hendrik Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *CoRR*, abs/1502.05082, 2015.

[15] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. What is an object? In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pages 73–80, 2010.

[16] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[17] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*. European Conference on Computer Vision, September 2014.

[18] Piotr Dollár and C. Lawrence Zitnick. Fast edge detection using structured forests. *CoRR*, abs/1406.5549, 2014.

[19] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. *CoRR*, abs/1312.2249, 2013.

[20] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015.

[21] Pierre Sermanet and Yann LeCun. Traffic sign recognition with multi-scale convolutional networks. In *The 2011 International Joint Conference on Neural Networks, IJCNN 2011, San Jose, California, USA, July 31 - August 5, 2011*, pages 2809–2813, 2011.

[22] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR'13)*. IEEE, 2013.

[23] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[24] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library.* O'Reilly Media, 1st edition, October 2008.

[25] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[26] Sören Kammel, Julius Ziegler, Benjamin Pitzer, Moritz Werling, Tobias Gindele, Daniel Jagszent, Joachim Schröder, Michael Thuy, Matthias Goebl, Felix von Hundelshausen, Oliver Pink, Christian Frese, and Christoph Stiller. Team annieway's autonomous system for the DARPA urban challenge 2007. In *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic, George Air Force Base, Victorville, California, USA*, pages 359–391, 2009.

universidade de aveiro

Programa Doutoral em Engenharia Mecânica

Disciplina	Sistemas Avançados de Automação	**Ano Lectivo**	2011/2012
Aluno	Pedro Emanuel Marques Dias Pinto	**Número mecanográfico**	66200

Trabalho realizado

No âmbito da disciplina Sistemas Avançados de Automação, o aluno realizou um trabalho de investigação na área de reconhecimento de padrões que consistiu no estudo, implementação e avaliação de um algoritmo de reconhecimento visual de veículos.

O algoritmo baseia-se no método "bag of features" no qual um conjunto de características de uma região de interesse são quantizadas com base num dicionário. O histograma resultante deste processo serve posteriormente de entrada a um classificador SVM que determina a existência ou não de um veículo.

A orientação deste trabalho contou com a colaboração do Professor Doutor Vitor Santos (Departamento de Engenharia Mecânica) e Professora Doutora Ana Tomé (Departamento de Electrónica, Telecomunicações e Informática).

Deste estudo resultou um artigo científico que será submetido à conferência ICAS 2013 (International Conference on Autonomic and Autonomous Systems) e que se anexa a este relatório.

Aveiro, 23 de Outubro de 2012

Visual classification of vehicles using a bag-of-features approach

Pedro Pinto[1], Ana Tomé[2], Vitor Santos[3]
DEM[1,3], DETI[2]
University of Aveiro
Aveiro, Portugal
pemdp@ua.pt[1], ana@ua.pt[2], vitor@ua.pt[3]

***Abstract*—This paper presents and evaluates the performance of a method for vehicle recognition in ROIs using a bag-of-features methodology. The algorithm combines SURF features with a SVM learning algorithm. An optimization to the bag-of-features dictionary based on genetic algorithm for attribute selection is also described and analysed. The results obtained show that this method can successfully address the problem of vehicle classification.**

***Keywords*-Intelligent vehicles; Object recognition; Machine learning algorithms; Support vector machines; Genetic algorithms;**

I. Introduction

The car manufacturers have increasingly been adopting more Advanced Driver Assistance Systems (ADAS) in order to make their vehicles safer. Some of these aim to support the driver by identifying and alerting of potential hazards in the road. The development of vision based methods for efficient obstacle detection and classification is one of the current research trends towards this goal.

The problem of detecting a car in an image can be divided in two steps, segmentation and classification. In the former, a Region of Interest (ROI) with a potential target is determined. Multiple methods can be used like sliding-window, saliency detection or an external laser as seed. In the second step image features are extracted from the ROI and a supervised learning algorithm determines if a vehicle is present.

This paper focuses on this last stage, and presents a method for vehicle classification which follows a bag-of-features approach combining SURF features with a Support Vector Machine for classification. The dataset used includes segmented images of vehicles and was collected by Miguel Oliveira on a previous research project [1].

II. Algorithm

Bag-of-features [2] [3] is a popular approach for classification due to its simplicity and performance. It uses the frequency of descriptors rather then spacial features to describe and classify an image. This method is inspired by document classification methods where word frequency is often preferred in detriment of semantic meaning [4]. Bag-of-features follows the same idea, using a dictionary of visual words that are image features.

