UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Pattern Matching of Footwear Impressions

A Degree Thesis

Submitted to the Faculty of the

Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona

Universitat Politècnica de Catalunya

by

Christian Segovia Barrera

In partial fulfilment

of the requirements for the degree in

***AUDIOVISUAL SYSTEMS* ENGINEERING**

Advisor: Manuel Keglevic & Xavier Giró

Barcelona, July 2018

Abstract

In this thesis different techniques are evaluated to recognize and match footwear impressions, using reference and real crime scene shoeprint images. Due to the conditions in which the shoeprints are found (partial occlusions, variation in shape) a translation, rotation and scale invariant system is needed. A VLAD (Vector of Locally Aggregated Descriptors) encoder is used to clustering descriptors obtained using different approaches, such as SIFT (Scale-Invariant Feature Transform), Dense SIFT or Triplet CNN (Convolutional Neural Network). These last two approaches provide the best performance results when the parameters are correctly adjusted, using the Cumulative Matching Characteristic curve to evaluate it.

Resum

En aquesta tesi s'avaluen diferents tècniques per reconèixer i aparellar impressions de calçat, utilitzant imatges de referència i d'escenes reals de crim. Degut a les condicions en què es troben les impressions (oclusions parcials, variació de forma) es necessita un sistema invariant davant translació, rotació i escalat. Per això s'utilitza un codificador VLAD (Vector of Locally Aggregated Descriptors) per agrupar descriptors obtinguts en diferents enfocaments, com SIFT (Scale-Invariant Feature Transform), Dense SIFT o Triplet CNN (Convolutional Neural Network). Aquests dos últims enfocaments proporcionen els millors resultats un cop els paràmetres s'han ajustat correctament, utilitzant la corba CMC (Characteristic Matching Curve) per realitzar l'avaluació.

Resumen

En esta tesis se evalúan diferentes técnicas para reconocer y emparejar impresiones de calzado, utilizando imágenes de referencia y de escenas reales de crimen. Debido a las condiciones en que se encuentran las impresiones (oclusiones parciales, variaciones de forma) se necesita un sistema invariante ante translación, rotación y escalado. Para ello se utiliza un codificador VLAD (Vector of Locally Aggregated Descriptors) para agrupar descriptores obtenidos en diferentes enfoques, como SIFT (Scale-Invariant Feature Transform), Dense SIFT o Triplet CNN (Convolutional Neural Network). Estos dos últimos enfoques proporcionan los mejores resultados una vez los parámetros se han ajustado correctamente, utilizando la curva CMC (Characteristic Matching Curve) para realizar la evaluación.

Dedication: *Especially grateful to Ilaria, Daniele, Daniel, Fabio, Francesco, Matteo, Despina, Katerina, Marw, Nefeli, George and Romà for making these months the best of my life.*

Acknowledgements

The author would like to thank Manuel Keglevic for the supervision and dedicated time to help in this project. The support of the Computer Vision Lab from TU Wien is gratefully acknowledged.
The author would also like to thank Adam Kortylewski for providing the database.

Revision history and approval record

Revision	Date	Purpose
0	17/05/2018	Document creation
1	25/06/2018	Document revision
2	30/06/2018	Document revision

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Christian Segovia	christiansegoviabarrera@gmail.com
Manuel Keglevic	keglevic@cvl.tuwien.ac.at
Xavier Giró	xavier.giro@upc.edu

Written by:		Reviewed and approved by:	
Date	17/05/2018	Date	30/06/2018
Name	Christian Segovia	Name	Manuel Keglevic
Position	Project Author	Position	Project Supervisor

Table of contents

List of Figures

List of Tables:

1. Introduction

Shoeprints are a valuable forensic evidence often found at crime scenes. It has been estimated that more than 30% of all burglaries provide usable shoeprints that can be recovered from the crime scene [25]. Because of the pattern of repeated offences, rapid classification of such shoeprints would enable investigating officers not only to link different crimes, but to identify potential suspect. Unfortunately, this process for thousands of images is highly time-consuming to be done manually. For that reason, in order to support the forensic experts, an automatic retrieval of the most likely matches is desired.
Different techniques in computer vision have been demonstrated useful in the object recognition field, especially since the increasing power of the processors in the last years [26]. In this project, we will focus on some of the most popular techniques nowadays, such as SIFT descriptors and VLAD encoder. Also, a convolutional neural network approach is evaluated.

The project main goals are do a literature research in this field, learning and familiarizing with the Python environment and the different frameworks, to design and implement a machine learning system by looking at the state of the art and adapting it to the specific problem of matching footwear impressions. A depth evaluation using available datasets will be done, comparing the different raised approaches and fitting the parameters to evaluate the best working.

1.1. Forensic Shoeprints

A shoeprint is a mark made when the tread of a shoe comes into contact with a surface. In forensic investigation, the shoe print is typically digitized either by photography or by lifting it from the ground with a sticky gel foil which is subsequently scanned. Real shoeprints are very often partial or incomplete prints resulting from incomplete contact between the shoe sole and the surface.

The main challenges are deal with the variation in shape and appearance, the partial occlusion and the unconstrained noise conditions of the crime scene shoeprints. Therefore, due to the conditions in which the real crime shoeprints are found, it is necessary to find translation, rotation and scale invariant features that guarantee us an efficient descriptor of these images.

Furthermore, training and testing data are scarce, because usually no or few crime scene impressions are available per reference impression. For this project the public FID-300 [1] database is used, which contains 1175 different reference images and 300 real crime scene labelled images.

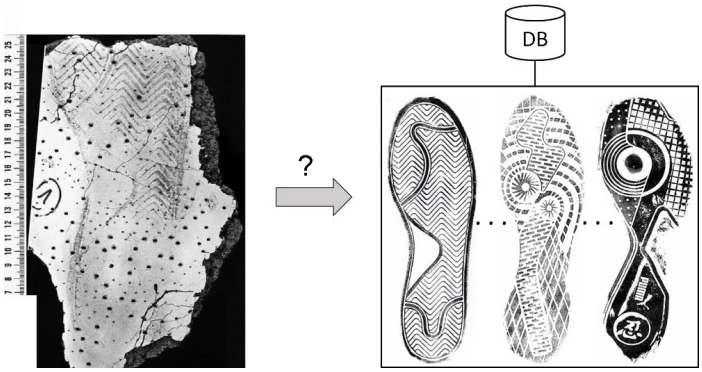

Fig 1. Problem statement scheme. [20]

These shoeprints tend to follow some patterns like lines, circles or squares (figure 2).

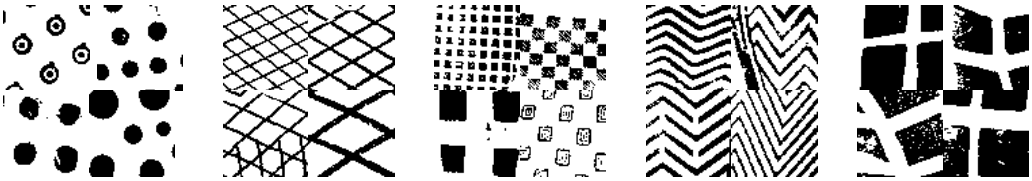

Fig 2. Some typical patterns.

1.2. Project Background

Ideas are in cooperation with the supervisor, Manuel Keglevic from the Computer Vision Lab of the Faculty of Informatics, at the TU Wien, Austria. The project is independent but frameworks from other projects can be used.

Computer vision techniques are the basis of the project, using image processing for pattern matching of the shoeprints. Also, a deep learning approach is studied and evaluated.

A Python 3.5 script is created using the PyCharm Integrated Development Environment (IDE) in order to implement the knowledge previously studied and evaluate the results obtained according to the chosen parameters.

2. State of the art of the technology used or applied in this thesis:

An intensive search of the related work on footwear impressions matching is crucial to understand the state of the art in shoeprint pattern matching. Several investigations in this field have been done, showing good results using only the reference images or synthetically computer-generated images with noise, rotations and partial occlusions [2,3, 4,5,7,8]. However, one key challenge of real data is that the noise is unconstrained and therefore cannot be simulated by such simple noise distributions. Our purpose is to use real crime scene images, in order to obtain results applicable to real situations.

