A baseline pipeline for pedestrians analysis in public environments

University of Modena and Reggio Emilia
Computer Vision & Cognitive System course, 2021/2022

Casari Giovanni - 253372@studenti.unimore.it
Martinello Pietro - 257261@studenti.unimore.it
Zini Leonardo - 258378@studenti.unimore.it

February 2023

Abstract – **In recent decades important economic and demographic events have led to a major increase in the world's population. This, coupled with the widespread trend of movement from rural to urban areas, has encouraged the creation of particularly populous environments where surveillance, whether public or private, can be problematic. With the advent of digitalization, numerous security cameras have begun to appear in urban centers, particularly in busy areas. This research aims to provide an initial tool for improving the productivity of current visual systems available for surveillance in such populated areas. In fact, the use of artificial intelligence models could lead to substantial time savings to the operator in following the movements of the individuals being viewed, and in the eventual investigation regarding their identity.**

1 Introduction

The implementation of an intelligent surveillance system could include several options. Some examples might be pedestrian tracking, pose analysis, study of individual behaviors within complex systems, retrieval of personal information.

Our work aimed to focus on three distinct points:

1. **Tracking of pedestrians**, i.e., identification of the same identities within one or more video streams made in a medium to short time interval. For each identity is also possible to visualize the temporal position evolution;

2. The possibility of using **super resolution** techniques to provide an enhanced visual representation of an individual compared to reality;

3. **Image retrieval**, hence the ability to trace a person's identity over a long-term scenario using a database containing images of people from past camera recordings.

Thus, given one or more video streams coming from surveillance cameras placed in public environments, the final goal of this work is to provide a pipeline which is able to label each visible pedestrian with an unique identifier, showing to the user the evolution of the spatial location of identities frame by frame. Furthermore, user will be able to select and to enhance a pedestrian's crop from a given frame, and to search the same person in an external database. In the following pages the term 'tracking' will mean the act of following pedestrians passively, therefore without an active prediction of future movements.

2 Related works

Being our research a pipeline of multiple blocks, we decided to divide the problem into smaller sections, studying each one independently. This section wants to give an overview of current state of the art solutions for each problem.

