University of Southampton
Faculty of Physical Sciences and Engineering
School of Electronics and Computer Science

Title of my MSc. Dissertation

by

Li Sun

September 2016

A dissertation submitted in partial fulfilment of the degree of

MSc. Artificial Intelligence

Abstract

Text of the Abstract.

Acknowledgements

I would like to express (whatever feelings I have) to:

- My supervisor
- My second supervisor
- Other researchers
- My family and friends

Contents

List of Figures

1 | Introduction

Human ear, as a very promising biometric identifier has drew lots of attentions in biometric communities recently [6]. It contains sufficient curved structures which will not change radically from age 8-70 years old and are unaffected by cosmetics, unlike face [7]. Compare to gait, it cannot be easily changed intentionally by the subject when doing a surveillance. Which makes it a perfect and promising biometric for access control, security and video surveillance.

Recently, people trends to focus on using 3D data or combine 2D with 3D data due to it can overcome the illumination and pose variation of ear recognition, although it can get much better result now, special equipment and expensive computation will be needed [8]. However, this manuscript will concentrate on 2D images, because of the consistency when deploy in surveillance or other planar image scenarios [9].

1.1 Motivation and Objectives

The main motivation for ear recognition would be the implementation of an algorithm which can automatically tell the identity of a person via only taking pictures of his ear. Such a system can then be used for security surveillance, collect crime evidences and attendance check. Combined with the state-of-art face recognition system, they can provide more accurate identification rate and be more reliable.

For an automatic ear recognition system, it contains three main parts: ear detection, feature extraction and identification (classification). Due to the limitation of time in this project, it is better to focus on one part out of all three. Therefore, this project will concentrate on the first part **ear detection** which has very significant impact to the

following procedures. For example, if the ear failed to be registered inside an image, the following steps tend to be meaningless and may cause incorrect detection rate data.

Therefore, the objective of this project can be listed below:

- Implement an algorithm which has good detection rate on ear images under some circumstances such as occlusion and noises which often occur in the real world scenario.
- Attempt to use state-of-art deep-learning convolutional neural network method instead of classic hand-craft feature spaces.
- Reduce the time for the detection so that it can be used as a live detection.

1.2 Outline

This manuscript is divided into 4 chapters which mainly covered all the informations required. Chapter 1 is the introduction of the whole picture followed by the literature review of classic methods on ear detection applications and the initial use of CNN on object detection. Chapter 2 introduce all the methods employed in this project, including 3 object proposal methods and the main fast R-CNN algorithm. The database that this project used is described followed as well. Chapter 3 shows the result of this method under different restrictions as it usually happened in the real world. And the analysis of time usage in different part of the algorithms. The final chapter will discuss the utility of results in the previous chapter, and make a conclusion of the achievement and contribution of this project. Appendix will include the platform specification and a brief introduction of the environment set-up.

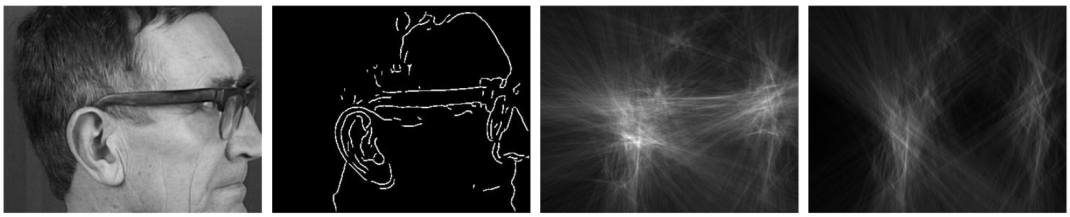
1.3 Literature Reviews

In this section, the previous work in ear detection is presented with descriptive words. It was mainly about the classic methods which focus more on how to design the feature

space then scan the whole image to match such features. However, we are implementing a novel method for ear detection which no one has done before, no related paper can be discussed in this section about it.

1.3.1 Reduced Hough Transform for Ear Detection

Hough Transform (HT) is a very classic algorithm in image processing widely used for feature extraction and patten recognition. It is useful to find the imperfect instances of objects in certain shapes which is very suitable for ear detection as the ear is just like a ellipse and remains that way. Hence, in 2007 Arbab-Zavar et al.[1] used it to design an algorithm for automatic ear detection. Despite the advantages, HT has certain drawbacks such as high computational requirement and memory usages, therefore they used a reduced Hough Transform to overcome those problems as it was specified for only detect ellipse using the known geometrical properties of it to decompose the parameter space from 5D to 2D.



(a) (b) (c) (d)

Figure 1.1: The process of Arbab-Zavar's method [1]. (b)Canny edge (c)accumulator of ellipse (d)reduce the horizontal vote

Firstly, a Canny operator is applied to get a smoothed edge detected image, then the reduced HT transform reconstruct it into an accumulator space as it shown in figure 1.1 (c). The locations of the peaks will provide the coordination of the best matching ellipse. However, there are some mismatches as well, such as the presence of the spectacles. The way to eliminate most of them is to get rid of the horizontal vote by HT, due to the ear shape is mostly a vertical ellipse in the database.

The result shows that this algorithm achieves error-free on the XM2VTS derived, 252-image database. But not that good in UND database, given that it include more backgrounds informations. When applying occlusions, despite the HT was known as tolerate to noise and imperfect, the detection accuracy still drops below 80% after 40% occlusion percentages and reaches 30% at 70% partial occlusion [1]. This algorithm can be seen as very successful under the condition of head profile only database, without backgrounds and large hair covers. It is a very good example of how to maximize the usage of classic methods, however the speed of detection needs to be evaluated if the algorithm needs to be improved.

1.3.2 Haar-like Features for Ear Detection

The Haar-like feature was a very successful algorithm when applied for human face detection by Viola at 2001[10]. Therefore, in 2009 Yuan et al.[11] designed an algorithm which used Haar-like features for ear detection. There are some extended asymmetric Haar-like features shown in figure 1.2 were added, due to the ear structure is very different from face it has more curved outlines.

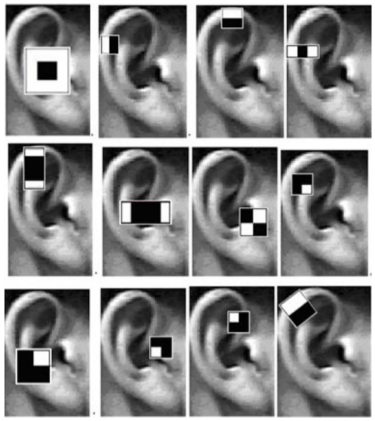


Figure 1.2: Haar-like features reflects the local features

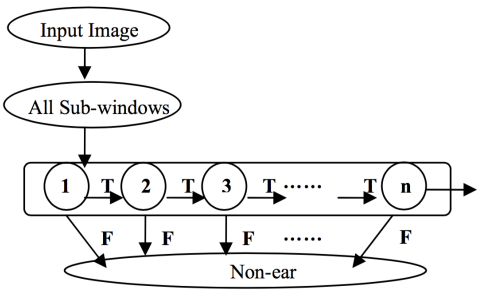


Figure 1.3: The procedure of classify ears

They trained several strong classifiers with AdaBoost algorithm and then cascaded them together into a multi-layer classifier [11]. As all the sub-windows of the image pass through each one of the single classifier, false response will immediately reject the cor-

responding sub-window. Only the one went through all classifiers will be marked as an human ear.

The training sample came from their own database USTB which including 11,000 images with half left ear and half right. There are also 10,000 negative samples from another face database. The result was very impressive with only 0.5% False Reject Rate and 2.3% False Acceptance Rate on USTB 220 testing images. However this method will be highly related to the quality of training dataset and require good control of over-fitting.