The firstly works were based on using the Fourier transform [2], being a translational and rotational invariant system, but sensitive to noise distortion and incomplete data. They report first rank classification results of 65% and 87% for rank 5 on full-prints using a database containing 1,276 reference images. For partial prints, a best performance of 55% and 78% is achieved, respectively, using computer generated images. These results were improved using the Fourier-Mellin transform [3] to produce translation, rotation and scale invariant features. Topological and pattern spectra [4] approach was based on repeated open operations with increasing size of structuring element, giving a distribution of Euler numbers. Describing the image with respect to its axes through Hu-Moment invariants [5] achieved good results where optimal performance is attained for images rotated by any angle. Also, an approach using fractals [23] to the detection and classification of shoeprints was carried out, it is however only working with small variations in the orientation and translation.

The last researches indicate that the use of local image features as a combination of local interest point detectors and SIFT (Scale-Invariant Feature Transform) [21] feature descriptors presents good performance. However, the better classification results are often obtained by computing the SIFT descriptor over dense grids in the image domain as opposed to at sparse interest points as obtained by an interest operator, receiving the name of dense SIFT [16]. Using MSER feature detectors and transforming into robust SIFT or GLOH descriptors [6] a first rank performance of 85% for full impressions and 84% for partial impressions is obtained using a database containing 368 different footwear patterns provided by the UK National Shoewear Database. A combination of the Modified Harris-Laplace (MHL) detector and the enhanced SIFT descriptor [7] is more robust to rotation and inverse contrast and is fast but not accurate enough, while a more accurate matching strategy based on RANSAC [8] is time-consuming, also only applicable to noiseless data. The use of the Wavelet-Fourier transform [9] presents significant improvements using a huge private database of more than 200.000 real crime images. One of the newest approaches uses convolutional neural networks and a multi-variate cross validation [10]. The Vector of Locally Aggregated Descriptors (VLAD) [10,11,13] has proven to be a useful low dimensional image descriptor, especially with large image datasets and thereby avoid expensive hard disk access.

Recent work on deep learning has demonstrated that local feature descriptors based on convolutional neural networks (CNN) can significantly improve the matching performance. Previous work has focused on exploiting pairs of positive and negative patches to learn discriminative CNN representations, like the Siamese Network [24] where the network is trained to distinguish between similar and dissimilar pairs of examples. The most recent investigations indicate that the use of triplets [14,15] of training samples instead of solely focusing on pairs shows benefits in terms of performance and speed.

2.1. Scale-Invariant Feature Transform

The Scale-Invariant Feature Transform (SIFT) is an algorithm to detect and describe local features in images. First, SIFT algorithm creates a scale space, taking the original image and generating progressively blurred out images, resizing them to obtain a gaussian pyramid (figure 3). Each octave's image size is half the previous one. Then creates the difference of gaussians pyramid (approximately equivalent to the Laplacian of Gaussian, but computationally faster and also scale invariant) for finding interesting points (key points). This task is done by iterating through each pixel of the DOG (Difference of Gaussian) image and checking all its neighbours, to find the maxima and minima. A pixel is marked as a key point if it is the greatest or least of the neighbouring pixels in the current scale, the scale above and the scale below (figure 4).

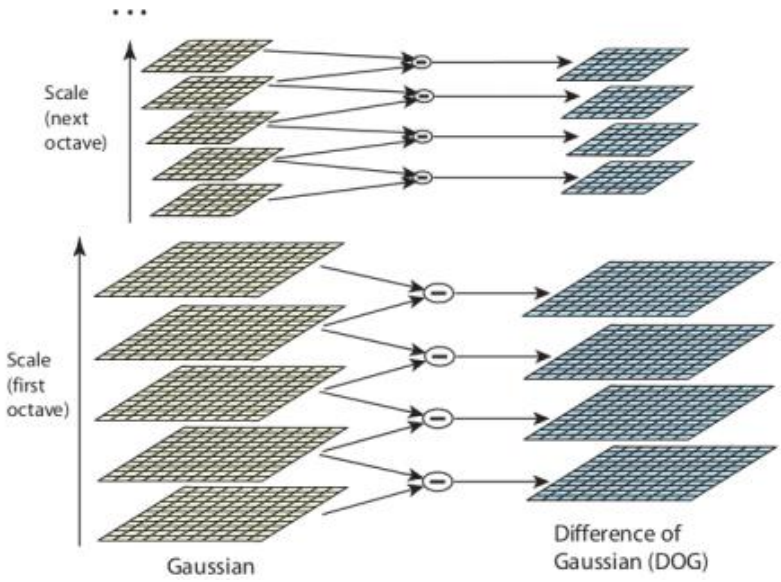

Fig 3. Gaussian pyramid used in SIFT algorithm. [21]

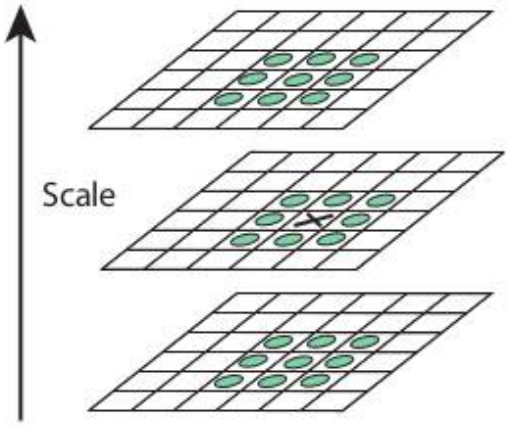

Fig 4. Key point search. [21]

Some key points are not discriminative and removing them allows the algorithm to work more efficiently and robustly. For removing features located on an edge, a similar Harris Corner Detector approach is used. For low contrast features, their intensities are checked. After that, an orientation is calculated for each key point, making it rotation invariant. This process is done collecting gradient directions and magnitudes around each key point.

Finally, a unique key point descriptor (feature vector) is created. A 16x16 neighbourhood around the key point is taken. Then, it is divided into 16 sub-blocks of 4x4 size. For each sub-block, 8 bin orientation histogram is created (a total of 128 bin values are available), receiving the name of Histogram of Oriented Gradients (HOG).

Some examples applied to reference images (figure 5) and their correspondent real crime scene shoeprint (figure 6) are shown.

Fig 5. Key points in reference images.

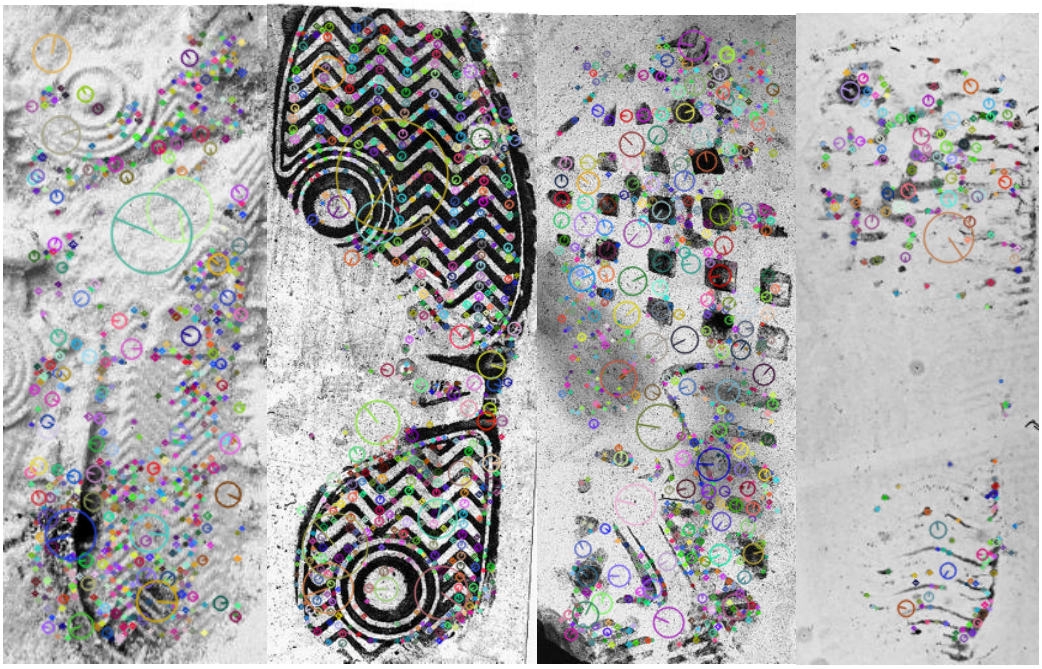

Fig 6. Key points in real crime scene images.

