LARGE MARGIN LOSS FOR IMAGE RETRIEVAL

Andrei RACOVIȚEANU[1], Corneliu FLOREA[2], Mihai BADEA[3]

Over the years the research community has searched for the most efficient descriptors for the task of image retrieval. Along with the development of hardware resources that led to the widespread use of convolutional networks, new alternatives emerged. Although the results were promising, there are still many possibilities to improve the efficiency of the descriptors provided by the convolutional networks. This paper proposes an efficient method based on CNN features extracted using a large margin loss to provide a better separability in the descriptor space for an image retrieval task. The method was tested in two scenarios on a complex database that contains images of different places. The first scenario was testing the large margin loss on the entire database, while the second involved more particular situations where the potential of this type of loss is better isolated. In both scenarios, promising results were obtained regarding most of the retrieval metrics.

Keywords: Large Margin Loss, Convolutional Neural Networks, Image Retrieval

1. Introduction

Convolutional networks have emerged as one of the most popular methods in the fields of Image Processing and Computer Vision as a result of the development of corresponding hardware equipment. Besides the classical problems of classification, segmentation or detection, CNNs (Convolutional Neural Networks) can be used in image retrieval tasks, especially in finding similar images. The applicability of image retrieval is vast and includes recommendation systems or finding relevant images in fields such as medicine or photography. With the development of Internet browsers and the necessity to access useful information, the first image retrieval algorithms appeared. These approaches are still used by the majority of search engines to locate relevant photos based on certain keywords. However, in these first stages, the visual information included in the images was not taken into account. Later, research started to focus on methods of extracting the relevant visual features according to the tasks that had to be addressed [1].

One of the most well-known visual descriptors in Computer Vision problems is HOG (Histogram of oriented gradients) [1] and LBP (Local Binary

[1] Ph.D. student, Applied Electronics and Information Technology Department, University POLITEHNICA of Bucharest, Romania, e-mail: andrei.racoviteanu@upb.ro

[2] Professor, Applied Electronics and Information Technology Department, University POLITEHNICA of Bucharest, Romania, e-mail: corneliu.florea@upb.ro

[3] Ph.D. student, Applied Electronics and Information Technology Department, University POLITEHNICA of Bucharest, Romania, e-mail: mihai_sorin.badea@upb.ro

Pattern) [3]. HOG was effective at filtering image details (contours, corners), while LBP excelled in recognizing textures. HOG [2] is derived from the previous SIFT (Scale Invariant Feature Transform) construction [4], which complements the descriptor with a keypoint locator that is not sensitive to rotation and scale changes. SIFT has proven to be very effective in terms of matching or finding objects in images, yet, Bay et al. [5] found room for improvement by means of using integral image to propose SURF (Speeded Up Robust Features). Binary Robust Independent Elementary Features (BRIEF) [6] was proposed for choosing key areas in the image through a simple binary comparison between the pixel intensity values in a image patch.

With the appearance of convolutional networks, the descriptors provided by them were used more and more often. Due to the increased ability of CNNs to find relevant features, pre-trained models on completely different data than those that had to be retrieved were also tried. Zhou et al. [7] used a pre-trained network on a different task for extracting effective embeddings for image retrieval. In addition, they proposed a novel small CNN which provides more relevant descriptors and reduces dimensionality at the same time. In [8] a method is presented that combines several local features from convolutional layers with the help of VLAD [9] encoding to obtain a unique descriptor vector per image. It was showed that intermediate layers can be, under certain conditions, more effective than the last fully-connected (FC) layer.

The paper is organized as follows: in the next section we emphasize the main steps of the proposed method, including large margin loss functionality and the framework used; Section 3 is dedicated to experimental results; Section 4 outlines the main conclusions of the paper.

2. Method

This article's main topic is a large margin loss, inspired by Center [10] and Island Loss [11]. For the best final classification using a convolutional network, the FC features should form a sparse descriptive space. This makes adjusting the separation boundary easier without confusing network decisions. Unfortunately, perfect data separation is not always possible, so methods to improve it were sought. Wen et al. [10] introduced Center Loss to reduce embedding density. Reducing intra-class compactness improves delimitation. The algorithm assigns each class a center and reduces the distance between samples and their own center. Unlike Center Loss, Island Loss forces a greater distance between descriptive space clusters. Thus, overlapping classes are more separated. Fisher loss [12] was defined as a combination of the 2 previously mentioned losses. In this case, the distance between the centroids, and not the angle measurement (island loss), is taken into account. More recently, Sun et al. [13] improved the triplet loss [14] by weighting the positive and negative examples differently in relation to the distance of the anchor(reference).