1.3.3 SIFT Point Matching for Ear Detection

Scale-invariant feature transform also known as SIFT is a famous computer vision algorithm for detecting and describing local features in images. It extract the interest point of the object in an image to provide "feature description" of the object, then a chosen distance can be used to decide if two description are the same between two images. Advantages of the SIFT for ear detection is that it is scale-invariant and not sensitive with noise and illumination.

Based on the method written by Brown et al.[12], they try to created a homography transform between a probe image and a known gallery object image using SIFT matches. If an homography can be created, means that the probe contains the gallery object. In addition, 4 matched SIFT points were used to align if they lies in one plane , due to the unreliability of more points. Although it can provided an accurate result, a RANSAC algorithm was used to select the best match.

The detection result was under the XM2VTS derived, 252-image database shown that it achieved 96% rank-1 detection accuracy [13]. However it requires the predefined galley of object image which needs a manual mask to locate the object.

1.3.4 Shaped Wavelets for Ear Detection

Due to the ear image mainly contains a lot of curvilinear structures, Ibrahim et al.[2] convolve the image with some curved wavelet filters called "banana wavelet" shown as the figure 1.4 to perform a generalized template matching to detect the location of ear.

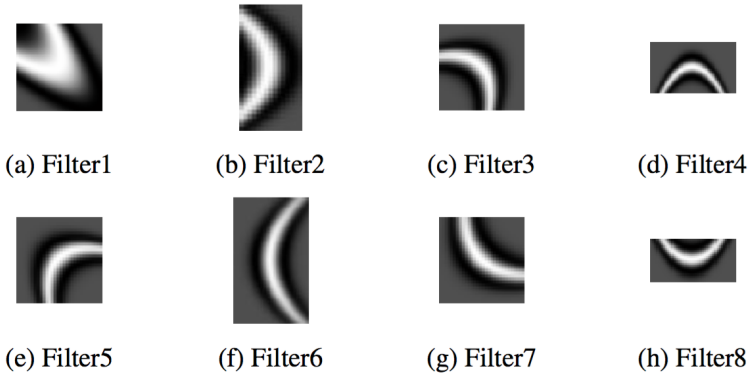


Figure 1.4: Banana wavelets used in this method[2].

The Banana wavelets are a generalization of Gabor wavelets and it can be parameterized by four variables: frequency, curvature, orientation and size. They use 8 filters which prove to be sufficient for ear detection as it shown above in figure 1.4. Initially, they do the convolution between image and the banana wavelet which resulting the magnitude of the filter response. Then the local maxima of the magnitude should be the position where ear has similar curvature, size and orientation to the specific corresponding banana wavelets, as it shown below in figure 1.5. Finally, some threshold and anti-overlapping algorithms can be applied to make the decision where is the ear.

The results demonstrate that this is a very promising methods which achieve 100% detection rate on the XM2VTS database and above 98% when the Gaussian noise ($\sigma = 100$) is presented. However, when testing on the SOTON database with some occlusion, the detection accuracy drop dramatically to 44.7% with partial head and small occlusion[2].

Although the algorithm treat the ear as the combination of some curved lines and focus on finding those lines, it cannot achieve better detection rate when occlusion occurs. The amount of calculation must be very big due to the several fully convolution of whole

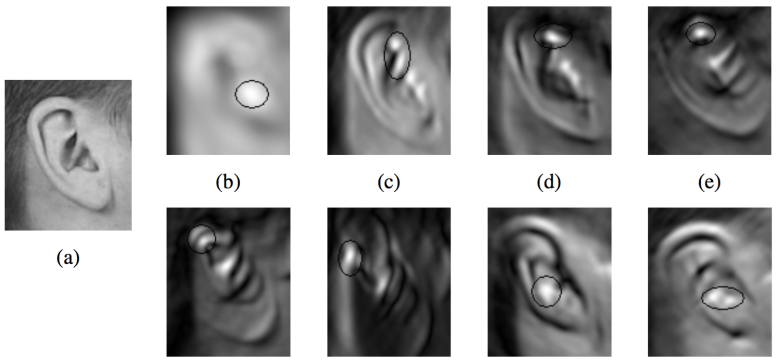


Figure 1.5: (a) Input image, and (b)-(i) after convolution with 8 banana filters[2]

image with 8 wavelet filters, but the author did not mention anything about speed.

1.3.5 Active Contour for Ear Detection

In 2011, Kumar et al.[14] tried to develop an online application which can use ear biometric as an authentication method. They build a wooden stand for acquiring the image as it shown in figure 1.6, so there are no extra illumination management require, the ear was captured inside a isolated box with utilized camera flash light. Which should make the detection more easy, however can not handle the real scenario image.

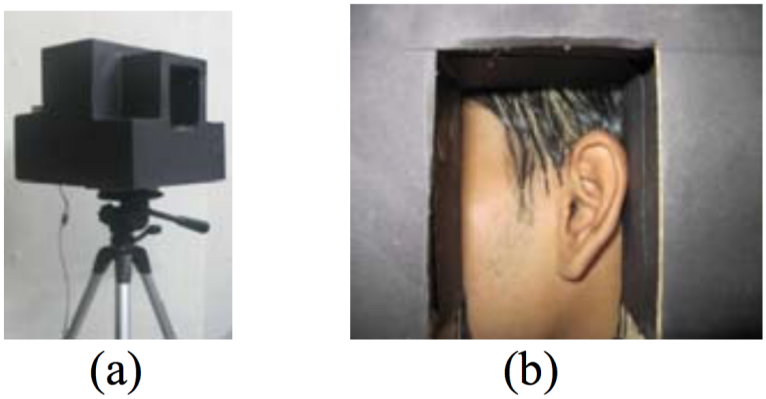


Figure 1.6: (a)Imaging setup (b)Sample captured image

First of all, they use Gaussian classifier to detect the skin region, then apply LoG(Laplacian of Gaussian) for edge detection and remove the false edge until only ear edge left. By using the top and bottom pixel of the ear edge, they can manage to rotate the ear with

reference to the vertical axis. Finally, a localized region based active contour model is applied to extract the ear part only from the ear-ROI(Region of Interest). The whole process is shown in figure 1.7.

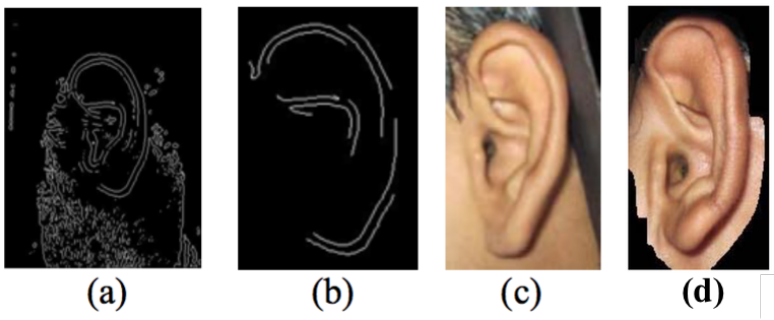


Figure 1.7: (a)Ear Edge by LoG (b)False edge removal (c)ear-ROI (d)Ear-contours

The database they created is 100 users with 7 pictures each. However, the detection rate from the ear-ROI is about 94.2%, which means the correct ear-contours extract is 660 images, the blurry images and the presence of hair are the ones failed [14]. As they only uses the specific controllable database, it is no doubt that this method will failed more when applying in real life images.

1.3.6 Deep Convolutional Neural Network

In 2012, Hinton and his colleges in University of Toronto decided to use deep convolutional neural networks solving the image classification problem [3] They designed a deep network which has 60 million parameters and 650,000 neurons, contains five layers of convolution and some max-pooling layers. Three fully-connected layers forwards data into a 1000-way softmax classification result, due to the ImageNet LSVRC-2010 database has 1000 different classes.