The Dense SIFT approach tries to simplify the algorithm using a sliding window and computing the descriptor over dense grids in the input image, getting more information than corresponding descriptors evaluated at a much sparser set of image points obtained by an interest operator. Usually this technique is accompanied with a clustering stage, reducing the individual SIFT descriptors to a smaller vocabulary of visual words.

2.2. Vector of Locally Aggregated Descriptors

The VLAD (Vector of Locally Aggregated Descriptors) descriptor encodes a set of local feature descriptors extracted from an image, using a dictionary built using a clustering method such as Gaussian Mixture Models (GMM) or K-means clustering and produces a fixed-length vector representation.

The idea of the VLAD descriptor is to accumulate for each visual word c_i the differences $x-c_i$ of the vectors x assigned to c_i., i.e. by accumulating the residual vectors (the difference between the descriptor and the centroid). This characterizes the distribution of the vectors with respect to the centre. For the kth cluster centre μ_k, the corresponding VLAD feature is calculated as the sum of the residuals as

$$v_k = \sum_{i=1}^{N} \alpha_{ik}(x_i - \mu_i)$$

where x_i is the set of local features from an image and α_{ik} is the association of data x_i to μ_k. In this step, the Approximate Nearest Neighbour (ANN) method is used. It means that each descriptor is then assigned to the closest cluster of a vocabulary.

Finally, various normalizations methods can be applied to the VLAD vectors, for example the component-wise mass (each vector v_k is divided by the total mass of features associated to it), the square-rooting (applies the function sign(x)* $\sqrt{|x|}$ to all scalar components of the descriptor), the component-wise L2 (the vectors v_k are divided by their norm) or the global L2 normalization (the VLAD descriptor is divided by its norm), which was suggested in the original approach.

The dimension of the VLAD descriptor is KxD, where K is the number of centres used in the K-means clustering and D is the size of the local input descriptor (for example, 128 in the SIFT case).

Recently a multi-VLAD approach has been proposed, investigating the benefits of combining multiple vocabularies, instead of solely representing the image by a single VLAD. Also using PCA (principal component analysis) and whitening to decorrelate a low dimensional representation presents significant improvements without noticeably impacting its accuracy.

2.3. Convolutional Neural Networks and Triplets

Convolutional Neural Networks (CNN) are deep artificial neural networks consists of a number of convolutional and subsampling layers followed by fully connected layers (see figure 7. These images are processed as tensors, matrices of numbers with additional dimensions (width, height and depth). Its effectiveness has been proven in areas such as image recognition and classification [27].

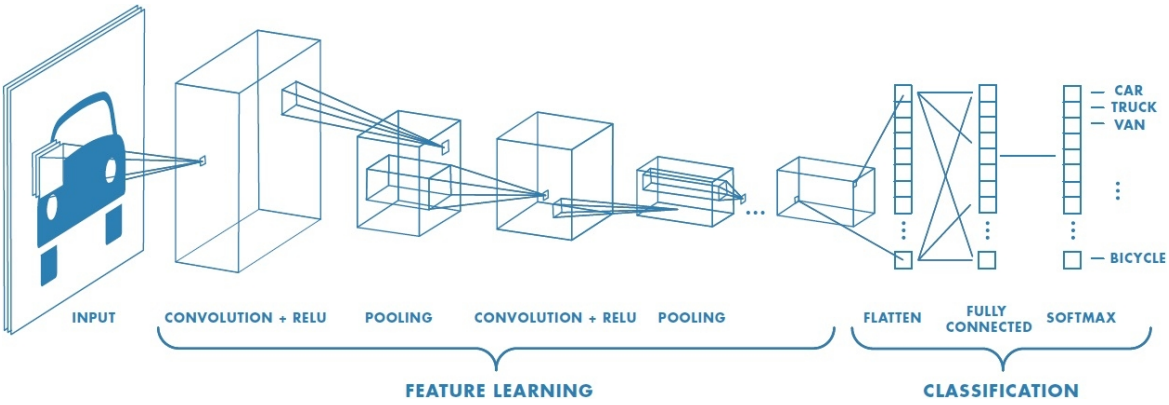

Fig 7. Example of Convolutional Neural Network. [19]

The convolutional layer computes the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume. The ReLU (rectified linear units) layer applies an elementwise activation function, which is f(x)=max(0,x), setting any negative element to 0. The pool layer performs a down sampling operation along the spatial dimensions (width, height). Finally, the fully connected layer computes the class scores, resulting in volume of size [1x1xN], where each of the N numbers correspond to a class score. To guide the training process of a neural network the loss function is used.

A typical approach on deep learning of feature descriptors is based on the siamese networks (figure 8, top), which consist of two CNNs which accept two parallel inputs and share parameters across networks. The loss function is optimized based on the output of the two networks according to their distinct inputs.

A more recent approach using three parallel inputs has been investigated [14,15], where two of them are positive patches from two views of the same point in the 3D space, and the third one is a negative patch extracted from a different point in space. They receive the name of triplets (figure 8, bottom).

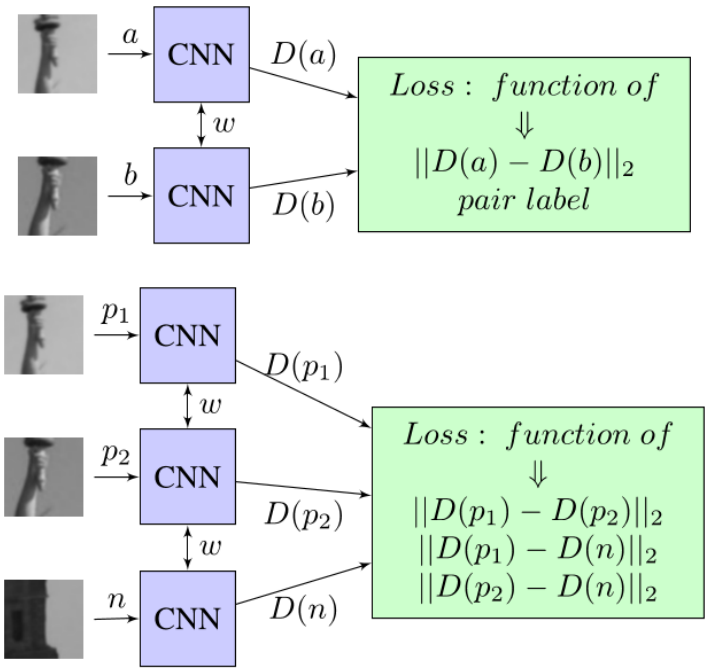

Fig 8. Difference between siamese and triplet architecture. [15]

In this approach, the three instances have the same feed-forward network and share parameters. The triplet network is fed with three inputs, which we can define as x^+, x^- and the reference x. Both x and x^+ pertains to the same class, while x^- pertains to another different class. Two outputs are obtained, which are the encoded pair of distances between each of x^+ and x^- against the reference x.

3. Methodology / project development:

Firstly, a pre-processing is applied to the images in order to facilitate the task of classification and to make the features more stable, transforming the colour space from RGB to greyscale (colours have no meaning for us) and optionally applying a binary threshold with the Otsu algorithm (choose the optimal threshold value). Then, the train images (which corresponds with the reference shoeprints database) are analyzed, searching for the key points in each one. In this step, the implemented methods depend on the approach we focus on, extracting the SIFT descriptors as is described in [21], using the approach in [16] to implement the Dense-SIFT algorithm or using the Triplets to create a Convolutional Neural Network as is shown in [15]. All these descriptors are normalized.

With these descriptors the VLAD is trained and the vocabulary is created, forming the different clusters using the K-means technique, as is described in [11].

The next step is to calculate the descriptors for each test image (which corresponds with the real crime scene shoeprints database), and encode it using the previous trained VLAD encoder to obtain the VLAD features vector, which have fixed length for all the images. In this process, the difference between the prediction and the centre of the clusters is added. A power normalization is applied to the resulting vector.

Then, the Euclidean distance between the VLAD features of each crime scene image and each reference image is calculated. These distances are sorted and ranked to obtain the list of most likely matchings, arranged in a matrix of dimension NxM where N is the number of test images (crime scene prints) and M the number of train images (references).

In order to evaluate the precision of the system, the Cumulative Matching Characteristics (CMC) curve is calculated to determinate the rank at which a true match occurs. For it is necessary to create the ground truth matrix, where for each test image, all the elements are zero except the correct one, which is a one. Then, we perform a cumulative sum of the good matchings through all the database. An example of the most likely matching matrix and the ground truth matrix are shown in the table 1.