2.1. Large Margin Loss

Center Loss [10] can be useful but does not affect other classes. Even with more compact data, there may be overlaps. The proposed loss should create a large gap between embeddings clusters which could lead to improved results.

The large margin loss can be defined as:

$$L_{LM} = \sum_{i=1}^{N} \left(\left\| \frac{x_i}{\|x_i\|_2} - \frac{c_j}{\|c_j\|_2} \right\| - \frac{1}{C-1} \sum_{k=1;k\neq j}^{C} \left\| \frac{x_i}{\|x_i\|_2} - \frac{c_k}{\|c_k\|_2} \right\|^2 \right) \qquad (1)$$

where x_i is the embedding belonging to class j and c_j is the centroid associated to this class. Basically, left term describes the Center Loss effect of gathering the data samples around the centroids. N represents the batch size, C is the number of classes and implicitly the number of centers and c_k is a different centroid compared to the belonging prototype to each sample c_j . The second term considers the Euclidian distance between the current sample and other cluster centers, imposing large margin effect between embeddings.

If we denote the normalized vector $\hat{x}_i = \frac{x_i}{||x_i||}$, the loss can be rewritten as:

$$L_{LM} = \sum_{i=1}^{N} \left(\|\hat{x}_i - \hat{c}_j\| - \frac{1}{C-1} \sum_{k=1;k\neq j}^{C} \|\hat{x}_i - \hat{c}_k\|_2 \right) \qquad (2)$$

Large margins enforce cluster sparseness compared to Center Loss. Normalizing the data and using the Euclidian distance for the second term differs for Island Loss. Normalizing the data improves numerical stability during training, while using a Euclidian distance emphasizes magnitude. In large margin loss, the focus is on embedding location relative to other classes.

The overall system is trained with a combined loss defined in eq. (3), where L_{CE} is the classical cross entropy classification loss and L_{LM} denotes the large margin loss. The parameter β is a weighting constant to balance the influence of the two terms in the total loss. In the experiments β was set to 0.01.

$$L_{TOT} = L_{CE} + \beta L_{LM} \qquad (3)$$

Fig. 1 depicts large margin loss behavior more intuitively. The figure depicts network descriptors provided by the network. Red and green dots represent the class centers. Given 2 new images and their embeddings *X1* and *X2*, the principle of the loss should minimize the distance from the data sample to its own centroid (red and green double continuous arrow) and enlarge the distance from the data sample to the other class centers (red and green double dash arrow).

Thus, the two pictures should approach the centers of the classes to which they belong, while the clusters move apart simultaneously.

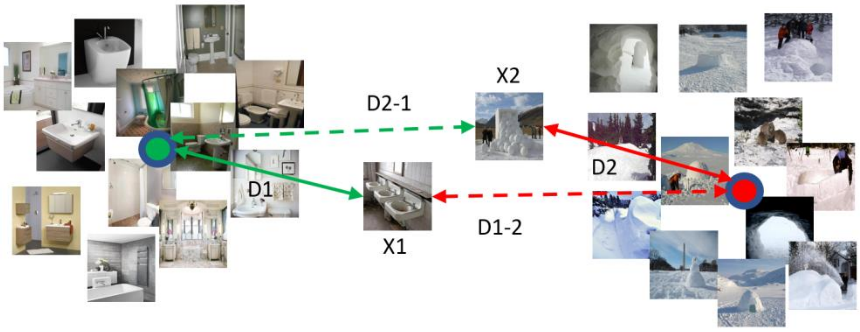

Fig.1. Behavior of Large Margin Loss

2.2. Framework and Dataset

Dataset. Due to the diversity of classes, image retrieval tasks increasingly use datasets with images of places or scenes. SUN [15] contains 130,519 images in 397 categories for scene recognition. The number of training samples is insufficient for deep learning. Inspired by the SUN work [15], Zhou et al. proposed a new more consistent dataset called Places [16] to cover the data needs of convolutional networks. Images were acquired using search engines, while Amazon Mechanical Turk was used to label the pictures. Each annotator was given 750 pictures from various categories and had to determine which belonged to the actual tag. They also checked 60 annotated examples by other persons [16].

Automatic classifiers were used to label images to increase the number of samples. The final dataset contains over 10 million images from 434 place categories and was divided into several subsets. Places365-Standard contains 1.8 million samples and 365 classes. Each class has 3,068 to 5,000 training images. The validation set has 100 images per class, while the test split has 900. Only images from the validation set were used [16]. Fig.2 shows Places365-Standard images.