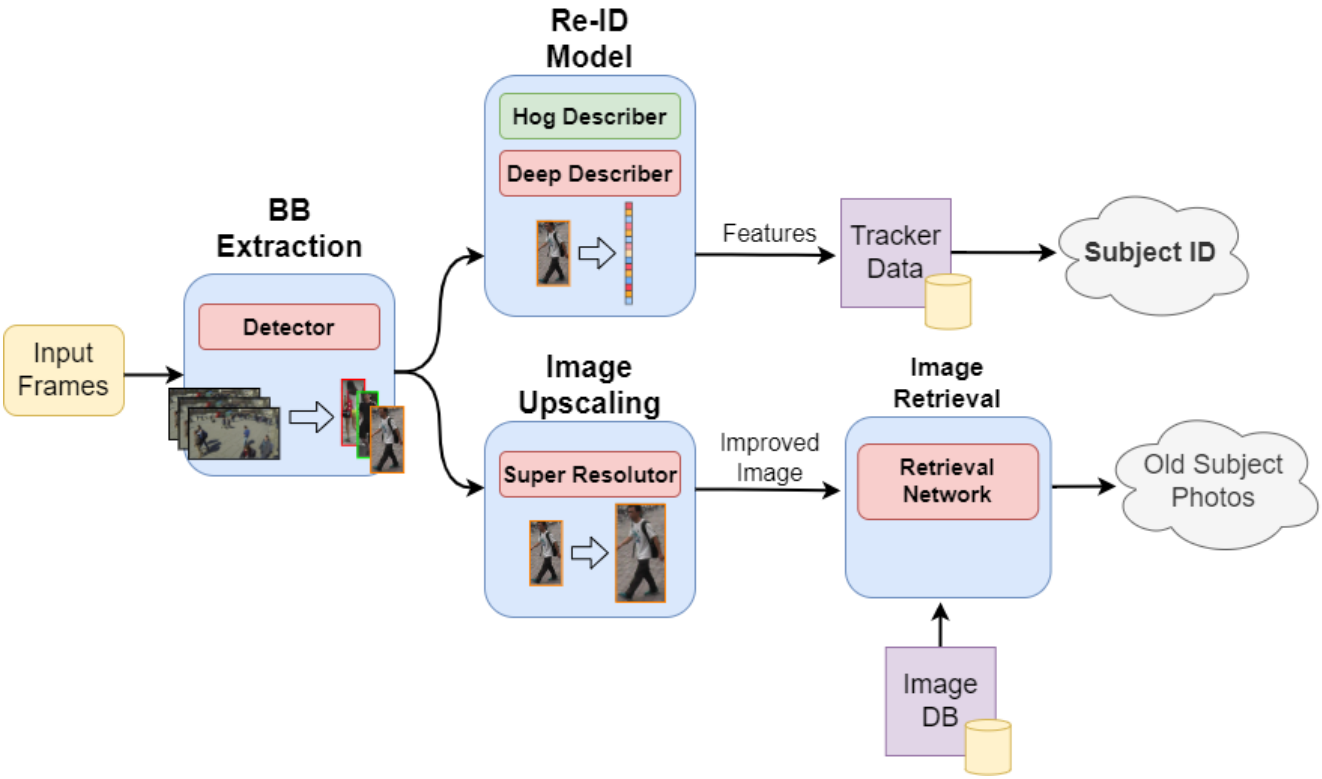

Figure 1: Framework architecture

2.1 Detection and Re-ID

A pedestrian follower can be decomposed in two major components, a detector and a re-identification model.

A **detector** is a unit which is able to find instances of one or more classes inside a picture. There are two distinct categories of detectors in the literature: multi-stage, in which the first stage is responsible for providing Regions of Interest (ROIs) that will be processed and classified by subsequent stages, and single-stage, which attempt to perform all these computations in one shot. In general, multi-stage models provide better results at the cost of greater use of computational resources and longer inference time.

The output of the detector, which in our case of interest will be a set of crops of pedestrians, is then elaborated by the second element of the tracker, the **Re-identification model**, or simply **ReID** unit. Person re-identification intends to associate images of the same individuals taken from different cameras or from the same camera in different occasions. This task before the deep era was approached, with poor results, using handmade descriptors and metric learning studies. In recent years, however, the advent of deep methods has led to a vast improvement in outcomes. Following the outlook given by Ye et al. [1], today a deep ReID model is composed by the following three parts:

1. **feature representation**, which aims to find the most expressive way to represent the image of a pedestrian using a feature vector;

2. **deep metric learning** focuses on designing the training objectives through different loss functions;

3. **ranking optimization**, which intends to optimize the retrieved ranking list using various algorithms.

2.2 Super resolution

Super-resolution is still an open problem and still lacks of state of the art's architectures. Main problems concern the degradation model that should be found, since a Low-Resolution image (LR) could be generated from multiple different High-Resolution (HR) images. In literature, first attempts tried to create synthetic images degrading some HR images with a fixed model. Anyway, this is a quite strong assumption that can't be made in real world scenarios, hence different approaches have been developed.

Blind super resolution methods deal with degradation models without any prior assump-

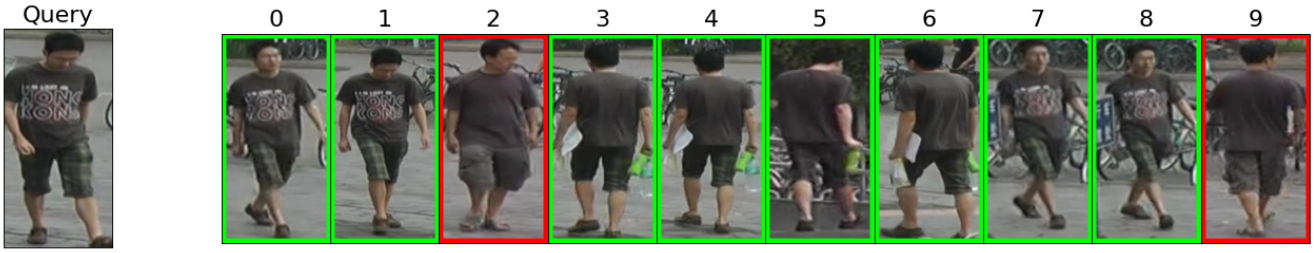

Figure 2: Example of retrieval ranked list processed by our ReID model. A green box denotes a successful retrieval, a red box a wrong one.

tion, using just their knowledge in the LR. Basically, these approaches try to understand the real degradation model of a particular image by downscaling the LR and estimating the degradation kernel, in order to increase their image upscaling knowledge. A famous example is KernelGAN [2].

Another approach is to use **Generative Adversarial Networks** (GAN) using the most possible varied set of degradation models, preferably of high-order type, like Real-ESRGAN[3].

As opposed to what might be perceived from the name, the difference between KernelGan and RealESRGAN is huge. The "GAN" in KernelGAN isn't about the generation of an image, but it refers to the **generation of a kernel which represent the degradation**, using the GAN mechanism. KernelGAN exploit only the input image, thus without the need of pretrained weights. In opposition, RealESRGAN is trained with only pure synthetic data, created with a degradation model of high order.

The degradation model used by RealESRGAN for generating LR images is composed by a pipeline of blur, resize, noise and jpeg-like compression. Obviously, each of these stages has many different possible hyperparameters. It has been tried also to apply the same pipeline multiple time for obtaining a high order degradation model, theoretically similar to what it happens in real-case scenarios. However, a final solution seems to be still quite far.