Test image	**Reference image**				
1	30	145	8	**1**	72
2	44	**2**	203	55	13
3	**3**	83	100	94	22

Test image	**Ground truth**				
1	0	0	0	**1**	0
2	0	**1**	0	0	0
3	**1**	0	0	0	0

Table 1. Most likely matching matrix (left) and ground truth matrix (right) example.

4. Results

To illustrate the performance of the different studied approaches, an empirical research is presented in this section. The experiments have been carried out on an Intel Core i5-4690 and a GeForce GTX 980 (only in the deep learning approach).

4.1. SIFT using OpenCV

In this first approach the image features are extracted using the OpenCV (Open Source Computer Vision Library) [12] methods to obtain the SIFT descriptors. For each image, we obtain a different number of key points (it depends on the difference of gaussians and the gradients, some images have more points of interest), but all the descriptors have 128 dimension. We train the VLAD with 50 centres, obtaining fixed feature vectors of length 6400 (128 x 50) per image.

The results are shown in the figure 9. The blue line represents the matching score in a random case and the red line shows the current results. As we can see, these results are not useful enough, slightly better than in the random situation.

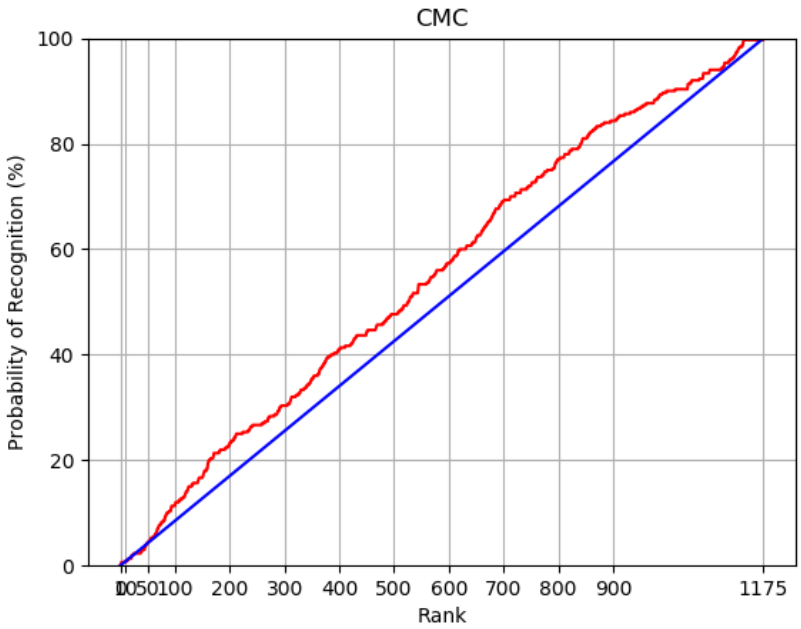

Fig 9. CMC using SIFT descriptors + VLAD with 50 centres

Aiming to improve the performance of the system, the number of centres in the VLAD encoder is doubled to 100, obtaining a fixed feature vectors of length 12800. This change increases the processing time but it hardly improves the results, as we can see in the figure 10. Therefore, this algorithm seems not to be effective, taking key points without real interest for our objective.

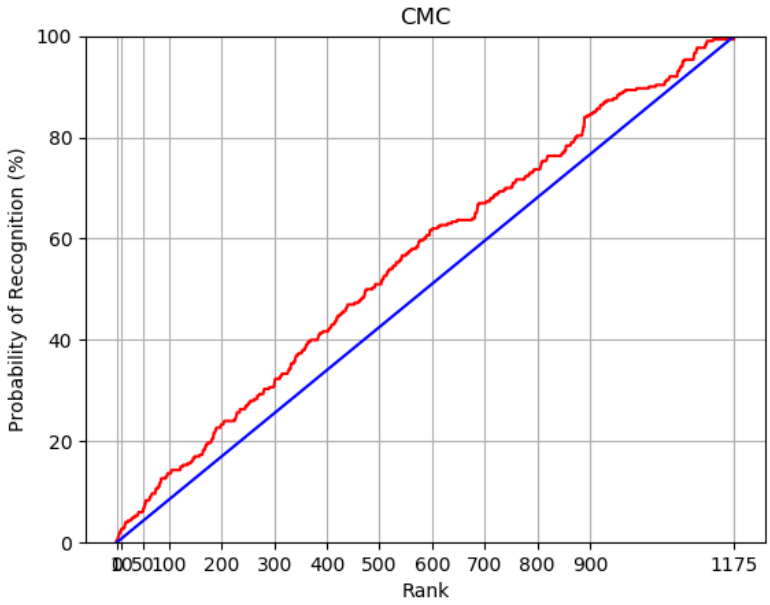

Fig 10. CMC using SIFT descriptors + VLAD with 100 centres

4.2. Dense SIFT using VLFeat

In this second approach the image features are extracted using the VLFeat (Vision Lab Features Library) [17] adaptation to Python (CyVLFeat) [18]. A different version of SIFT is used, called Dense SIFT. The main difference is that the descriptors are not calculated through an algorithm in all the key points of the image but are calculated through a sliding window at every location. This method allows us to control the window size and the sliding steps, and also is supposed to be faster.

For a first attempt, we select a window size of 10 and a step size of 3. The VLAD encoder is set with 50 centres, due to as we have observed in the previous experiment, increasing the number of centres seems to increase the processing time without improving the performance of the system. The results in this case are shown in the figure 11. As we can see, the obtained results are a better than the obtained in the first try using normal SIFT descriptors, but not good enough.

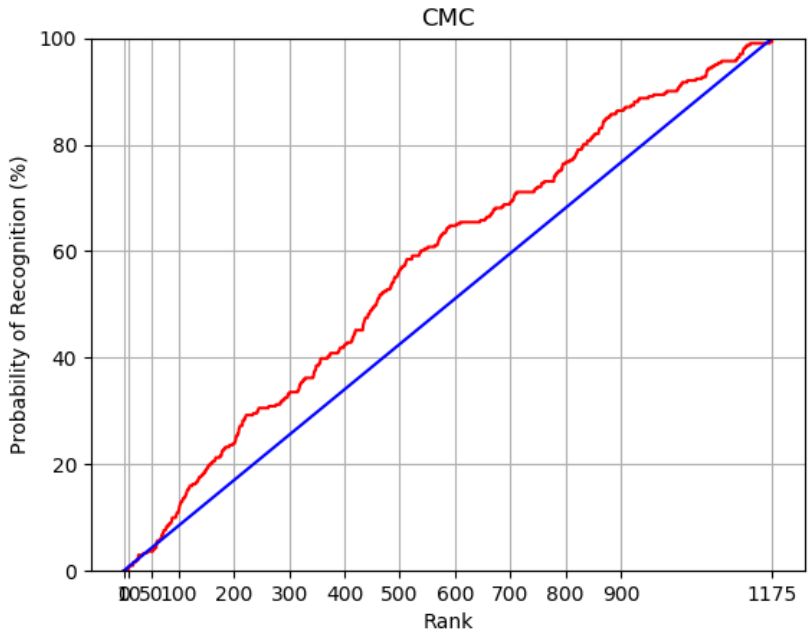

Fig 11. CMC using Dense SIFT (window size 10) descriptors + VLAD

Aiming to improve the results, the window size is doubled to 20, but keeping the step size of 3. The results are shown in the figure 12. In this case, the results are improved considerably except for some images. It will be useful to investigate these images to know the cause of the problem and identify which type of patterns and shapes are difficult to match.