Fig.2. Images from Places365-Standard

Framework. The ResNet-18 architecture trained using the ADAM optimizer was used for experiments. Fig. 3 shows the framework. The penultimate fully connected layer (orange in Fig. 3) before the output provided image retrieval embeddings. Every image was represented by 512 features. The output layer (yellow in Fig. 3) had 365 neurons, one for each class. On the 512-neuron fully-connected layer, the large margin principle was used to improve embedding separation for retrieval. The last output layer's cross-entropy loss was applied using the Softmax activation function.

Two scenarios tested the large margin method. First, a ResNet-18 network was fine-tuned on Places 365 and all convolutional layers were frozen (red arrows in Fig. 3) and only the connections with the descriptive layer (512 FC layer) were kept. Freezing convolutional layers may prevent the large margin loss from changing weights, leading to a better descriptive space representation. For the second part, the entire network was retrained. The second scenario consisted of testing the large margin loss in a different situation: a problem with non-separable data where classes are easily confused. The choice of classes to simulate these situations was a problem. The easiest solution was to use a pre-trained network on the Places dataset to check if subjectively chosen classes met the imposed conditions. These classes were extracted from the total 365 places from the dataset and were organized as followed: roof garden, tree farm, vegetable garden, yard and zen garden.

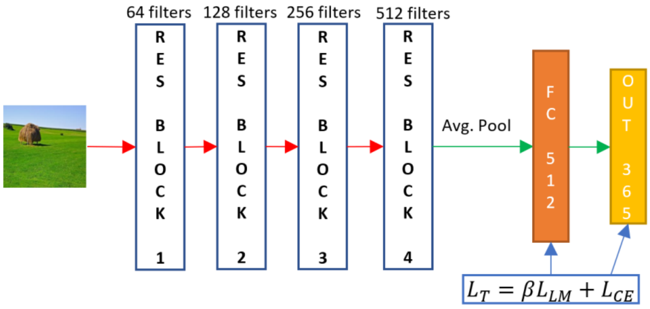

Fig.3. Framework of the method proposed

Fig. 4 displays the embedding space for the proposed situation to demonstrate the efficient choice of labels for the scenario. In addition, a figure with separable classes was created to highlight the differences. The points represented in the figure are the embeddings provided by the orange FC layer (Fig.3.) with the pre-trained ResNet-18 architecture on Places365. To address the major challenge of representing the descriptive vectors with 512 features in a human-perceivable map (2-3D), we used the t-SNE model [17] which is a dimensionality reduction technique mostly used for visualizing data in 2D and 3D spaces. Although the figures below were built with a pre-trained model on Places365, the embeddings for the scenario with 5 difficult classes are completely tangled as if the network was not really trained.

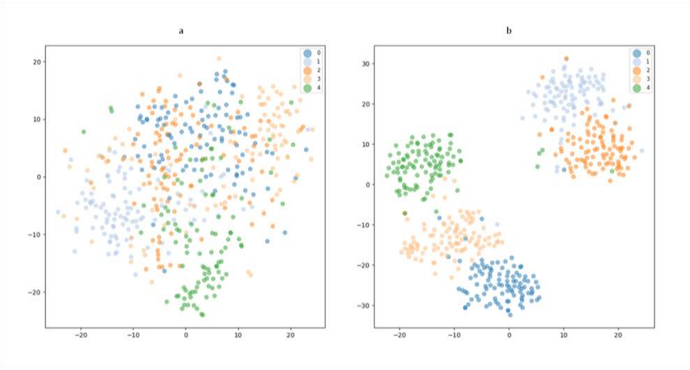

Fig.4. t-SNE representation of the 2 different situations. a – Non separable data scenario; b-Separable data scenario

3. Experimental results

As previously mentioned, 2 scenarios were tested. First, a ResNet-18 was trained on the Places365 training set. In the first phase, the pre-trained network was used to determine retrieval performance. mAP, top 1 and 5 error rate were used as performance metrics. mAP is based on average precision, another common retrieval metric (AP). The AP for an independent class can be defined as in the left-hand part of eq. (4). There, N_{TP} is the number of true positive images or the returned images with class of the query images. N_Q represents the number of retrieved images and N_{IC} is the number of same-class test images. In this manner, it is computed an average precision per every class. Finally, to determine mAP, all APs are divided by the number of classes (left hand part of eq. 4).

$$AP = \frac{1}{N_{IC}}\sum_{i=1}^{N_{IC}}\frac{(N_{TP})_i}{N_Q}\, X\ 100\%; \qquad mAP = \frac{1}{C}\sum_{k=1}^{C} AP_k \qquad (4)$$