The architecture can be found above in figure 1.8 which shows two parallel processes of the network. It was because that the database has 1.2 million training examples which too big to fit on one GTX 580 GPU with only 3GB memory. Therefore each GPU takes half of the first layer convolutional kernels which are 48 and performing a concurrent

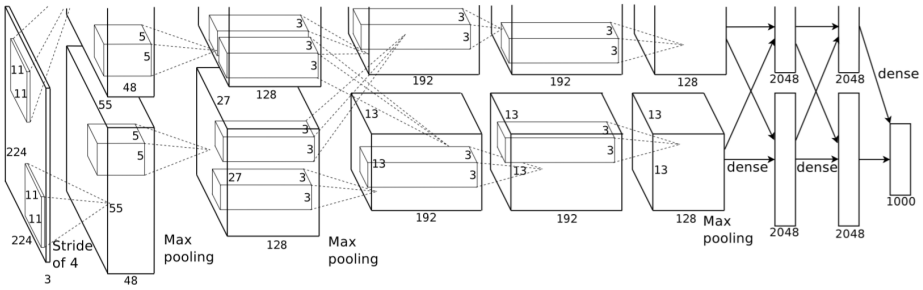


Figure 1.8: Architecture of the Convolutional Neural Network [3]

training with data communicating at certain layers.

In order to reduce the chances of over-fitting which can easily happen in this network due to the large amount of parameters, they augment the data by extracting random 224 x 224 patches (and their horizontal reflections) and uses "dropout" techniques which randomly shut down hidden neurons with 0.5 probabilities. The result they achieved is way better than the previous state-of-art, decreased almost 10% error rates on ILSVRC-2010 test set. They noticed that the architecture is important for network performance, even the remove of a single convolutional layer will degrades result.

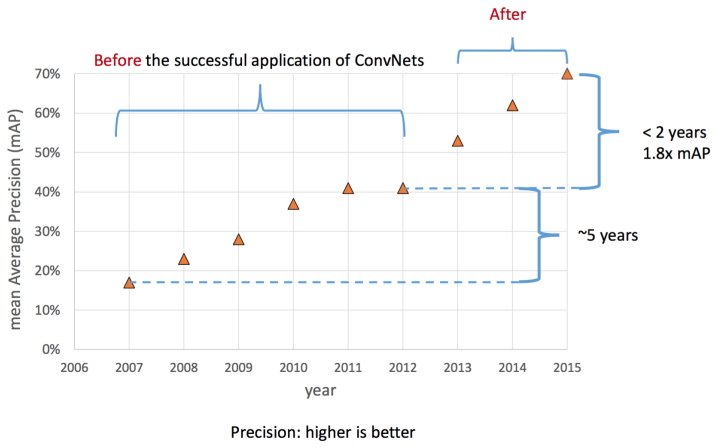


Figure 1.9: Historical PASCAL VOC object detection rate

The contribution of this CNN architecture is enormous, lots of following researches aimed on object detection and image classification were based on the modified version of it. As it shown in figure 1.9, the successful application of CNN quickly pulls up the mean Average Precision (mAP) in PASCAL VOC challenge recently. The method that used

in this project also take advantages on it.

1.3.7 Big Improvement on Object Detection

The most famous and also the first rank of PASCAL VOC 2012 object detection challenge in 2014 is the algorithm called R-CNN designed by Ross Girshick [15]. It amazingly improved the mAP by 30% and firstly indicate that a CNN can lead to dramatically higher object detection performance on PASCAL VOC as compared to systems based on simpler HOG-like hand-craft features.

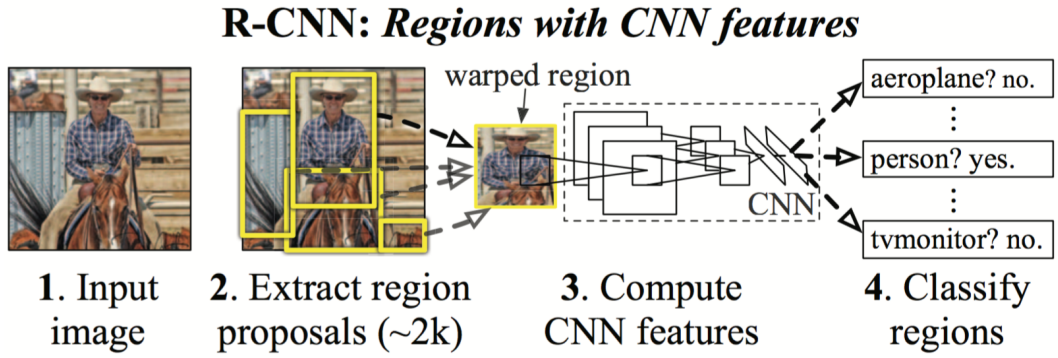


Figure 1.10: Object detection system overview

Object detection require one more step than image classification, localization. It can be solved by a sliding-window detector which CNN has been used for at least two decades. However, Ross solved it by obtains an region generating procedure first and name it as the âĂIJrecognition using regionsâĂİ paradigm [15]. First of all, they generates around 2000 category-independent size-unfixed region proposals of the image. Then use a simple method to wrap those region proposals into fixed size for CNN input. Finally CNN will output fixed-length feature vector for each proposal, then it can be classified by category-specific linear SVMs.

The second principle contribution of this paper is called "fine-tuning" when your database is scarce due to the size of CNN. ROSS uses the very big ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset as an auxiliary supervised pre-training to ini-

tialise weights in CNN. Then fine-tuning it on a more domain specific smaller dataset which one can normally have. Their experiment proves that fine-tuning improves mAP performance by 8 percentage points.

It is a big step towards good performance computer vision object detection which stagnated for years. By the inspire of this method, we hereby use it to solve the ear detection as a specific object detection method in this project.

2 | Methodology

This chapter will be explaining the method used in this project in order to detect the ear under a 2-Dimensional image. Firstly, explain which approach will be used for the preprocessing of all images and why. Secondly, three object(region) proposal methods will be introduced in order to produces many category-independent and size-unfixed regions per image. Then the fast R-CNN algorithm for feature extraction and classification of all the regions to determine which one is the actual ear we want. If there are two many regions correctly classified as ear, it will perform a bounding box regression to precisely localize the box. Finally, the SOTON ear database is introduced.

2.1 Object Proposal Methods

In order to locate the object (in this case the ear), classic approaches over the past two decades have been scanning the whole image by sliding a window which is computational intensive and consequently consuming a lot of times [15]. Therefore, recently this object proposal approach has become the state-of-art method for object detection in computer vision. It dramatically reducing the amount of candidate bounding boxes from tens to hundreds of thousands of locations per image into hundreds of it. Moreover, it is generalized for all object categories, unlike the classic method which is necessarily difficult to design and choose features for every object category.

According to the survey by Jan Hosang et al. [4] in 2014, they evaluated 12 different object proposal methods regarding to four perspectives detection time, repeatability, recall and object detection accuracy. The result of their comparison was shown in figure 2.1. According to the comparison, BING and EdgeBoxes stood out as they are the most fast ones which can finish within 1 second. However, considering of the all other three

features, Selective Search seems the best method.

Method		Time	Repeatability	Recall	Detection
Objectness[1]	O	3	·	★	·
CPMC[4]	C	250	-	★★	★
Endres2010[9]	E	100	-	★★	★★
Sel.Search[30]	SS	10	★★	★★★	★★
Rahtu2011[24]	R1	3	·	·	★
Rand.Prim[22]	RP	1	★	★	★
Bing[6]	B	0.2	★★★	★	·
MCG[3]	M	30	★	★★★	★★
Ranta.2014[25]	R4	10	★★	·	★
EdgeBoxes[33]	EB	0.3	★★	★★★	★★

Figure 2.1: Overview of detection proposal methods. Time is in seconds.[4]

Therefore, we chosen these three methods as our candidates to performing the preprocessing of ear detection in this project. It will be described in detail in the flowing sections.