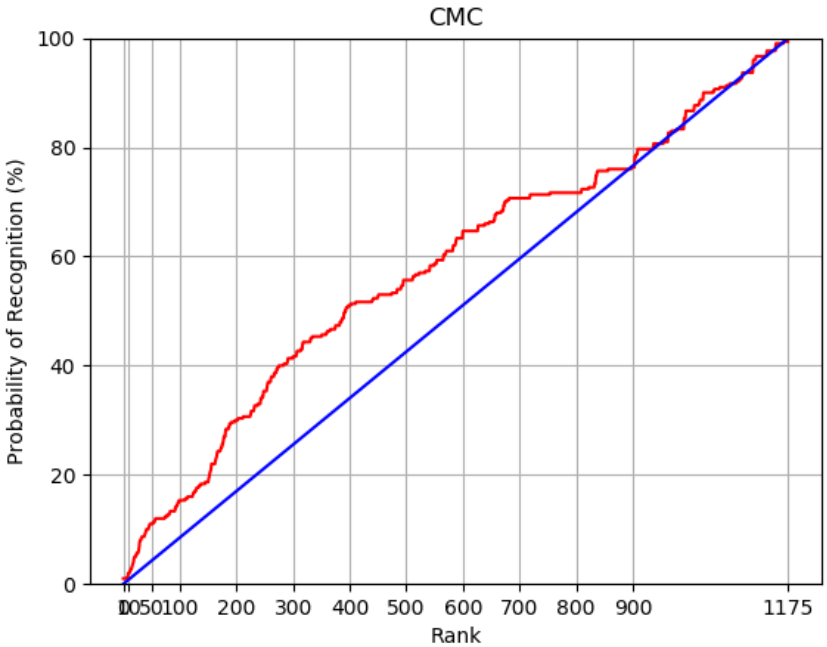

Fig 12. CMC using Dense SIFT (window size 20) descriptors + VLAD

Analysing the ~20% of the test dataset that presents difficulties to match properly the crime scene shoeprints and also the correspondent reference images, some patterns likely to produce faults are found. Particularly striking are the footwear impressions in the snow (figure 13), which always have a low recognition rank. Probably this fact is due to the remarkable difference between the original reference (or, in general, the rest of shoeprint crime scenes) and this type of shoeprint with depth component.

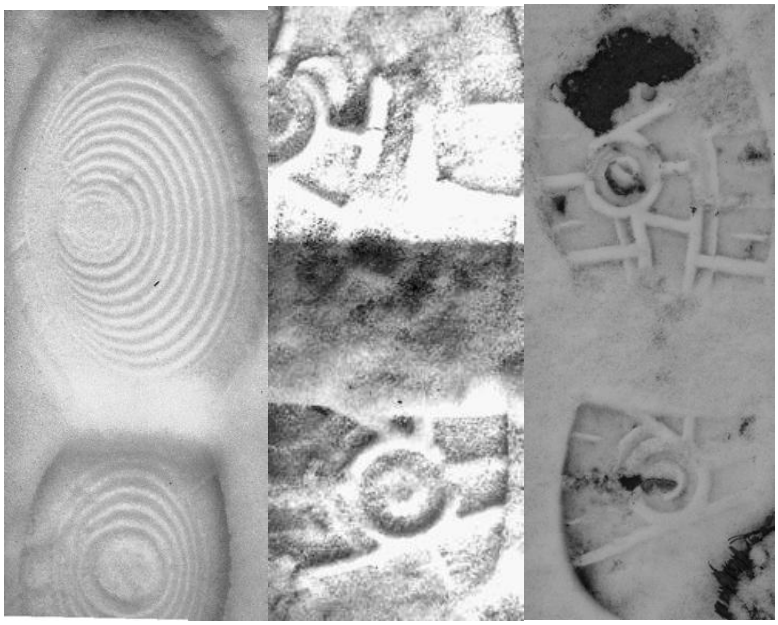

Fig 13. Snow shoeprints.

An interesting analysis is to check, for this crime scene shoeprints, which reference footwear impressions have the highest probability of matching, trying to know why the system does not recognize the patterns correctly. For some crime scene prints, we show the chosen five more probably by the system (figure 14). As we can observe, the system tries to match the first two shoeprints following a granular pattern, with points or little circles. It could be due to the structure of the snow, producing matching errors. In the last shoeprint this does not happen, and the matched reference images are more similar than in the previous impressions, probably due to the contrast.

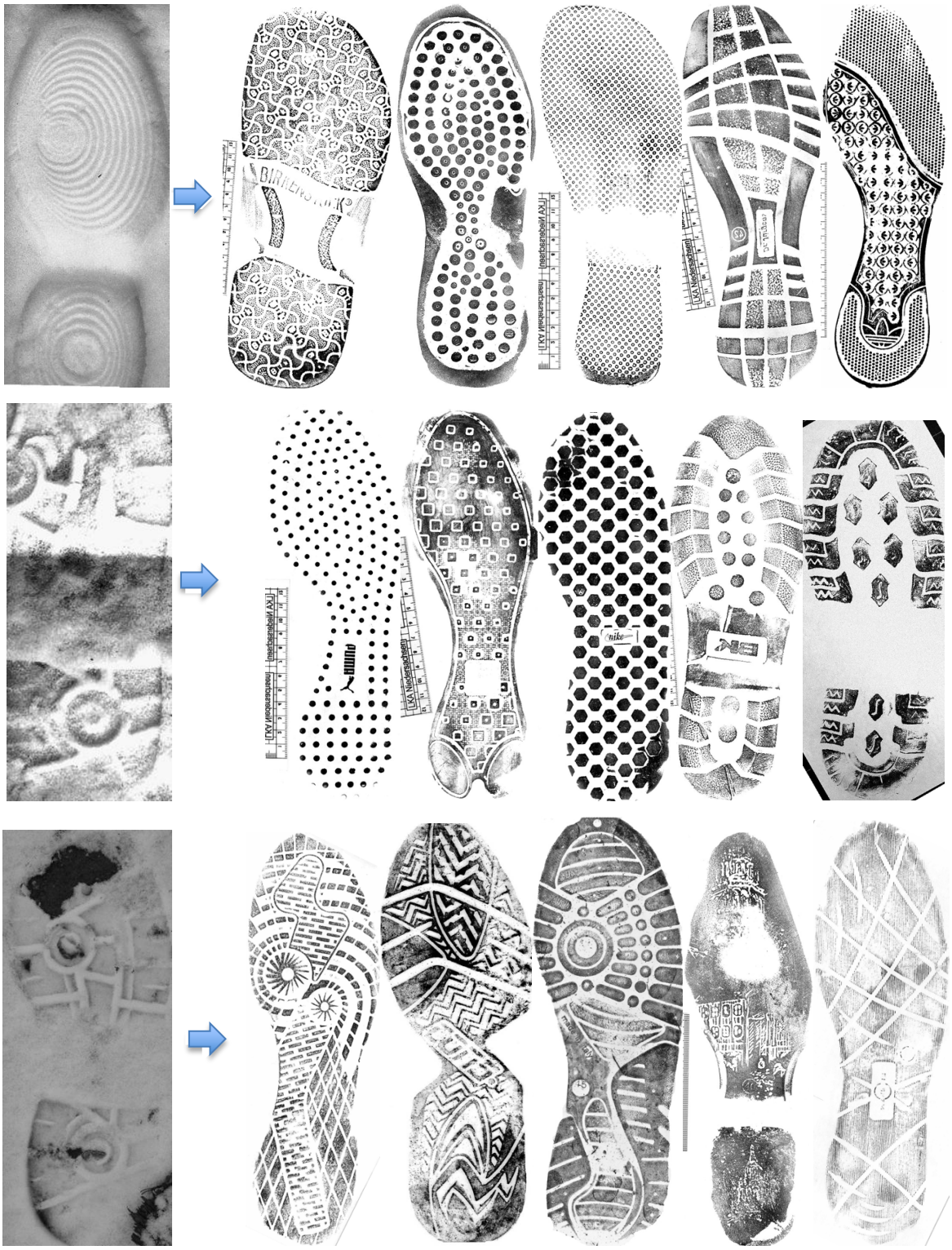

Fig 14. References with highest probability of matching.

Other shoeprints that present recognition difficulties are shown in the figure 15. As is observed, these footwear impressions present highly unconstrained noise and low contrast conditions, making the task of recognition difficult.