2.3 Image retrieval

The goal of an image retrieval system is to retrieve, from within a database of images, those images which are similar to the input image. This is a complex operation as there is a need to define a metric on which to evaluate whether two images are similar or not. In fact, a single image can be analysed from different perspectives, and consequently different results can be obtained by changing the way it is processed.

In our case, we assumed to own a database containing images of subjects previously identified by the surveillance system. The goal is thus to be able to create an image retrieval system that, having supplied an image of a given subject, could retrieve images belonging to the same subject from previous records of the surveillance system.

Our image retrieval system will have to deal with the problem of people re-identification. Most re-identification methods rely on the assumption that an individual caught on camera will reappear under another camera in a relatively short amount of time, typically less than 30 minutes. This short-term re-ID scenario assumes that there is a low likelihood of the person **changing their clothes**. On the contrary, for this case we assume that this time interval will longer, such as more than a day, so the chance of clothing or accessory changes increases, leading to a **long-term re-ID scenario**. This type of scenario has not yet been thoroughly addressed in the currently available literature, as it poses much more complex challenges than the classic short-term re-id problem.

One of the currently proposed solutions to this problem is ReIDCaps [4], a deep neural network trained using vector-neuron capsules instead of the traditional scalar neurons, aiming to archive better performance.

3 Proposed method

Our proposed method for completing the task from detection to retrieval is composed of three parts. The first and main one is the tracking by detection, where the detection is performed by YOLOv7 and the tracking is carried by us. For this task, we propose two different approaches, of which one exploits classical algorithms, while the other one is carried by the usage of a custom neural network. The second section is related to image processing, where crops of pedestrians are visually improved, gaining quality and resolution. This part is a composition of classical methods among with a pre-trained neural network. Finally, the third part has the goal of retrieving similar

images to the ones pointed by the user, aiming to get images belonging to same individual shot in different visual contexts, including ones with different clothes.

3.1 Detector

Although it would have been interesting to investigate how to implement a pedestrian detector from scratch, balanced for our baseline purposes, we decided to deepen the tracking task just through the development of a custom ReID model. Therefore, we opted for using a pretrained network to perform the first step of out pipeline, the pedestrian detection. After a research regarding balances between performance and computational inference costs of current state-of-the-art methods, we chose to adopt YOLOv7-tiny model [5] as detector, given its low inference costs, which are designed for real-time tracking, but still a very high throughput.

Figure 3: Visualization of HOG feature vector

3.2 Pedestrian ReID models

For our research we decided to use two different models to resolve the ReID task, of which one doesn't involve neural networks. As expected, results offered by this model, which is based on Histogram Of Gradients (HOGs) feature vectors, cannot compete with those showed by the deep one. Goal of both implementations is to provide a valid representation of pedestrians, that will then be stored and compared by a common final module, named PeopleDB. The datasets we used to train the deep model are MOTSynth[6] and Market-1501[7]. Being a synthetic dataset, MOTSynth offers broad availability of data and labels for various computer vision tasks.

Results have been checked using three different metrics: **mAP** (Mean Average Precision), **CMC** (Cumulative Matching Characteristics) and **mINP** (Mean Inverse Negative Penalty). The latter is a new metric introduced by [1], which measures the penalty to find the hardest correct match. In order to study the effect of cross-domain datasets, we calculated these metrics also on a different dataset, MARS[8].

3.2.1 HOG based model

As our first model to resolve the ReID task, we wanted to try a simple and non-deep resolution method. We therefore had to choose a feature descriptor which could perform a basic discrimination between different pedestrian images. We opted for a **Histogram of Gradients descriptor**. The intent of a feature descriptor is to summarize the object in such a way that a target belonging to the same class produces a result as close as possible to the same feature descriptor, even when viewed under different conditions. A HOG descriptor processes an image using an **edge operator**, such as the Sobel filter. The resulting feature map is then divided in blocks, and for each block gradient is quantized based on its direction. Hence, the final result is a tensor in which the last dimension is a histogram of the quantized gradient directions of each block. Figure 3 shows an example of a pedestrian image processed with HOG descriptor.