2.1.1 Selective Search

It is common in the last decades that to use a sliding window for performing exhaustive search to generates region proposals. Although the search space was reduced by using some method such as regular grid, fixed scales, and fixed aspect ratios. In most cases, it still remain huge number of locations to visit. Need not to mention that many of the regions is not supportive at all, because there are no rules for selection of those regions. Therefore, people start to design an algorithm that performs data-driven analysis on an image which can produce generalised region for only object contains in the image. One of the popular approach appeared in public called Selective Search proposed by J.R.R. Uijlings et al. in 2013[16]. It has been widely used for object detection method in 2012, and the detection method based on it produced a very good result in the PASCAL VOC challenge.

Selective search combined the best of intuitions of exhaustive search and segmentation

to produce an image-driven method. They consider the objects in an image as hierarchical, different size, texture and colour objects result in different layers. In order to capture all possible objects in different sizes, they use the "Efficient GraphBased Image Segmentation"[17] method to generate segmentation of an image by different segment size parameters as shown in figure 2.2. It is clearly illustrated that human and sheep can be boxed in different sized image segmentation. Hence, the main steps of this method

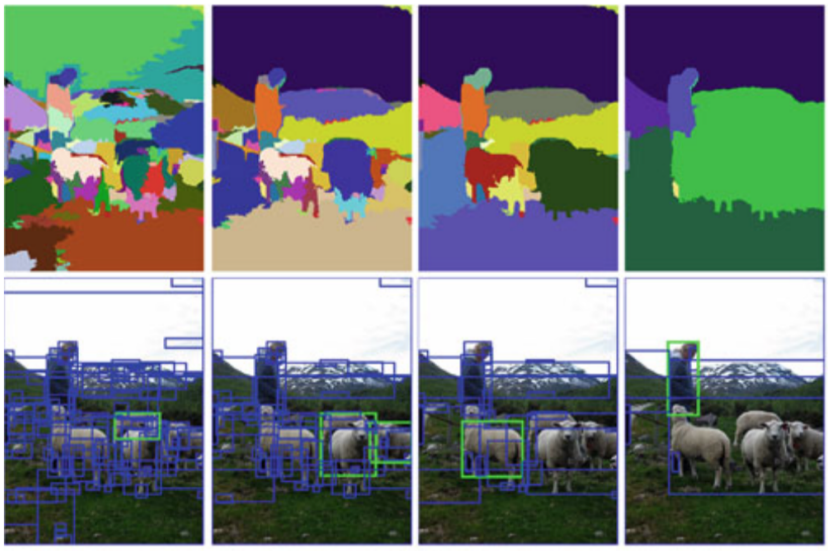


Figure 2.2: Example of Selective Search method on "multiscale"

can be described as below:

1. Produce segmentations of image under different scales for the size of segments,
2. Computing the similarity between each segment and merge the most similar region.
3. Keep record of the box of regions, and keep doing last step until the whole image become a single region.
4. Choose a Stochastic scoring method to ranking those regions and the subset of top k region is the result.

When computing the similarities in step 2, there are two **diversification strategies** used to increase the possibility of capturing all objects. One for colour spaces which including RGB, Lab, HSV and so on, in order to account for different scene and lighting

conditions. The other one strategy for the calculating of region similarity involves colour, texture and region size. In other words, small regions tends to be merged first, same colour or texture tends to be merged first as well.

Therefore in the application of selective search method, the choose of these strategy is an trade-off between calculation complexity and object proposal completeness. We performed an analysis upon this in the following section 3.1.

2.1.2 Edge Boxes

One year later, a different idea of generating object proposal stands out as it not focus on generating all the potential object boxes, but hope to reduce the number of proposals left only high-quality proposals. According to C. Lawrence Zitnick and Pitor Dollár, since object proposal method aims primarily reduce the computational cost of the detector, it should be significantly faster than the detector itself. Then, in 2014 an improved approach which based on the edge information of an image has been developed by them [18]. It can produce the amount of 10^3 proposals in about 0.25 seconds. Edge, as a simple representation of an image, it can provide more informations than the segmentation. And it also holds computational advantages by using another edge-map producing method from Pitor Dollár "Structured Forests for Fast Edge Detection" [19], it can obtains sparse edge maps.

The main idea of this method is that **the amount of contours wholly enclosed in a bounding box indicate the possibility of box containing objects.** Therefore, the final result of object proposals are ranked based on a score computed from the wholly enclosed contour in a candidate proposal. They designed an approach to calculate the magnitude and orientation of each pixel in an image, then by combining these information the affinity can be calculated for each pixel. As it shown in figure 2.3, row 3, before obtaining contours the edges are formed into groups using a simple greedy method which combined 8 connected edges until their orientation differences went beyond a certain

threshold. The small groups are automatically merged into neighbouring groups.

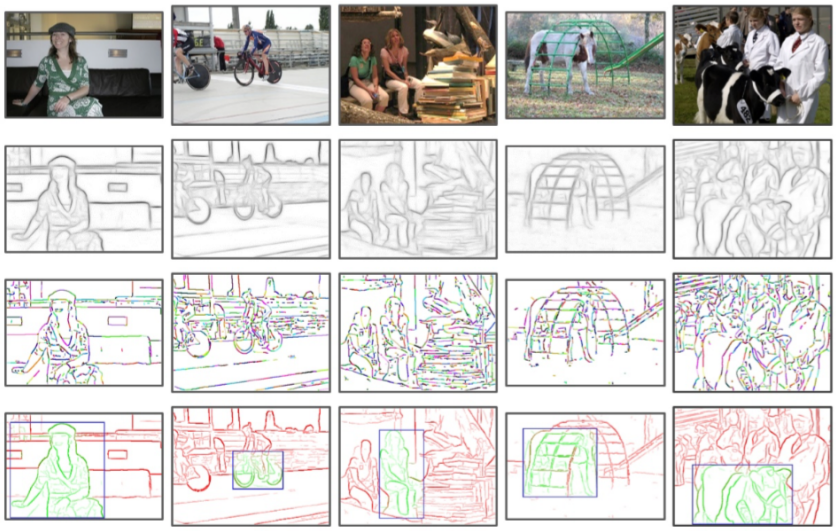


Figure 2.3: Illustration examples for the process of Edge Boxes method

The main procedures of the Edge Box method can concluded as described below:

1. Structured edge detector [19] was used to efficiently obtain edge informations of an image.
2. A Non-Maximal Suppression (NMS) orthogonal was used to obtain a more sparse edge image.
3. Grouping all the edge point into different edge groups as shown in different colour in figure 2.3, row 3.
4. Calculate the affinity between edge groups using author's formula.
5. A smart sliding window method search over position, scale and aspect ratio to obtains candidate boxes.
6. Computing object proposal score for each candidate box, higher score will be potential objects.

In step 5, a naive sliding window method would be prohibitively expensive. They proposed an very efficient method for finding intersecting edge groups based on two additional data structure. Due to the purpose of this project, details of it are not presented here, it can be found in [18]. The model we used in this method for ear detection is trained under PASCAL VOC 2007 dataset.

2.1.3 Binarized Normed Gradients (BING)

In order to achieve a real-time object detection, the object proposal approach must be very fast. Therefore a very efficient objectness estimation which can achieve 300fps with a single laptop CPU on PASCAL VOC 2007 dataset designed by Ming-Ming Cheng shown up in 2014[20]. This method is the fastest one among all the object proposal method recently according to Hosang's survey[4]. They also used edge information to find the object location, but in a very special way which only require a few atomic operation such as ADD, BITWISE SHIFT, etc. That is why this method is very fast.

Very mush like Edge Boxes method, BING was motivated by the fact that if stand-alone things have well-defined closed boundaries and centres then it is highly possible to be an object. In order to find the generic object inside an image, they firstly took the gradient of the image and scan it over predefined scales and aspect ratio like figure 2.4 (b).