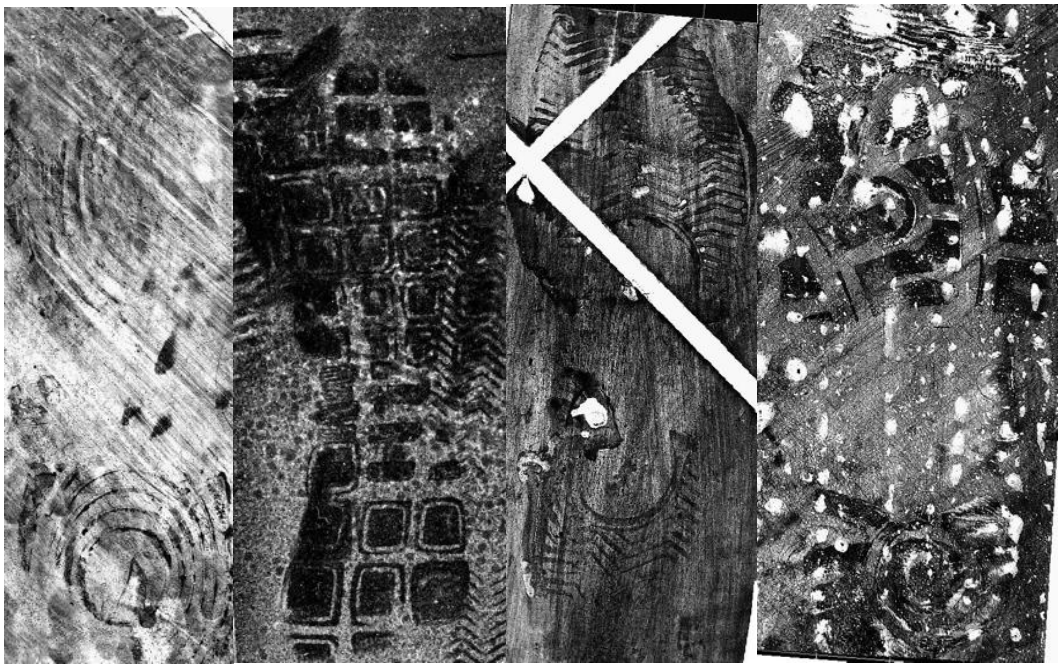

Fig 15. Shoeprints with low recognition rank.

Another interesting analysis is to check which reference images usually present complications to be matched. In the figure 16 are shown the shoeprints with the lowest probability of recognition. At first sight, we can find some similarities between the three shoeprints on the right, showing a likeness that could cause system problems. It is important to remember the limitation of the database, where there are no real crime scene images for each reference image. Therefore, it is difficult to draw conclusions about which patterns are less likely to be recognized from the available resources.

Fig 16. Reference images with low recognition rank.

To reinforce the system is intended to create a new database division, introducing some real crime scene shoeprints in the training set and also reference images in the test set, obtaining two 50-50 sets. The division is carried out in such a way that all the real shoeprints

in the test set have the corresponding matching in the training set, where only the real crime scenes are evaluated. The experiment is repeated using the Dense SIFT with a sliding window of size 20, obtaining the results shown in the figure 17. As is observed, the results are slightly better using this mixed dataset, but they can vary depending on the initial random split.

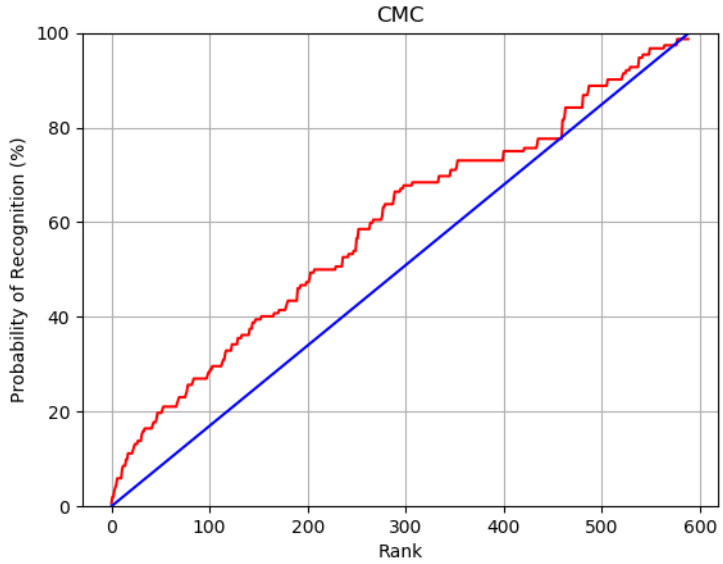

Fig 17. *CMC using Dense SIFT (window size 20) descriptors + VLAD in a mixed dataset*

4.3. Triplet using PyTorch

In this third approach the image features are extracted using PyTorch [22], a deep learning framework that provides a tensor computation with strong GPU acceleration. As has been explained, a Convolutional Neural Network (pre-trained on photo tour dataset [28]) is used to extract the image features which will be encoded with the VLAD encoder. As it has been explained, the number of centres in the VLAD is set to 50 due to the compromise between performance and processing time.

In the first attempt a window of size 16 is used, obtaining the results shown in the figure 18. As is observed, using such a small patch size the results are almost random and therefore are not useful for our purpose.

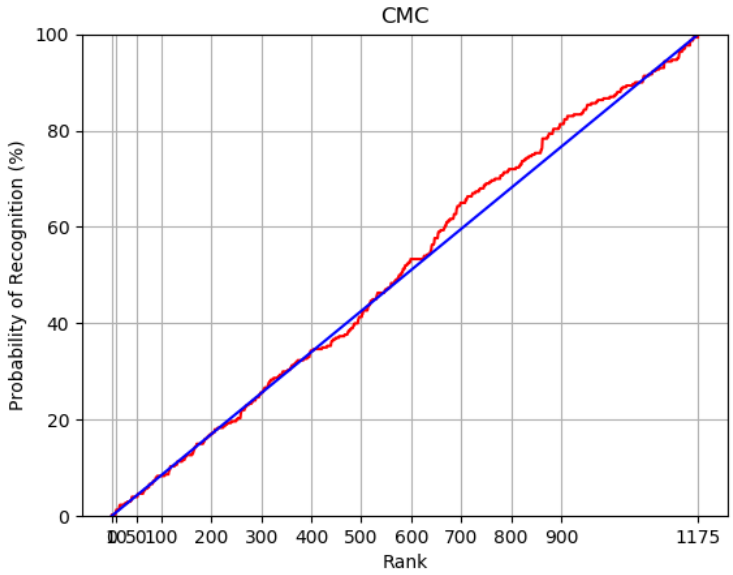

Figure 18. CMC using Triplets (window size 16) + VLAD

In the second attempt the window size is doubled to 32, trying to improve the results and check if the Convolutional Neural Network needs a larger patch size to identify correctly the pattern of the shoeprint. These results are shown in the figure 19, where we can observe how an improvement has occurred. Even so, the use of deep learning techniques has not improved significantly the effectiveness of the system.

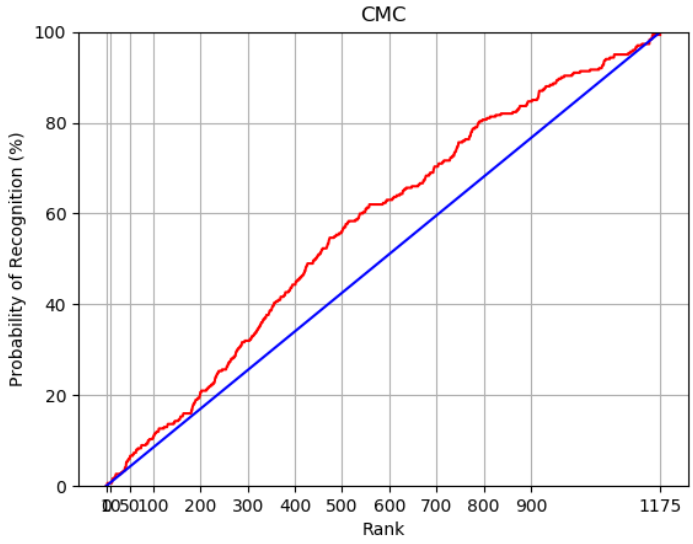

Figure 19. CMC using Triplets (window size 32) + VLAD

In the third attempt the window size is doubled again to 64, aiming to check if bigger is better or if the previous size is the optimal to get the best performance. As we can see in the figure 20, the results are worse in comparison to the previous attempt, demonstrating that this patch size is too big to produce a good pattern matching.