rank-1	35.8
mAP	10.8
mINP	3.9

Table 1: HOG-based model results (%) on MOTSynth

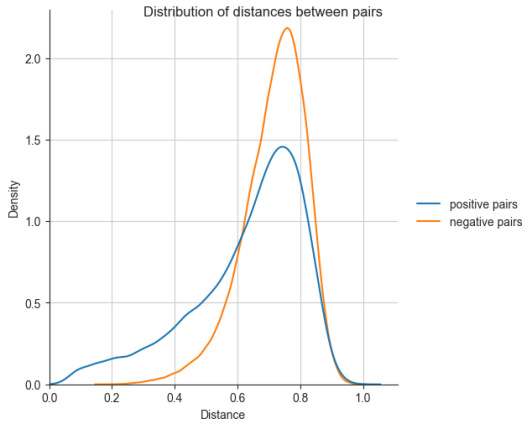

Figure 4: Discrimination ability of the HOG based model on MOTSynth

To find out the ability of HOG describer in succeeding at ReID task, we tested it with different metrics conditions. We tried both cosine and euclidean distances, and also using a RGB-sensible descriptor (which uses three times more memory respect the standard one). However, all this cases brought weak results. In table 1 it is

possible to see metrics performed on MOTSynth dataset, using cosine distance. Other combinations had similar results. Figure 4 shows how this descriptor separates the distribution of distances between pairs of images with the same identity respect the distances between different identity pairs. It is easy to note that the two curves, which ideally should be separable, are mostly overlapped, denoting a weak discrimination ability in this task.

3.2.2 Deep ReID

For our deep model, we decided to use ResNet networks as backbones. In particular, we chose ResNet50 and ResNet18 models, in order to compare results offered by networks with different classification performances. For both the backbones we removed the last fully connected layer and substitute with a new one. Following the studio conducted by Luo et al. in [9], we also added a **Batch Normalization** layer (BN). The latest FC layer has been added just for training, and has an output class number equal to the number of different identities used to train. During inference mode, the FC layer is detached, thus giving the ability to **store the final representation** of each pedestrian image. Specifically, this feature vector is sized 512 for the ResNet18 and 2048 for ResNet50.

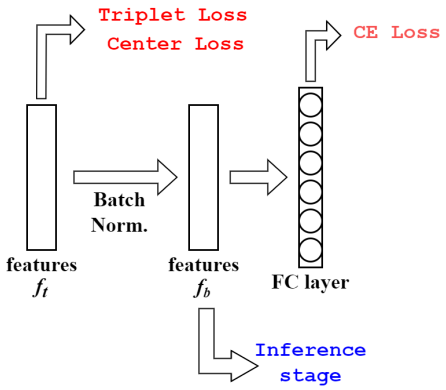

Figure 5: Last layers of the deep model. The three different losses are computed on different feature vectors

The objective function $\mathcal{L}$ used during training sessions is composed by three different parts:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \alpha\mathcal{L}_{\text{Tr}} + \beta\mathcal{L}_{\text{C}}$$

where α and β are hyperparameters. Here $\mathcal{L}_{\text{CE}}$ is the **Cross Entropy loss**, defined as

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{N} q_i log\left(\frac{exp(s_i)}{\sum_{j=1}^{N} exp(s_j)}\right) \begin{cases} q_i = 0, & y \neq i \\ q_i = 1, & y = i \end{cases}$$

where y is the truth ID label and s_x is the score given to the class x. $\mathcal{L}_{\text{Tr}}$ is the **Triplet loss** (also called Hinge loss), which is

$$\mathcal{L}_{\text{Tr}} = max(0,\ d_p - d_n + \gamma)$$

where d_p and d_n are distances of positive and negative pairs, and γ is a margin factor. Triplet loss aims in **shrinking distances** between couples of features referring to the same identity (positive pairs) respect features of different identities (negative pairs). Although the basic version of the triplet loss uses random pairs of feature vectors to compute distances d_p and d_n, for better results we chose to use the **hardest feature pairs**, i.e. the furthest positive match and the closer negative match, respectively.

Triplet loss only considers the difference between d_p and d_n, ignoring their absolute values, hence not caring about the intra-class compactness. For this reason, we decided to include also the **Center Loss** $\mathcal{L}_{\text{C}}$:

$$\mathcal{L}_{\text{C}} = \sum_{j=1}^{N} \left\| f_{t_j} - c_{y_j} \right\|_2^2$$

where y_j is the label of the j-th image in a mini-batch. c_{y_j} denotes the j-th class center of deep features. This loss intends to **squeeze the feature cluster size** of each identity. It should be payed attention to the fact that triplet loss and center loss are computed on features before the batch normalization layer, as showed in figure 5, following the research conducted in [9].

Regarding the hyperparameters α and β, we tried a few combinations for $\alpha \in \{1, 10\}$ and $\beta \in \{0.0025, 0.025\}$, and results showed best results with $\alpha = 10$ and $\beta = 0.025$. The margin factor γ has been fixed to 0.3.