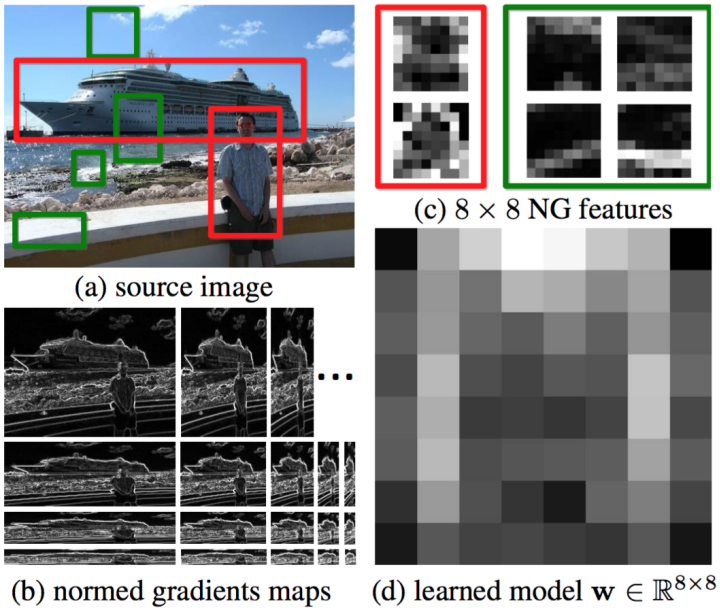


Figure 2.4: Object(red) and non-object(green) (a) corresponds to normed gradients(NG) features (c) in proper scales and aspect ratios (b), a single 64D linear model (d) can be learned

Then by using Non-Maximal Suppression (NMS) and some linear filter model [20], they obtained the normed 8*8 gradient (NG) features from each scaled images. By doing that, they declared that the NG features are insensitive to variation of scale, aspect ratio

and translation. As it shown in figure 2.4 (c), red NG features corresponding to object ship and person while the green represent non-object. It shows clear correlation of the object NG features although the cruise ship and the person have huge different in terms of illumination, texture, colour, etc. Hence, the object NG features can be gathered to train a generalised 8*8 NG model which shown in figure 2.4 (d).

The reason why the shape of 8*8 is their secret to significantly increase the speed. Naively, it would require a for loop to accessing 64 ($8 * 8 = 64$) positions and calculating which needs time, however they treat every feature as a int64 type number so that no need for iteration. The Binarization of NG feature was a work from Sam Hare in 2012 [21], which transform the NG feature into Binarized NG (BING) feature as shown in figure 2.5.

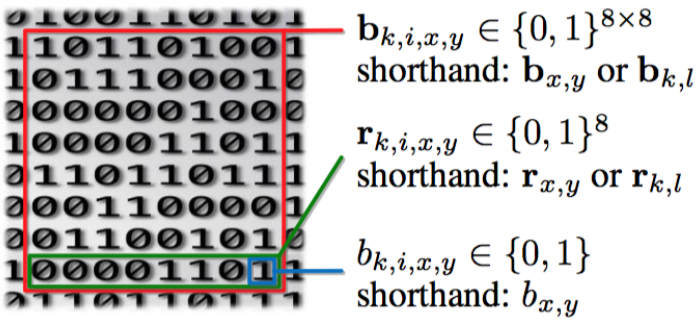


Figure 2.5: Illustration of variables: a BING feature

Hence the 64 digit of BING feature can be saved as a single int64 and its last row can be saved as a byte variable. All the calculation can be done by BITWISE SHIFT and BITWISE OR which is very natural computer memory operation and can be super fast. Finally those int64 BING features can be compared with the learned model to determine whether it is an object or not. The whole process can be concluded as steps below:

1. Produce a series of normed gradient from the image in different scales and aspect ratios.
2. Use Non Maximum Suppression(NMS) to obtain 8*8 gradient feature.
3. Binarize those features in order to obtain Binarized Normed Gradient(BING) features.

4. Compare BING with learned model, calculate the value of objectness, use it to rank the object proposals.

The implementation of this method is written on c++ which makes it even more fast. The model that we used for ear detection in this method is trained on PASCAL VOC 2007 dataset. There are not too much parameters we can tune, only the number of proposals to be generated, so we will find the most suitable one in section 3.1.

2.2 Deep Learning Method

In recent years, the using of deep convolutional neural networks (also known as deep learning method) has significantly improve the accuracy in image classification and object detection. The biggest advantage of deep learning is that no need to spend time design and decide for the specific hand-crafted features to use, which also means it can become a generalized solution for different tasks. Although the disadvantages are quite clear, it requires large amount of memories, calculation and consequently time-consuming. One of the solution now is to use GPU to calculate which accelerate the computation quite a lot.

The Fast R-CNN method has been chosen for this project as a solution towards the ear detection problem, because the source code in open in public, require less time for training than the previous version (RCNN) and can modify different object proposal methods compare to the latest version (Faster R-CNN).

2.2.1 Fast Region-based Convolutional Network (Fast R-CNN)

Because the ear detection can be seen as a specific object detection task, therefore, the state-of-art method on PASCAL VOC challenge will be very useful. In 2015, Ross Girshick designed this Fast-RCNN algorithm [5] based on the improvement of his previous

RCNN [15] method, and quickly become one of the popular method in the challenge. It reaches 70 mean Average Precision(mAP) under the dataset of PASCAL VOC 2007, and runs 9x faster than R-CNN at training-time and 200x faster at testing-time.

Very much like the R-CNN approach which we talked about in section 1.3.7, this fast R-CNN method reduced the multiple calculation in previous approach and combine the final box location regressor together with the CNN module. So that it can directly produce two result: object probability and corresponding box location. In R-CNN the convolution will be performed on each object proposal boxes from input which wasted lot of time due to the boxes are often overlapping. Then this time Ross only perform the convolution once in the beginning and then use the proposal boxes as a projecting map to obtain corresponding convolutional result. As it shown in figure 2.6, this algorithm can be divided into 4 main steps:

1. Object proposal boxes and ground truth object box for each image goes into the network.
2. Use convolutional network to extract features from each proposal boxes, involving Region of Interest (ROI) projection method to prevent multiple calculation.
3. Training classifier to decide whether a proposal box belong to one of the classes.
4. For each set of overlapping proposal boxes, use a regressor to obtain the final precise location.

Therefore, two outputs were presented as in figure 2.6, one for the probability of which class the corresponding box is, the other for the bounding box location. In this case, *ear* and *background* are the only two classes which actually simplified the training process.

It is also very important to choose which pre-trained Convolutional Neural Network (CNN) shown as the middle part in figure 2.6 to use, different network contains different structures such as how many levels of convolution layers, pooling layers and full connecting layers. The author of fast-RCNN, Ross Girshick recommended to use his pre-trained (trained under PASCAL VOC dataset) models:

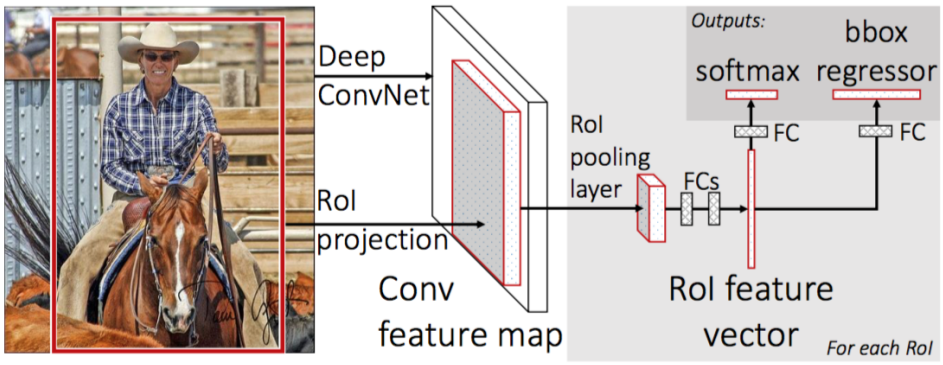


Figure 2.6: Fast R-CNN architecture[5]

- **CaffeNet** refer to author as essentially "AlexNet" which is the first CNN invented by Hinton et al. [3]. This is a small one compare to others but still require 1GB GPU memory.
- **VGG_CNN_M_1024** which has the same depth as CaffeNet, but the architecture is wider [22], referred as medium size.
- **VGG16** is a very deep model from Karen Simonyan [23] in 2015. This is the largest model.