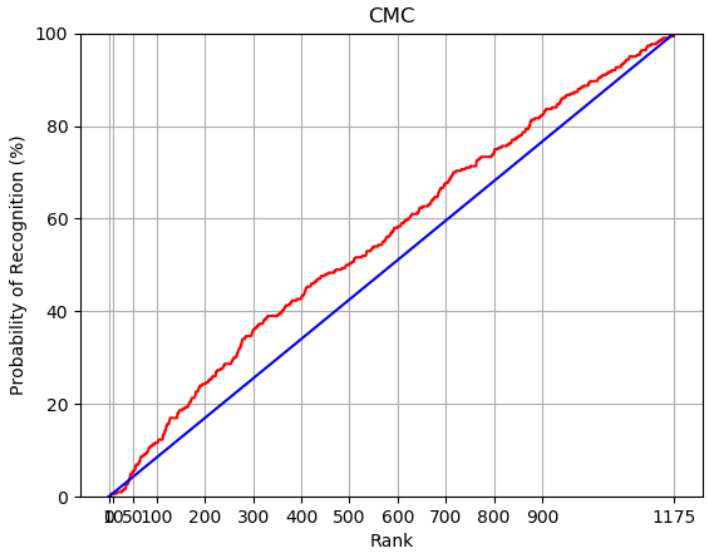

Figure 20. CMC using Triplets (window size 64) + VLAD

As in the previous experiment, the database is split in two new sets containing both reference and crime scene images, aiming to improve the performance and to obtain a more robust system.

The experiment is repeated using the Triplet CNN with a window size of 32, which we have observed is the optimal value to obtain a good performance. The results are shown in the figure 21, where we can observe that are very similar to the obtained in the original database, meaning that can not properly cope with different looking images.

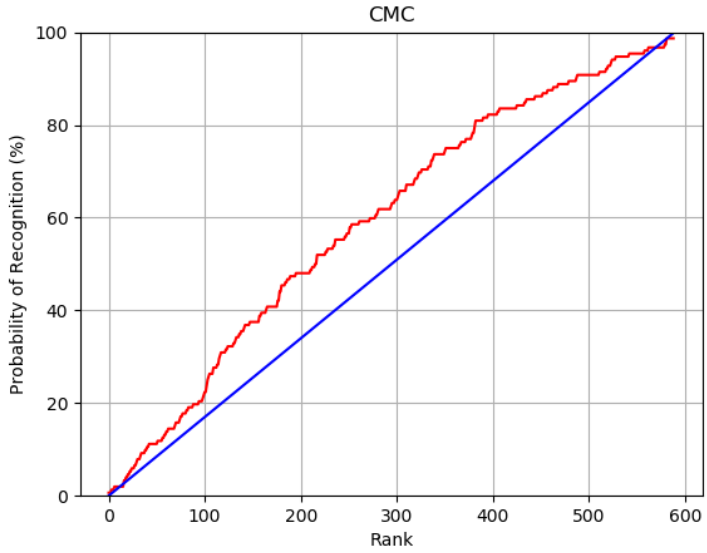

Fig 21. CMC using Triplets (window size 32) + VLAD in a mixed database

5. Conclusions and future development:

In this project we have implemented different techniques aiming to evaluate the work of a pattern matching system, focused on the recognition of footwear impressions. The project is applied to real crime scene shoeprints, a field without an excessive previous work and therefore, with a wide margin of improvement and future study.

The main techniques used to carry out the project have been the SIFT descriptors, the VLAD encoder and the triplet CNN. All these methods accept variations in the parameters, allowing modifications in the performance and originating different results, sometimes more significant than other times, as we had observed and we will discuss.

A summary of all the obtained results, expressed in form of Cumulative Match Characteristic curve, is shown in the figure 22.

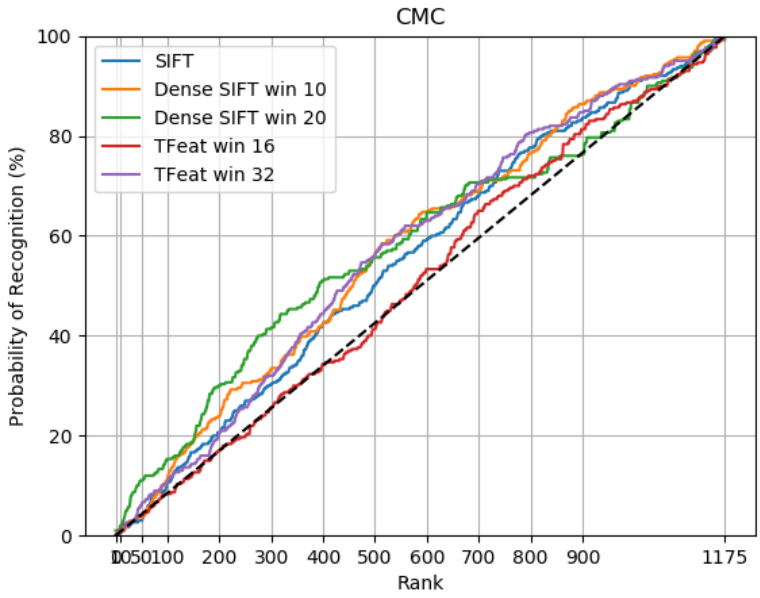

Fig 22. CMC with all the evaluated results.

After the evaluation of the different methods and some of their parameters, we can conclude that the use of Dense SIFT descriptors combined with the VLAD encoder for the task of the pattern matching of footwear impressions presents the best results in terms of performance but the SIFT approach achieves the fastest performance (the table 2 shows processing times). In order to obtain the maximum performance of the Dense SIFT algorithm, we suggest to use a window of size 20 and a VLAD encoder with 50 centres, because it has not been proven that increase this number improves the results and it only increases the processing time.

	First attempt	Second attempt	Third attempt
SIFT	9225.514826274943	10402.529093477875	-
Dense SIFT	57081.12756513618	42703.613905509	-
Triplet	43416.57754942821	45616.72697753692	33722.369546536356

Table 2. Processing time in seconds. SIFT first attempt (VLAD with 50 centres), SIFT second attempt (VLAD with 100 centres), Dense SIFT first attempt (window size 10), Dense SIFT second attempt (window size 20), Triplet first attempt (window size 16), Triplet second attempt (window size 32), Triplet third attempt (window size 64).

The Triplet Convolutional Neural Network also provides a good performance, with results that compete with those obtained in the Dense SIFT approach. After evaluate different patch sizes, we can conclude that the optimal value is 32. In the figure 23 we can observe the patch obtained with a window size of 16x16, 32x32 and 64x64. As we can see, the use of a small window size does not provide enough information about the pattern. In the other hand, using a big window size does not define properly the determined pattern, taking more information than necessary.

Fig 23. Example of patches of size 16 (left), size 32 (centre) and size 64 (right).

A comparison between the Dense SIFT and Triplet approaches is shown in the figure 24, in order to evaluate more accurately the two studied methods with best performance in the project. As we can see, the Dense SIFT has a better beginning of the work, recognizing faster than the Triplet the first patterns. But it presents problems to match some of the shoeprints in the end, decreasing the performance unlike the Triplet approach which is more stable even though it has a worse starting.

Something similar happens when the database is split in two subsets containing both reference and real crime scene images, as we can see in the figure 25. In this case, the use of the Triplet Convolutional Neural Network provides a more stable system to match the footwear impressions.