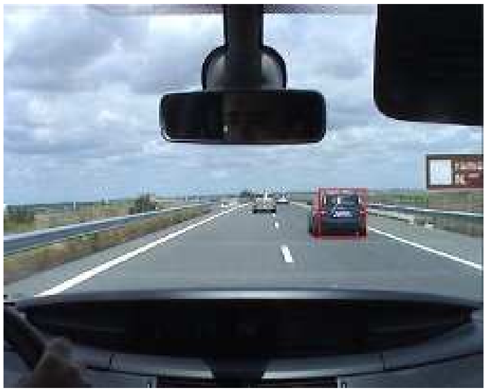

Figure 1. Image sample from the dataset

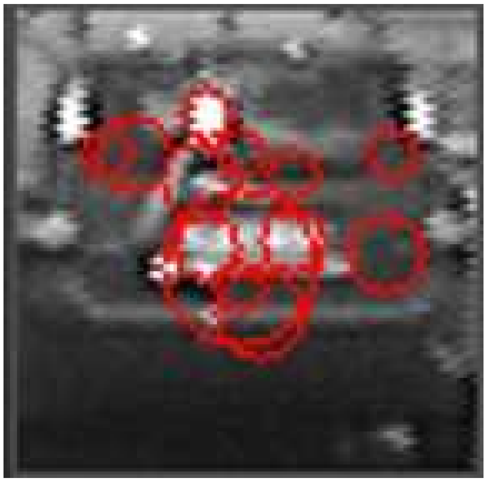

Figure 2. Features extraction of a ROI

Before feature extraction the ROI was pre-processed by converting to grayscale and its histogram equalized to enhance the contrast.

The visual vocabulary is based on Speed-Up Robust Features (SURF) [5]. SURF shares some similarities with SIFT combining an interest point detector with a descriptor that is scale and rotation invariant. However, because it is designed for performance, it is faster and more suitable for a real-time environment.

A. Dictionary

The dictionary is a set of visual features. These are computed around points of interest and consist of a histogram of local oriented gradients around the interest point and stores the bins in a 128-dimensional vector (8 orientation bins for each of the 4 x 4 location bins). The points of interest are detected by approximating the scale-normalized determinant of the hessian. This is a scale invariant blob detector characterized by good computational performance and accuracy.

Because the feature space is continuous, it is not feasible to generate all possible combinations using a brute-force approach. The solution was to sample features from a dataset that included images with and without vehicles. The dictionary is then formed by clustering the Visual features using K-means, the centroids of each clusters are the entries of the dictionary. The use of a clustering algorithm reduces the variable number of features to a more compact fixed sized set. Naturally the number of clusters is a design parameter of the system and it is experimentally evaluated in this work.

B. Classification features

In bag-of-words classification strategies the features at the input of the classifier are achieved by post-processing the basic features (semantic features) detected on documents [2]. In this case the dictionary provides the means to compute this high-level features. Then by characterizing each ROI with a set of visual features and after matching them in the dictionary an histogram of the frequencies of the entries of the dictionary is constructed. The matching of a SURF feature vector to the dictionary is achieved by using the K-NN (K Nearest neighbor strategy) using Euclidean distance measures. In this case K is equal to 1 which corresponds to approximating to the visual word with the lowest distance. Furthermore notice that the classification feature vector will have now the size of the dictionary overcoming then the problem of having different number of points of interest.

This entries of the classification vector were normalized by dividing the absolute frequencies by the dictionary size. Since the vector components are all of the same magnitude, normalization should not be a relevant factor.

Naturally there is another tradeoff that needs to be addressed, increasing the size of the dictionary the number of classification features increases increasing the complexity of the classifier. Several dictionaries sizes were tried in order to determine an suitable value.

C. SVM

Image classification is done using a SVM (Support Vector Machine) [6]. This is a supervised learning algorithm that, given a set of training examples, is able to create a model that can classify new examples.