Due to the limitation of the current GPU hardware we have as described in appendix A and the amount of training time we are willing to tolerate, the small one "CaffeNet" has been chosen as our network. However, the full training time of 40,000 iteration (as recommended) on the database of our choice in section 2.2.2 is still up to 6 hours.

2.3 Database

This project uses the ear image database obtained by University of Southampton gait lab [24]. It covers more than 400 peopleâĂŹs profile of gait, face and ear while they walking pass a tunnel. Therefore the ear images was collected in an unconstrained way, as subjects walking pasted a camera. The illumination while capturing pictures is also not very good as shown in figure 2.7, moreover, the tunnel has some painted background used for camera calibration which also make it more challenge for ear detection. But,

these are good for training a robust ear detection algorithm. Another advantage of this database is it also contains the ear location of each image which is convenient for directly use as the ground truth in deep learning network.

Figure 2.7: Sample images from SOTON ear database

The database contains totally 600*800 sized 548 colour images with 2-4 images per subject, we randomly divided it into 80% training (437 images) and 20% testing (111 images). The only augment for images while training the network is a horizontally mirror image due to the ear is pair-wised for human beings.

3 | Results

This chapter will include all the results produced during this project experiment. It included the plot of time usage under each method from section 2. As for the simulate of real world scenario under database image, we obtained two ways which are Gaussian Noises and Partial Occlusions. Different inspection methods were applied for the accuracy, robust and reliability comparison between each methods.

3.1 Object Proposal Method Tuning

For each object proposal method, there are some parameters needed to be chosen in order to perform better detection accuracy. In order to evaluate the performance of object proposal method, we use a method called Intersection of Unit (IOU) as shown in figure 3.4 below:

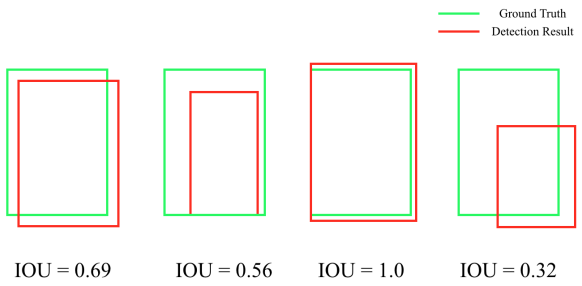


Figure 3.1: Illustration of the measurement of Intersection of Unit (IOU)

Because the performance of object proposal method in highly related to the number of proposal boxes which contains ear (high IOU with ground truth box). We calculate the IOU for every proposals and count only the one with $IOU >= 0.1$ then divided by total number of proposals to get the Potential Proposals Ratio (PPR).

3.2 Time Usages

As for the three Object Proposal methods we introduced, the former two algorithms Selective Search (SS) and Edge Boxes Detector (ED) was released only in MATLAB version, but the Binarized Normed Gradients (BING) was in C++. Therefore, BING will be faster with respect to the data exchange between MATLAB and the neural network will be reasonably slower.

By the record of a survey produced by Jan Hosang et al. [4], the approximate time of these algorithms are 10, 0.3 and 0.2 seconds for SS, ED and BING. However it is slightly different under our hardware specification as measured and plotted below in figure 3.2. We randomly chosen 200 images for the time measurement.

- For **SS**: it takes about 1.3 seconds for average 400 object proposals to be calculated.
- For **ED**: 0.5 seconds in average when producing the least proposals at 200.
- For **BING**: the number of proposals is fixed to 800 per image and the time is almost the same as ED at 0.5 seconds.

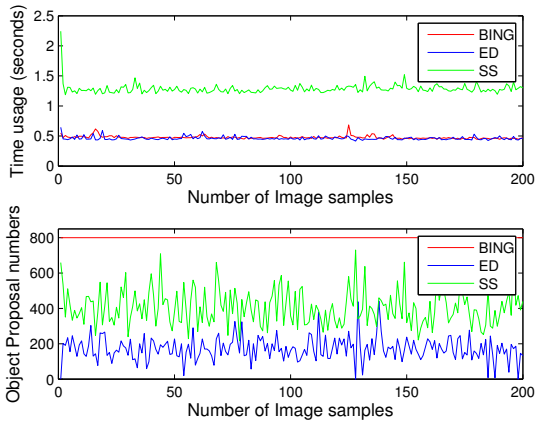


Figure 3.2: The time usage and corresponding object proposals

As for the time usage in the CNN part is highly related to the amount of object proposals. Hence after several attempts and measurements, the average times for the 600*800 colour

image in the SOTON database for those 3 methods SS, ED and BING are respectively **1.75, 0.73 and 1.05 seconds**.

3.3 Detection Rate Measurement

This section shows the results of the accuracy when performing this ear detection algorithm. Firstly, it described the conditions for true positive detection and present the result when performing it on testing dataset. Then the noise and occlusion were applied to test the robustness and reliability of this algorithm.

3.3.1 Conditions for Positive Detection

As it shown in figure 3.3 (a), the fast-RCNN will produces many boxes along with its probability to the ear. Therefore, we apply a simple threshold to remove the boxes with $P(ear|box) < 0.8$. Then the remaining boxes was filtered by a Non Maximum Suppression method for only one bounding box left. However, sometimes it may left more than one boxes due to false positive detection.

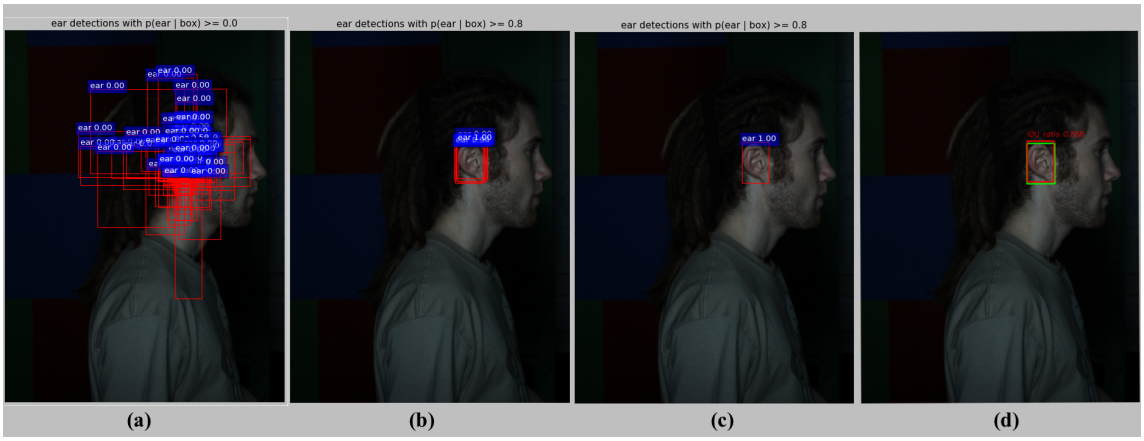


Figure 3.3: (a) All output boxes and probability from fast-RCNN (b) threshold by $P(ear|box) > 0.8$ (c) Non Maximum Suppression of left boxes (d) final result with IOU

We use another method to measure the false positive detection, which is the Intersection of unit rate. It indicate how many area this predicting box overlapped with the ground

truth box as illustrated in figure 3.4. If the IOU rate was less than 0.5, then we determine it as a false positive detection.