Training has been performed using two different datasets:

- The first one contains pedestrian crops extracted from the synthetic dataset MOTSynth[6], and will be denoted with "**Mot**". Training set is based on 126866 images, subdivided in 1000 different identities. Test set uses 750 identities, with 4 probes and 20 gallery pictures for each of them;

- the second dataset is Market1501[7], later indicated with "**Mar**". Its train set is formed by 12936 crops belonging to 751

	Name:	A	B	C	D	E	F	G	H	I
	Model:	ResNet18				ResNet50				
	Training:	50e Mot	100e Mot	50e Mar	100e Mar	50e Mot	100e Mot	50e Mar	100e Mar	50e Mot + 50e Mar
MOTSynth	rank-1	92.3	92.8	82.2	83.3	92.6	**93.2**	79.0	76.7	83.6
	mAP	66.8	68.0	45.8	48.0	68.8	**69.2**	42.6	40.8	48.9
	mINP	20.6	21.8	9.0	9.2	22.6	**23.0**	8.0	7.5	9.9
Market 1501	rank-1	94.1	94.7	96.9	**97.6**	92.8	91.9	94.7	95.0	94.2
	mAP	19.5	20.2	44.9	**50.7**	19.5	18.3	37.6	36.7	40.3
	mINP	0.6	0.7	5.5	**8.2**	0.6	0.5	4.0	3.6	4.1
MARS	rank-1	82.9	83.2	89.8	**91.4**	82.3	81.9	86.9	86.5	86.9
	mAP	28.4	29.2	49.7	**54.5**	27.9	26.8	43.4	42.2	43.7
	mINP	1.4	1.4	6.8	**9.0**	1.4	1.3	4.8	4.2	4.2

Table 2: Results (%) on models trained differently

pedestrians, while test set contains 19732 images regarding 750 identities.

Table 2 displays resulting metrics of 9 different trainings. First ones (A - D) have been performed using ResNet18 as backbone, showing promising results. Here both trainings performed on Mot and Mar showed improvements on all test datasets when prolonged from 50 to 100 epochs, denoting a good generalization. All other trainings (E - I) used ResNet50 as backbone, which is deeper and more powerful respect the first one. However, **this greater potential didn't result in better results**: on the contrary, we saw a general degradation of performances, which can be explained by two factors:

- trainings E and F, performed on MOTSynth dataset, showed better performance on the relative test set in comparison to A and B. Nevertheless, outcomes on the other test sets showed worse values, meaning that we are loosing in generalization ability. Network is thus overfitting over the dataset;
- G and H, trained on Market1501, displayed worse performances even in metrics about the relative test set (compared with C and D). Tests performed while training showed an improving trend, therefore excluding a possible overfitting over the training set. This phenomena can be explained with the training set being too small for ResNet50; training performed on Mot didn't show this problem, since that training set is much bigger.

Finally, training I has been performed as experiment, trying to combine the high quantitative of training data offered by Mot with the less but more realistic data contained in Market1501.

We choose to keep training D as our model, having better performances in real-world scenarios.

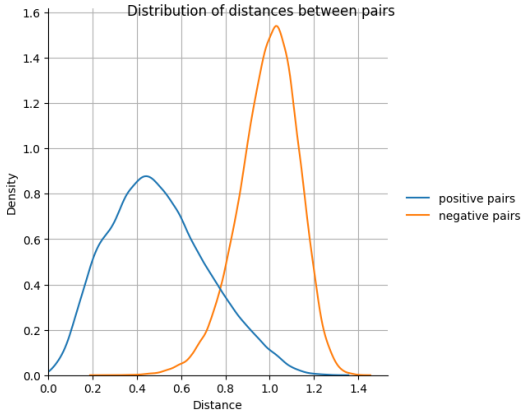

Figure 6: Discrimination ability of our deep model on MARS

3.2.3 PeopleDB

The last section of our tracker model has been called "PeopleDB" and, like a database, is used to store the feature representation of each pedestrian processed by the whole network. This component is also responsible to choose whenever a given feature vector represents a new subject, rather than one already seen. The workflow is quite simple and can be explained as follows:

1. a query feature vector q, representing a pedestrian, is given in input;
2. for each identity $j \in \{0, 1, ..N\}$ currently stored in the database, for which can exist multiple samples, the cluster center c_j is calculated;