SVMs support non linear classification using the "kernel trick". This designation stands for the dot product in a high-dimensional space without explicitly mapping of the original data. Then a kernel function defined with the original provides the means to work in the new space as in RBF (Radial Basis Function) kernel where σ is a user-defined parameter (equation 1). This is the function that maps the input space in other of higher dimension where a linear separation can be made.

$$k(x_1, x_j) = \Phi^T(x_i)\Phi(x_j) = exp(-0.5||x_i - x_j||^2/\sigma) \quad (1)$$

Linear SVMs are less computationally demanding, at least after the training phase, because the coefficients w that determine the separation can be stored while in the non-linear kernels the subset of the training data (the so called support vectors) need to be stored to perform the decision on every new data example x. The decision is taken computing equation 2.

$$\sum_i^{N_s} \lambda_i y_i \Phi^T(x_i)\Phi(x) + b = \begin{matrix} > 0 & \text{class 1} \\ < 0 & \text{class 2} \end{matrix} \qquad (2)$$

Where N_s is the number of support vectors x_i and naturally the pair (x_i, y_i) belongs to this subset of the training set and $y_i = 1, -1$ represents the label values. Finally λ_i and b are parameters learned, in conjunction with the selection of the support vectors, also during the training phase. In particular case of the linear SVM, where there is no mapping, the previous equation simplifies by computing $\sum_i^{N_s} y_i \lambda_i x_i$ The decision is now $w^T x + b$, the the support vectors do not need to be available. This is an important issue for on-line applications

The SVM was trained using as input the frequencies of the dictionary's features. ROIs with cars should have higher frequencies of certain features not present in other images.

Each dictionary item is an attribute which means the SVM will have as many dimensions as the dictionary size.

D. Optimization

In a second step, an optimization was introduced in order to improve the performance and robustness of the learning algorithm. It was performed an attribute selection optimization based on a genetic algorithm.

Genetic algorithms are biologically inspired search methods. A population of chromosomes is submitted to an evolutionary process for several generations in order to improve a fitness metric. Each chromosome is formed by a string of binary genes that can recombine with others (crossover) or suffer random changes (mutation). The probability of crossover increases according with the chromosome fitness, directing the search towards the goal. Mutation is a way of introducing diversity and avoiding a fast convergence for a local minimum.

The dictionary was optimized by generating a 1024 feature set using the K-means clustering algorithm. In the genetic algorithm each chromosome represented a mask of enabled visual words. The result of the optimization is therefore a combination of visual features from the initial 1024 feature set. Attribute selection helps to tune the dictionary so that only the most relevant features are included.

III. Results

A. Implementation

This algorithm was implemented in C++ using the OpenCV library [7]. OpenCV is a widely used open source library for real-time computer vision. It focus on image processing but also includes machine learning algorithms, like the K-NN or SVM used in this implementation.

B. Dataset

The algorithm was validated with a dataset of images collected from a camera on board a vehicle. 2265 images: 828 with at least one car and 1434 with no car. For the first group a manual segmentation has been performed [1] which provides the coordinates of regions of interest (ROI) to be further processed (figure 1). In the case of regions of interest without cars a random squared ROI with a size between 50 and 200 pixels is considered.

The dataset was divided in three subsets: a training dataset (70%), a validation dataset (20%) and a test dataset (10%). The training dataset is used to generate the predictive model. The validation dataset is used to perform the attribute selection optimization. Finally the test dataset is used to evaluate the algorithm performance.

Figure 2 shows examples of interest points extracted using SURF from a ROI. These are identified by red circles.

C. Algorithm performance metrics

Precision and recall are best visualized using a confusion matrix. In this table each column represents predicted instances while each row represents actual instances. Precision reflects the number of correctly identified ROIs relative to the total amount classified as having a vehicle. Recall is the percentage of correctly identified positive ROIs from the universe of the total ROIs with vehicles.

To measure the algorithm performance it is not suitable to use performance or recall alone, as this could lead to erroneous results. As an example, an algorithm that classifies all the ROIs as positive has a high recall but a low precision. On the other hand, an algorithm that classified correctly one ROI and all the others as negative would have a high precision but a low recall. The solution is to use F-score which combines both metrics as defined by equation 3.

$$F - score = 2\frac{precision.recall}{precision + recall} \tag{3}$$

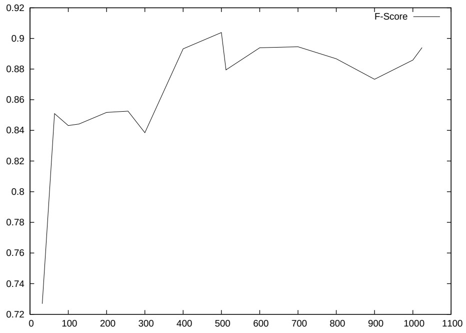

Figure 3. Algorithm performance relative to the dictionary size (without genetic optimization)

D. Dictionary size

Determining an adequate size for the dictionary is important as a very small dictionary may not be expressive enough and a very large dictionary not only consumes more computer resources but can also introduce learning problems because of the high number of dimensions.