- **Positive Detection**: has more than one box with $P(ear|box) > 0.8$.
- **False Positive Detection**: fulfill requirement for positive detection but the biggest IOU is less than 0.5
- **Negative Detection**: no boxes with $P(ear|box) > 0.8$.

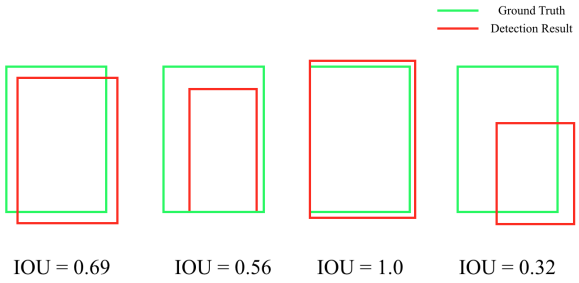


Figure 3.4: Illustration of the measurement of Intersection of Unit (IOU)

3.3.2 Gaussian Noise Distraction

In this section, we add random Gaussian noises into the raw images with 0 mean and variational σ from 5 to 30 to stimulate the noises in real world scenario such as CCTV surveillance images. The sample images can be seen from figure 3.5 below. It illustrates how severe the noises are with different scale of σ.

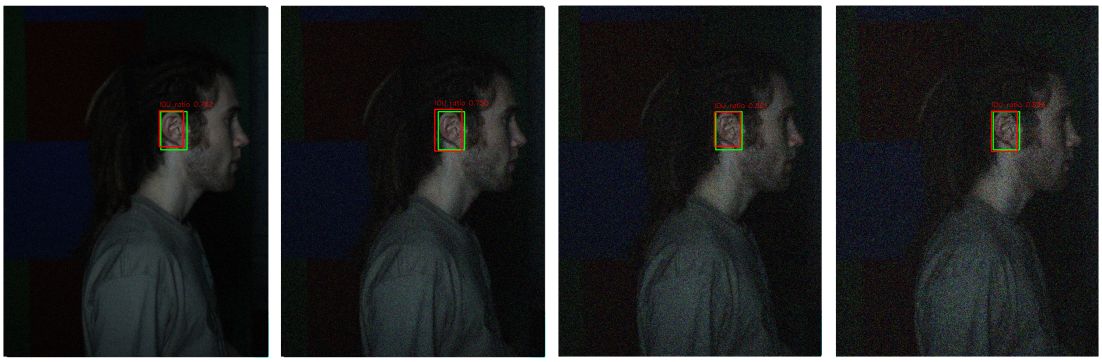


Figure 3.5: Gaussian noise distraction samples by ED method

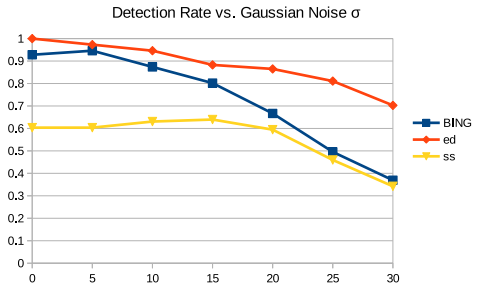


Figure 3.6: True positive detection rate under different Gaussian σ

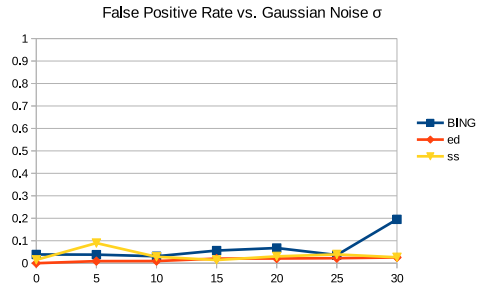


Figure 3.7: False positive detection rate under different Gaussian σ

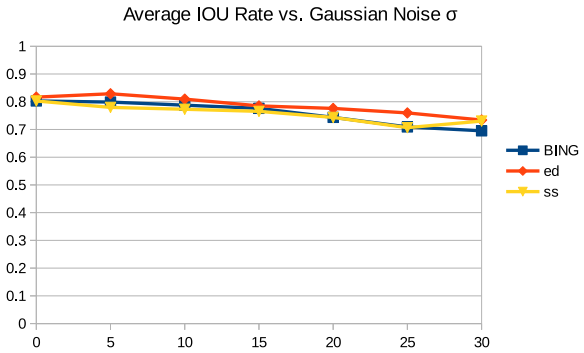


Figure 3.8: Average of IOU rate under different Gaussian σ

Three result parameters were chosen as the performance indicators to help compare these 3 object proposal methods. They are well explained in section 3.2.1.

For the detection rate, it is plotted in figure 3.6 and 3.7 showing good accuracy even for $\sigma = 30$ with more than 70% positive detection rate via ED object proposal method. The BING method, even it provided the most object proposal boxes up to 800 per images, the detection accuracy decreases almost linearly with the increase of noise ratio. It only reached beyond 90% in the raw images and with $\sigma = 5$ Gaussian noise.

Surprisingly, both BING and SS methods have an increase of detection accuracy along the increase of noise in the beginning. It might be because of the Gaussian noises change the number and locations of object proposal boxes, and accidentally triggered some positive detections.

On the other hand, all 3 methods produced good reliability along the increase of Gaussian noises. All of the False Positive Detection rate remains below 10% except for BING at

$\sigma = 30$. Another result in figure 3.8 also proved good reliability, which states that the average IOU rate of all the positive ear detection by these methods are above 70%.

3.3.3 The Use of Partial Occlusion

Usually ear will be covered by either hair or some kind of earrings, therefore a robust ear detection algorithm must be able to detect ear under that circumstances. Therefore we simulate the occlusion of ear by put a black mask onto the ground truth ear box. The mask size was determined both by the ear box size and the parameter called "occlude percentages". As it shown below in figure 3.9, we test from 10% to 50% occlusion, found that 50% occlusion was very difficult for our algorithm to perform detection.

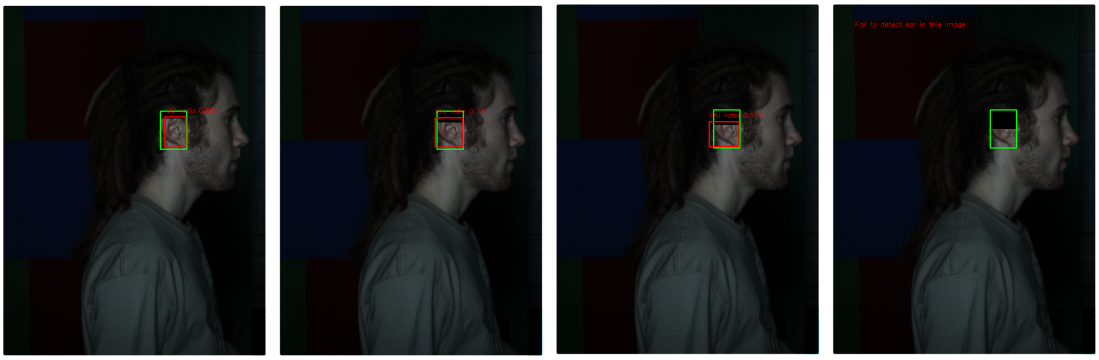


Figure 3.9: Partial occlusion samples by ED method

The detection rate drops dramatically when the occlusion percentages hit 50%, however, the ED method can have more than 90% accuracy when dealing less than 50% occlusion with less than 20% false positive detection. The BING method only performs well when occlusion percentages less than 30%, reaches more than 90% detection accuracy. However, it decreases exponentially after 30% with the rise of false positive detection rate. The SS method performs normally from 60% in raw images to 10% in 50% occlusion images.