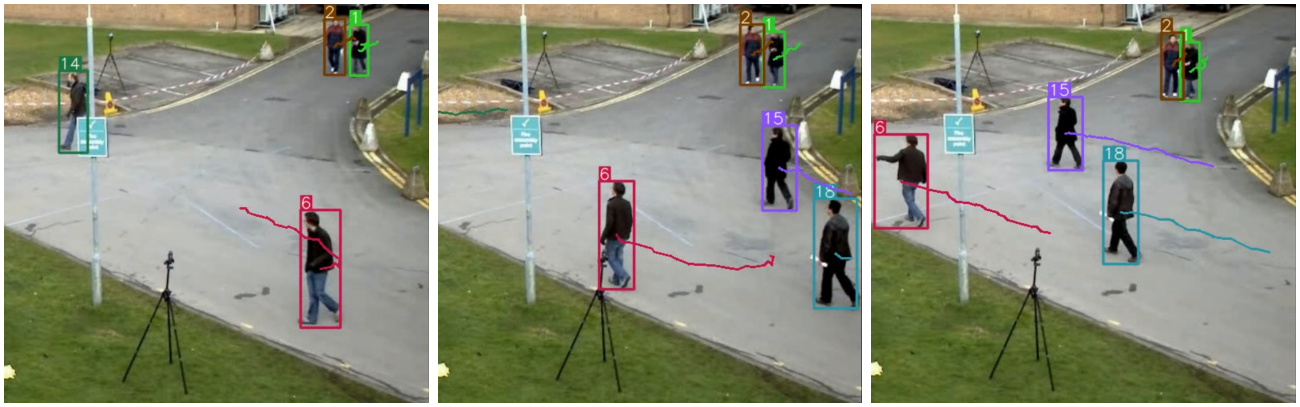

Figure 7: Evidence of correct tracking performed by our method

3. it is processed the distance $d(p, c_j)$ respect the probe and every center c_j. Results are then sorted in increasing order;

4. taken the first result c_x, $x \in \{0, 1, ..N\}$, which is the cluster center closer to the query q, it's compared to a threshold th:

   - if $c_x <= th$, then we recognize this query as belonging to the identity x. The probe q is thus added to the database samples labelled with identity x, and the value x is returned.
   - otherwise, if $c_x > th$, this feature vector belongs to a new identity. Hence, the database will create a new identity z which will be then returned as result, and q will be saved as sample of identity z.

Analyzing resulting metrics, we concluded that the distance function $d(p, c_j)$ which gave us better results is the **cosine distance**. Euclidean distance performed a little worse.

The choice of the **right threshold value** th is indeed an important success factor: a too high value brings to a **loose identification**, in which several pedestrians are classified as the same person. On the opposite, if th is too little, it will be easy that the same individual in different frames will be classified with different identities, resulting therefore in a **too strict classification**. Nevertheless, being the target application a ReID task applied to a video stream, we can take advantage of the fact that there won't be too much difference in feature vectors belonging to the same pedestrian in contiguous frames, letting us to choose a threshold value small enough to not give the same identity to different individuals. This temporal hypothesis has been shown to be quite truthful, letting the whole component to act using -very simple- **notions of temporal information**. A more complex system could be designed indeed, taking advantage in a more complex way of the flow of time. Figure 7 shows an example of a correct tracking evolution.

Finally, as some practical uses, we limited the maximum number of samples which can be saved for each identity, after which every new sample will replace a random pre-existing descriptor. Furthermore, if a stored identity is not updated with new data for a long time, its data are then deleted, being the individual not present anymore in the environment.

3.3 Super resolutioner

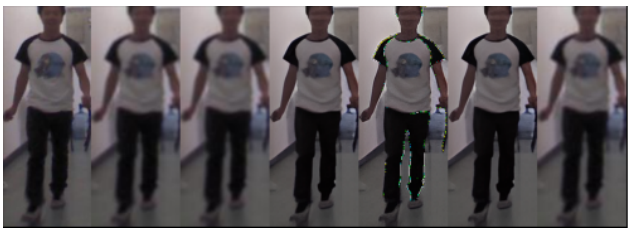

Figure 8: Same image, downsampled then reconstructed with different methods

At this point of the pipeline, the user could have necessity to analyze a tracked pedestrian, i.e. to recognize him or to extract other information. Since it is not difficult for detected people to be far from the camera, this analysis -performed by a human or by a computer- could be compromised. To avoid this eventuality, we added a step where image can be visually improved. This should be useful also in case of an eventual image retrieval process.

As first approach, we tried to exploit classical methods, such as *Bicubic interpolation* and opencv's method *pyrup*, that upsamples the image and then blurs it, used for gaussian pyramids. Be-

		Downsampled			Bicubic			Superres		
		mAP	rank1	rank5	mAP	rank1	rank5	mAP	rank1	rank5
Our ReID Model	Euc	10.9	59.0	77.3	10.9	58.0	78.0	10.6	55.8	77.1
	Cos	**12.4**	**61.9**	**80.1**	**11.9**	**59.0**	**78.5**	**11.0**	54.9	76.5
ReIDCaps	Euc	8.8	54.2	71.8	8.8	50.9	71.4	10.1	62.7	83.8
	Cos	8.8	54.2	71.8	8.8	50.9	71.4	10.1	**62.7**	**83.8**

Table 3: Results (%) of the Image Retrieval system using different methods for the upscaling and different re-ID models

ing not satisfied by obtained results, we decided to investigate the problem using deffierent methods: **Real-ESRGAN** [3] and **PDM-SR** [10]. The latter is a blind-SR method that treats degradation as a random variable.