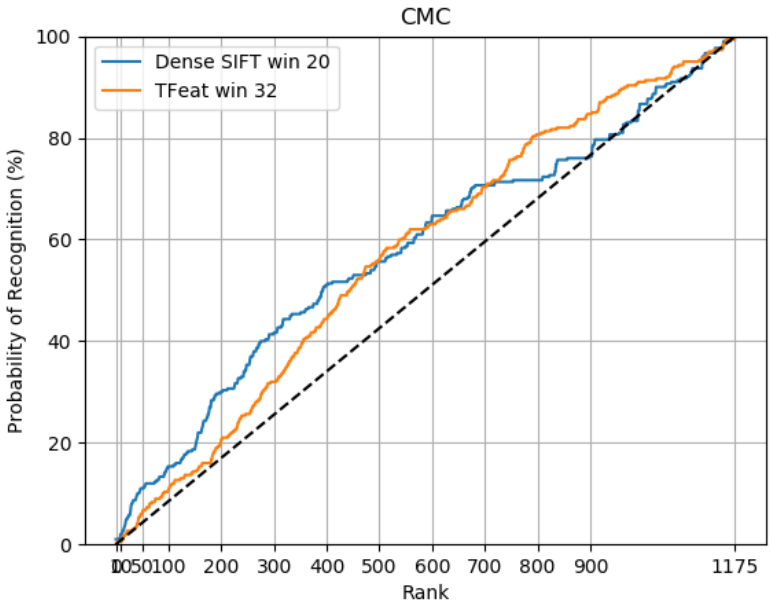

Fig 24. CMC comparison between Dense SIFT and Triplet

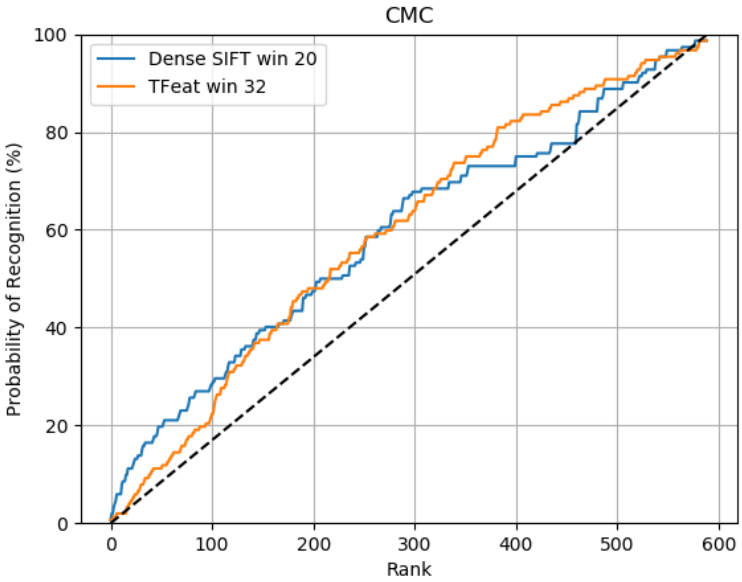

Fig 25. CMC comparison between Dense SIFT and Triplet in a mixed dataset

To sum up, in this project we have achieved results with room for improvement but quite corresponding to the state of the art in the footwear impression matching field. The descriptors used in the system are invariant to rotation, translation and scale, which are a critical point when we are working on a real crime scene database. The use of the Dense SIFT descriptors is comparable to the use of the Triplet Convolutional Neural Networks, obtaining in both approaches the best results.

To keep working on this topic in the future, one of the most important requirement is to continue increasing the available datasets, especially the real crime scene shoeprints database because this is one of the biggest limitations to work, making hard the extraction of more reliable conclusions. It is important to note that the standard databases used in machine learning contain thousands or millions of images. Other techniques to extract the main features and obtain the descriptors could be studied and evaluated in detail. Training the Convolutional Neural Network with the shoeprints database would be a great start. Also, the script could be optimized, evaluating new algorithms in order to improve the processing time and the use of the CPU.

Bibliography:

[1] Adam Kortylewski, Thomas Albrecht, Thomas Vetter. *Unsupervised Footwear Impression Analysis and Retrieval from Crime Scene Data. ACCV 2014, Workshop on Robust Local Descriptors.*

[2] Philip De Chazal, John Flynn, and Richard B Reilly. *Automated processing of shoeprint images based on the fourier transform for use in forensic science. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 27(3):341–350, 2005.*

[3] Mourad Gueham, Ahmed Bouridane, Danny Crookes, and Omar Nibouche. *Automatic recognition of shoeprints using fourier-mellin transform. In Adaptive Hardware and Systems, 2008. AHS'08. NASA/ESA Conference on, pages 487–491. IEEE, 2008.*

[4] H Su, D Crookes, A Bouridane, and M Gueham. *Shoeprint Image Retrieval by Topological and Pattern Spectra. In International Machine Vision and Image Processing Conference, 2007.*

[5] Gharsa AlGarni and Madina Hamiane. *A novel technique for automatic shoeprint image retrieval. Forensic science international, 181(1):10–14, 2008.*

[6] Maria Pavlou and Nigel M Allinson. *Automatic extraction and classification of footwear patterns. In Intelligent Data Engineering and Automated Learning– IDEAL 2006, pages 721–728. Springer, 2006.*

[7] H Su, D Crookes, A Bouridane, and M Gueham. *Shoeprint Image Retrieval Based on Local Image Features. In Third International Symposium on Information Assurance and Security, 2007.*

[8] Omar Nibouche, Ahmed Bouridane, D Crookes, M Gueham, et al. *Rotation invariant matching of partial shoeprints. In Machine Vision and Image Processing Conference, 2009. IMVIP'09. 13th International, pages 94–98. IEEE, 2009.*

[9] Xinnian Wang, Huihui Sun, Qing Yu, and Chi Zhang. *Automatic Shoeprint Retrieval Algorithm for Real Crime Scenes. Dalian Maritime University, Dalian, China, 2015.*

[10] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, et al. *Aggregating local image descriptors into compact codes. IEEE Transactions on Pattern Analysis and Machine Intelligence, Institute of Electrical and Electronics Engineers, 2012, 34 (9), pp.1704-1716. <10.1109/TPAMI.2011.235>. <inria-00633013>*

[11] R. Arandjelovic and A. Zisserman, *"All About VLAD," 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, 2013, pp. 1578-1585. doi: 10.1109/CVPR.2013.207*

[12] OpenCV 3.0.0-dev documentation. *Introduction to SIFT (Scale-Invariant Feature Transform)* [Online] Available: *https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html*

[13] Jingxiao Zheng, J. C. Chen, N. Bodla, V. M. Patel and R. Chellappa, *"VLAD encoded Deep Convolutional features for unconstrained face verification," 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, 2016, pp. 4101-4106.*

[14] Vassileios Balntas, Edgar Riba, Daniel Ponsa and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In Richard C. Wilson, Edwin R. Hancock and William A. P. Smith, editors, Proceedings of the British Machine Vision Conference (BMVC), pages 119.1-119.11. BMVA Press, September 2016.

[15] Balntas, Vassileios; Johns, Edward; Tang, Lilian; Mikolajczyk, Krystian. *PN-Net: Conjoined Triple Deep Network for Learning Local Image Descriptors. Eprint arXiv:1601.05030, January 2016.*

[16] J. G. Wang, J. Li, C. Y. Lee and W. Y. Yau, *"Dense SIFT and Gabor descriptors-based face representation with applications to gender recognition," 2010 11th International Conference on Control Automation Robotics & Vision, Singapore, 2010, pp. 1860-1864.*

[17] VLFeat documentation. *Dense SIFT as a faster SIFT.* [Online] Available: *http://www.vlfeat.org/overview/dsift.html*

[18] CyVLFeat github repository. [Online] Available: https://github.com/menpo/cyvlfeat

[19] Mathworks. *Introduction to Deep Learning: What are Convolutional Neural Networks?* [Online] Available: https://es.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html

[20] Computer Vision Lab, TU Wien. *Pattern Matching of Footwear Impressions.* [Online] Available: https://cvl.tuwien.ac.at/teaching/diplomarbeiten/pattern-matching-of-footwear-impressions/

[21] David G. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints. In International Journal of Computer Vision, 2004.*

[22] Pytorch. [Online] Available: https://pytorch.org

[23] Bouridane, A & Alexander, A & Nibouche, Mokhtar & Crookes, D. (2000). *Application of fractals to the detection and classification of shoeprints. Proceedings / ICIP ... International Conference on Image Processing. 1. 474 - 477 vol.1. 10.1109/ICIP.2000.900998.*

[24] Bromley, Jane, Bentz, James W, Bottou, Léon, Guyon, Isabelle, LeCun, Yann, Moore, Cliff, Säckinger, Eduard, and Shah, Roopak. *Signature verification using a siamese time delay neural network. International Journal of Pattern Recognition and Artificial Intelligence, 7(04):669–688, 1993.*

[25] G. Alexandre. *Computerised classification of the shoeprints of burglars soles. Forensic Science International, 82:59–65, 1996.*

[26] Karl Rupp. *40 Years of Microprocessor Trend Data.* [Online] Available: https://www.karlrupp.net/2015/06/40-years-of-microprocessor-trend-data/

[27] Waseem Rawat & Zenghui Wang. *Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. Neural Computation 2017 29:9, 2352-2449.*

[28] School of Computer Science & Engineering, University of Washington. *Photo Tourism. [Online] Available: http://phototour.cs.washington.edu*

Glossary

SIFT: Scale-Invariant Feature Transform

VLAD: Vector of Locally Aggregated Descriptors

CNN: Convolutional Neural Network

MSER: Maximally Stable Extremal Regions

CPU: Central Processing Unit

GPU: Graphics Processing Unit

CMC: Cumulative Matching Characteristic

ReLU: Rectified Linear Unit

IDE: Integrated Development Environment

GMM: Gaussian Mixture Model

HOG: Histogram of Oriented Gradients

MHL: Modified Harris-Laplace

DOG: Difference of Gradients

RANSAC: Random Sample Consensus