Despite of the low detection rate with SS method, the average IOU of it reaches the same

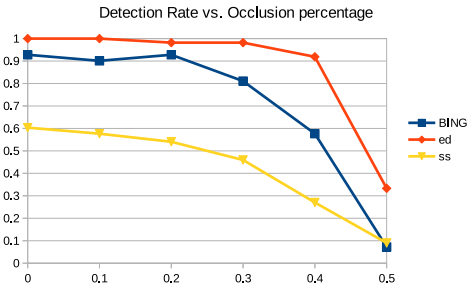


Figure 3.10: True positive detection rate under different occlusion percentage

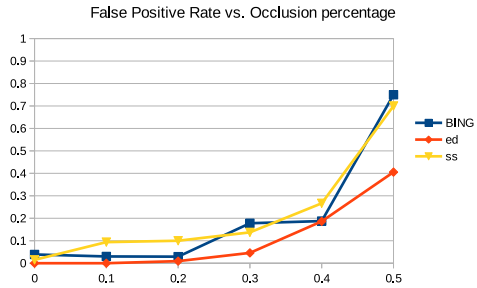


Figure 3.11: False positive detection rate under different occlusion percentage

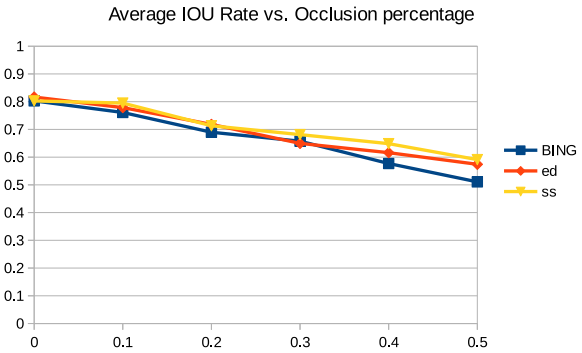


Figure 3.12: Average of IOU rate under different occlusion percentage

level as the other two methods. It only drops from 80% to 50% alongside the increase of occlusion percentages.

4 | Discussion & Conclusion

The objective for this project is to apply deep learning method into ear detection and achieve a good detection rate under different circumstances. In the section before, result proves the feasibility of such an algorithm by combining object proposal method with fast Region Convolutional Neural Network. Although the time for our algorithm is still more than 500ms per image which unable to perform a live detection, the old laptop we used is responsible for that. It can be easily solved by using a higher performance computer.

4.1 Summary of Project Achievements

In the past decades, the classic hand-craft feature matching method was dominating the ear detection, or even object detection fields. It is a competition of finding the most useful features to representing ear and highlight it in an image. Many researchers have found their way to achieve rank-1 detection rate over 90% or even 100% on their chosen database [25]. However, it is the generalization and robustness problem that prevent this technique from widely used in real world applications.

Along with the explosion of computer performance and the successful application of Convolutional Neural Network (CNN) into object detection and classification recently. Although CNN was invented in 2012 by Hinton et al. [3], it boosted the development in many fields not only for computer vision. Figure 1.9 illustrates that it only took 2 years for CNN to almost doubled the mean Average Precision in PASCAL VOC object detection challenge. Theoretically, ear detection is an object detection method specifically for ear. Therefore it should be easy to combined with the state-of-art CNN method in order to perform such a complexity-reduced problem, yet no record can be found on

the Internet that anyone has attempt this before.

This project successfully bring the state-of-art method fast-RCNN into the first step of ear biometrics, ear detection, and achieved a rather good rank-1 100% detection accuracy. We attempt to inspire the following researchers by

4.2 Applications

Applications.

4.3 Future Work

Future Work.

A | Hardware & Software Specification

The Training and testing of this project was undertaking by a LENOVO Y470 laptop which has specification below. The reason for not using high performance computer in the Computer Lab is the graphic driver of the lab computer is incompatible with the installation of fast-RCNN network.

- **Purchase Year**: September 2011
- **Operation System (OS)**: Ubuntu 15.04
- **CPU**: Intel Core i5-2540M @2.60GHz x 4
- **GPU**: Nvidia GeForce GT550M
- **GPU Memory**: 1024MB
- **RAM**: 6GB

The main part of the whole project is written in Python and can be easily downloaded from Internet at website `https://github.com/harrysocool/ear_recognition.git` (exclude the ear database). The version of python is 2.7.11, and the three object proposal methods are also modified and kept in the Github repository. The training and testing script of fast-RCNN are highly modified for compatible with our own database.

B

Bibliography

[1] B. Arbab-Zavar and M. S. Nixon, "On shape-mediated enrolment in ear biometrics," *Advances in visual computing*, 2007.

[2] M. I. S. Ibrahim, M. S. Nixon, and S. Mahmoodi, "Shaped Wavelets for Curvilinear Structures for Ear Biometrics." *ISVC*, pp. 499–508, 2010.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks." *NIPS*, 2012.

[4] J. H. Hosang, R. Benenson, and B. Schiele, "How good are detection proposals, really?" *CoRR*, 2014.

[5] R. B. Girshick, "Fast R-CNN." *International Conference on Computer Vision (ICCV)*, 2015.

[6] M. I. S. Ibrahim, M. S. Nixon, and S. Mahmoodi, "The effect of time on ear biometrics." *IJCB*, pp. 1–6, 2011.

[7] M. Burge and W. Burger, "Ear biometrics in computer vision," vol. 2, pp. 822–826 vol.2, 2000.

[8] L. Yuan and Z.-C. Mu, "Ear recognition based on local information fusion," *Pattern Recognition Letters*, vol. 33, no. 2, pp. 182–190, Jan. 2012.

[9] B. Arbab-Zavar and M. S. Nixon, "On guided model-based analysis for ear biometrics," *Computer Vision and Image Understanding*, vol. 115, no. 4, pp. 487–502, Apr. 2011.

[10] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, 2001.

[11] L. Yuan and F. Zhang, "Ear detection based on improved AdaBoost algorithm," in *2009 International Conference on Machine Learning and Cybernetics (ICMLC)*. IEEE, 2009, pp. 2414–2417.

[12] M. Brown and D. Lowe, "Invariant Features from Interest Point Groups," in *British Machine Vision Conference 2002*. British Machine Vision Association, 2002, pp. 23.1–23.10.

[13] J. D. Bustard and M. S. Nixon, "Robust 2D Ear Registration and Recognition Based on SIFT Point Matching," in *2008 IEEE Second International Conference on Biometrics: Theory, Applications and Systems*. IEEE, 2008, pp. 1–6.

[14] A. Kumar, M. Hanmandlu, M. Kuldeep, and H. M. Gupta, *Automatic Ear Detection for Online Biometric Applications*. IEEE, 2011.

[15] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." *CVPR*, 2014.

[16] 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*, vol. 104, no. 2, pp. 154–171, Sep. 2013.

[17] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[18] C. L. Zitnick and P. Dollár, "Edge Boxes: Locating Object Proposals from Edges," in *Computer Vision – ECCV 2014*. Springer International Publishing, Sep. 2014, pp. 391–405.

[19] P. Dollár and C. L. Zitnick, "Structured Forests for Fast Edge Detection," *2013 IEEE International Conference on Computer Vision*, pp. 1841–1848, 2013.

[20] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr, "BING - Binarized Normed Gradients for Objectness Estimation at 300fps." *CVPR*, 2014.

[21] S. Hare, A. Saffari, and P. H. S. Torr, "Efficient online structured output learning for keypoint-based object tracking," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 1894–1901.

[22] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the Devil in the Details: Delving Deep into Convolutional Nets," in *British Machine Vision Conference 2014*. British Machine Vision Association, 2014, pp. 6.1–6.12.

[23] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition." *CoRR*, vol. cs.CV, 2014.

[24] M. S. Nixon, "Soton multimodal database," Unversity of Southampton. [Online]. Available: http://www.cspc.ecs.soton.ac.uk/ear

[25] A. Pflug and C. Busch, "Ear biometrics: a survey of detection, feature extraction and recognition methods," *IET Biometrics*, vol. 1, no. 2, pp. 114–129, 2012.