For performance measurement we used the **PSNR** -Peak Signal to Noise Ratio- in order to obtain a score about the quality of reconstruction, computed on dataset "Set5", which is a benchmark dataset for super resolution. From here we took some images and performed a degradation, obtaining LR variants which acted as input for the models. Degradation has been performed using the same method used by RealESRGAN, so with a mixture of blur, resize, noise and jpeg-like compression. Resulting performance are shown in Table 4. We choose as our model Real-ESRGAN, having better results respect other models. Real-ESRGAN is an architecture of the GAN's family whose generator was taken from SRGAN[11] , that use Residual-in-Residual block (RRDB) and a second part, UNet, which is in charge of dealing a larger degradation space, implemented in VGG-style with skip connections.

Method	PSNR
PDM-SR	17,1
Real-ESRGAN	**20,7**
Bicubic	11,2
PyrUp	11,7

Table 4: Resulting metrics about different models

To facilitate the subsequent process of image retrieval, we decided to preprocess those images which will be evaluated using a mixture of gaussian and sharpening filters, followed by the RealESRGAN model. The usage of this preprocess method resulted in an overall improvement of image retrieval performance, meaning that the goal of this chapter has been reached. Table 3 displays image retrieval performances using low-resolution images, or HR computed both with a simple bicubic interpolation and our more complex method. Our image retrieval model, which is ReIDCaps, shows better performance when using out super resolution method. Our method can be useful also for enhancing the quality of the image for human purposes.

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

Figure 9: Example of super resolution model comparing. (a) is the GT image, while (b) is the computed low resolution image. (c) and (d) show how PDM-SR and RealESRGAN, respectfully, performed over the LR image.

3.4 Image retrieval

An image retrieval system can be created by defining two components: a **feature extraction system**, which compresses images into feature vectors, and a **similarity measure** that will be used to compare images feature vectors resulting in a measure about which of them are relevant respect to the given one.

To test obtained results we chose to exploit the Long-Term Cloth-Changing Dataset [12], which is a recently proposed dataset that fits perfectly with out needs. This dataset contains 17k images belonging to 152 different identities.

With regard to the feature extraction component, we first tried to use the ReID model described in the chapter 3.2.2, being a similar task. Obtained results are visible in Table 3. These metrics don't show bad, but surely can be improved; being the model trained for a different task, it is likely that is focuses too much on pedestrian clothes respect other morphological information. For this reason, we decided to integrate the ReIDCaps network [4]. This recent model

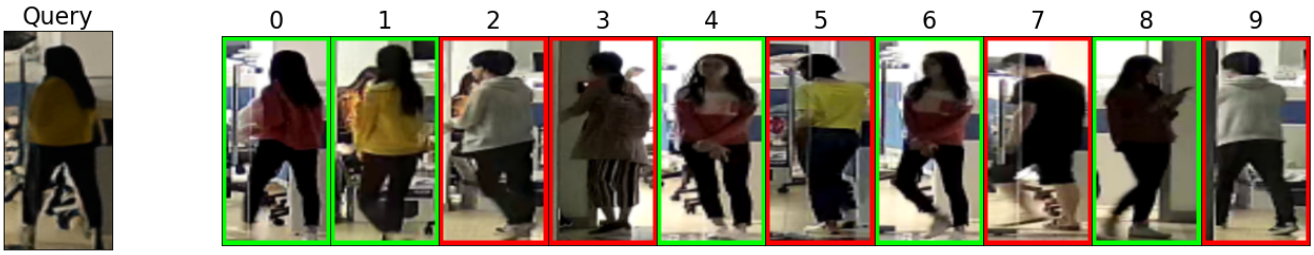

Figure 10: Example of retrieval ranked list processed by our image retrieval model. A green box denotes a successful retrieval, a red box a wrong one.

proposed an novel architecture, based on **neural capsules**, instead of the traditional scalar neurons.

ReIDCaps is divided into three modules: the feature extraction module, the ID and dressing perception module and the auxiliary modules. Feature extraction component is the main component, as it provides the corresponding feature vector for each image within the image retrieval system. In this implementation, the DenseNet-121 architecture processes an image of size 224x224, extracting a feature vector of size 1024x7x7. This vector will be then reduced into a single vector of 1024 components.

In order to understand the novelties proposed by this model, we will briefly describe how it was trained by its creator, by analyzing the second component: the ID and Dressing perception module, along with the Auxiliary Modules, which is responsible for the correct training of the module. This module is the actual innovation of the network as the technique of **neural capsules** is exploited. Neural capsules training technique was firstly proposed [13] by Sabour, Frosst, and Hinton in 2017 and it was initially applied to classify the MNIST dataset. The module takes as input the feature vector produced by the previous module and through reshape operations will create a first layer of capsules of 8 components each. The number of capsules in the next layer will be equal to the number of people present in the train set. The **Routing by Agreement** will be then applied where it will be asked to each capsule of the first layer to predict the value of each capsule of the next layer, using different transformation matrices that will be learnt from the network. In this step, we can image each capsule of the first layer as a piece of the total image, and starting from that we ask to try to predict the final image, that will be represented by the second layer capsules. The final value of a capsule in the second layer will be computed using a weighted sum of the different predictions, giving more relevance to those with a similar value. In order to train the model, the magnitude of each capsule is then computed and then the margin loss is used to obtain the highest value in the capsule corresponding to the subject in the given input image.