Several dictionary sizes were tested whithin the range of 32 to 1024 visual words. Figure 3 shows that performance increases with dictionary size until a value of about 100 visual words. After that, the algorithm performance does not improve significantly.

E. SVM kernels

SVMs support non linear classification using the ”kernel trick”. In this study, both linear and a RBF kernels were evaluated.

1) Linear kernel: In the linear kernel dot product is done in input space. The linear kernel is defined by equation 4.

$$K(x_i, x_j) = x_i^T x_j \tag{4}$$

Tables I and II show the results obtained for the test dataset using a 128 words dictionary.

Table I
Confusion matrix for the test dataset using a linear kernel

	Predicted positives	Predicted negatives
Actual positives	70	12
Actual negatives	22	39

Table II
Performance measurements of the test dataset using a linear kernel

Precision	0,760870
Recall	0,853659
F-score	0,804598
Time (seconds)	0,02

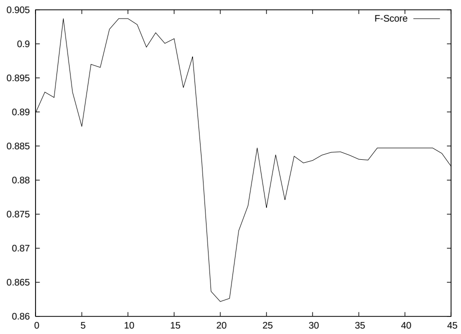

Figure 4. Evolution of the algorithm performance during genetic optimization

2) RBF kernel: RBF kernel maps the input space into a space of higher dimension using a gaussian function.

Tables III and IV how the results obtained for the test dataset using a 128 words dictionary. The results show that RBF kernel performs better then the linear kernel.

Table III
CONFUSION MATRIX FOR THE TEST DATASET USING A RBF KERNEL

	Predicted positives	Predicted negatives
Actual positives	66	16
Actual negatives	11	50

Table IV
PERFORMANCE MEASUREMENTS OF THE TEST DATASET USING A RBF KERNEL

Precision	0,857143
Recall	0,804878
F-score	0,830189
Time (seconds)	0,01

F. Dictionary optimization

To further improve the results, attribute selection was applied to the dictionary using a genetic algorithm. Each chromosome was implemented as a mask where each gene enabled or disabled a visual word. The fitness metric used was the F-score. Figure 4 shows the evolution of the average F-score of each generation. The maximum F-score was (0.903703) and was obtained during the initial cycles. The algorithm converged to a lower F-score.

IV. CONCLUSION

This study focused on studying the applicability of a bag-of-features methodology to the problem of vehicle recognition.

The experiments have shown that the proposed algorithm is able to successfully classify vehicles with performance metrics above 80%. On the other hand, the tuning of the dictionary using attribute selection did not produce the desired results. This can be explained by the fact that a genetic algorithm can take a high number of iterations until reaching an optimized genotype.

It was also shown that the choice of the kernel in the SVM learning algorithm is relevant for the performance of the algorithm. The RBF kernel obtained better figures than the linear kernel.

Combining SURF with an SVM proved to be a good choice. In the future different types of features should be evaluated and compared.

REFERENCES

[1] M. A. R. de Oliveira, "Development of a foveated vision system for the tracking of mobile targets in dynamic environments. msc. thesis." 2007.

[2] L. Fei-Fei, R. Fergus, and A. Torralba, "Recognizing and learning object categories, cvpr 2007 short course," Cambridge, MA, 2007.

[3] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha, "Real-time bag of words, approximately," in *In Proc. ACM Int'l Conf. Image and Video Retrieval*, 2009.

[4] Y. Zhang, R. Jin, and Z.-H. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, pp. 43–52, 2010.

[5] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *In ECCV*, 2006, pp. 404–417.

[6] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.

[7] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. Cambridge, MA: O'Reilly, 2008.