Once our system has generated the feature vector for each image in the gallery, stored in an external database, we compute the feature vector for the query image. A similarity measurement is then applied between the probe descriptor and gallery descriptors. For this step, we tried both cosine and euclidean distance. As seen in the Table 3, for our ReID model the cosine distance showed better results, while regarding the ReIDCaps model, it returned quite equal results compared the the Euclidean distance. We then decided to use Cosine one. This similarity score is thus used to rank all the possible images in the dataset, and the 10 most similar ones will be returned to the user, so that he will be able to perform a last check and to choose subjects which looks more similar.

4 Conclusion

In conclusion, our computer vision project focused on video surveillance has shown promising results.

Our deep learning-based re-identification model outperformed the classical HOG model, demonstrating the effectiveness of using deep learning in this field. We also demonstrated that ReID task can be achieved by means of synthetic datasets, such us MOTSynth. In our case, at the end we decided to keep as our model training performed with Market1501, but with the help of regularization and re-ranking techniques, models trained with synthetic data can surely be utilized to pedestrian re-identification. Our super-resolution component successfully improved the quality of identified subjects' images, leading to better results in the image retrieval task. Finally, our third component, which exploited the capsule neural networks, displayed good performance in identifying individuals within a dataset of surveillance images, even when they are wearing different clothes.

Overall, our project showcases the potential of computer vision in enhancing video surveillance systems and provides a foundation for further research in this area.

4.1 Future work

Our deep ReID model can surely be improved by adding **re-ranking techniques**, which are algorithms that improve performances of retrieval problems by performing an analysis of the generated ranked list.

Another valid idea is substitute our simple PeopleDB component with a more complex component. In fact, in the framework explained in this paper we don't take into account temporal flow information, which can be a very useful data in re-identification task. A component base on a **recurrent neural network** (RNN) could be a much more effective solution for real-life scenarios.

For the Super Resolution part, it could be interesting to create a dataset composed on only paired couple of pedestrian images (HR e LR) and to fine-tune the architecture that we used. Since the challenge in this task is to recreate the complexity of a real degradation (that cannot be reduced to a series of compression and adding noise), a deeper study in understanding better this complicated argument could lead to important improvement.

Also the image retrieval system can be perfected though the usage of new techniques, in order to get better results in the problem of long-term re-identification. Possible solutions could be new architectures aiming to split person morphological data, such as the face or the body shape, from the clothes information, that for this task are irrelevant. Some other ideas involve usage of keypoints, in order to better understand the pose of the subject, or even better the usage of some attention mechanism.

References

[1] M. Ye et al. "Deep Learning for Person Re-Identification: A Survey and Outlook". In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 44.06 (2022), pp. 2872–2893. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3054775.

[2] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. "Blind Super-Resolution Kernel Estimation using an Internal-GAN". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/5fd0b37cd7dbbb00f97ba6ce92bf5add-Paper.pdf.

[3] Xintao Wang et al. "Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data". In: *International Conference on Computer Vision Workshops (ICCVW)*. 2021.

[4] Yan Huang et al. "Beyond Scalar Neuron: Adopting Vector-Neuron Capsules for Long-Term Person Re-Identification". In: *Transactions on Circuits and Systems for Video Technology (TCSVT)* (2019).

[5] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors". In: *arXiv preprint arXiv:2207.02696* (2022).

[6] Matteo Fabbri et al. "MOTSynth: How Can Synthetic Data Help Pedestrian Detection and Tracking?" In: *International Conference on Computer Vision (ICCV)*. 2021.

[7] Liang Zheng et al. "Scalable Person Re-identification: A Benchmark". In: *Computer Vision, IEEE International Conference on*. 2015.

[8] *MARS: A Video Benchmark for Large-Scale Person Re-identification*. Springer. 2016.

[9] Hao Luo et al. "A Strong Baseline and Batch Normalization Neck for Deep Person Re-Identification". In: *IEEE Transactions on Multimedia* 22.10 (2020), pp. 2597–2609. DOI: 10.1109/TMM.2019.2958756.

[10] Zhengxiong Luo et al. "Learning the Degradation Distribution for Blind Image Super-Resolution". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 6063–6072.

[11] Xintao Wang et al. "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. Sept. 2018.

[12] Xuelin Qian et al. "Long-Term Cloth-Changing Person Re-identification". In: *arXiv preprint arXiv:2005.12633* (2020).

[13] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. *Dynamic Routing Between Capsules*. 2017. DOI: 10.48550/ARXIV.1710.09829. URL: https://arxiv.org/abs/1710.09829.