The other performance metric is the error rate of the system, not its ability to return relevant images. Top 1 error rate is only measured for the system's first image. It measures the percentage of first retrieved images that are not the test image's class. Top 5 error rate measures the same thing, but for the first five returned photos. The Top 1 error rate formula, where NT is the number of test images:

$$Top\_1_{error} = \frac{N_T - \sum_{i=1}^{N_T}(N_{TP})_i}{N_T} \qquad (5)$$

Table 1 shows the first scenario results. The first row shows the results with the pre-trained ResNet-18 on *Places365*. The second row presents the values achieved with the fine-tuned pre-trained model only on the FC layers weights (convolutional part frozen), while the third row brings into consideration the results when a new model is trained from scratch. The network was trained for 90 epochs with the ADAM optimizer and a starting learning rate of 10^{-4} which was decreased at every 30 epochs with a factor of 10.

The performance metrics were computed for a considerable part of the *Places365* validation set. The validation set contains 36500 images from 365 classes. 29200 images were randomly chosen for the references database (80/class) and 2920 for the query dataset (8/class). mAP was computed for 5 and 8 retrieved images, but error rates were only used for 5. In addition to retrieval statistics, the table also includes accuracy. The accuracy was determined using the entire validation set.

Discussing the results, in the first 2 cases, retrieval metrics are similar. Freezing convolutional layers limits the contribution of the large margin loss to a more efficient descriptive space structuring. However, the accuracy is about 1.5% better and comparable with [16], despite using a ResNet-152 architecture instead of a ResNet-18. It must be highlighted that a better accuracy does not necessarily bring the optimal performance on the retrieval side because the retrieval task performance is usually measured on more images compared to a classification problem. Instead, the model trained from scratch increased mAP by 1.5%, despite being less accurate.

The area under the Precision-Recall curve was also calculated to evaluate the large margin loss retrieval performance. The curve was created by calculating precision and recall for each retrieval (starting from 1 to the entire reference database). Even in this case, when the large margin was used, an improvement of approximately 2% was achieved. Even though the fine-tuned model was more accurate, the AUC measure was not improved, confirming that due to frozen convolutional layers, the dominant factor in changing weights is still the cross entropy loss.

Apart from the objective performance measures, a visual analysis of the retrieval images provided by the 2 methods (CE-baseline vs LM from scratch) is also important. Fig. 5 exposes the first 5 images retrieved for CE and LM. Visually, the returned images are usually related to the query image, even if they have a different label (red bullet). CE and LM commit image class errors depending on the picture. Considering the high probability of a major overlap in a problem with 365 classes, the results are not surprising. Many similar classes in Places365 make retrieval difficult as is seen in Fig. 5. The first image in the CE case looks like a bedroom, but it is in the hotel room class which is not the only similar category with bedroom. For the ruin image the LM method gathers 5 consecutive mistakes, but the classes of wrong images are quite similar with the one requested (cemetery, temple-asia, castle or amphitheater). . It was crucial to provide a metric that measures performance for any amount of retrieved pictures to prove the utility of large margin loss because for a particular number of queries, the classical CE trained network could have better results. But what matters is the retrieval capacity on the entire data set.

Table 1

Experimental results on Places365 (mAP-mean average precision; AUC – area under curve; CE- cross entropy, LM- large margin)

Scenario/Metric [%]	mAP-5-query	mAP-8-query	Top-1-err-rate	Top-5-err-rate	Accuracy	AUC – PR curve
CE-pre-trained-baseline	29.93	28.35	66.74	37.53	53.10	9.17
LM-fine-tuned	29.89	28.56	66.99	37.95	54.69	9.23
LM-from-scratch	**31.35**	**29.98**	66.18	37.79	53.51	11.18
Resnet-152 [16]	**-**	**-**	-	-	54.74	-

Even with powerful tools like t-SNE, it is difficult to visually represent a problem with so many categories. Having so many classes may cause overlap. Even though the principle of large margin loss should better spread the descriptors, for difficult problems with many similar classes, it is obvious that its potential is limited. Even if a large margin effect is forced, many data samples will move away from certain classes and overlap with others. This is the reason why one more particular scenario with fewer classes was proposed. In this way, the impact of a large margin loss on situations involving a different data separation could be studied.

Having less data and classes available, the t-SNE technique was used again to visualize what happens to the embeddings during training. For this particular scenario the same ResNet-18 architecture was used, trained from scratch with the ADAM optimizer and constant learning rate of 10^{-4}, for 25 epochs. The weighting constant β for large margin loss had the same value 1/100. The training set was created by randomly selecting 1000 images from Places365 for each class. Images from Places365 validation set were used to create test data. The search (reference) database was another 1000 random images from Places365's training set, while the query data was all the validation set images. Each class was represented by 1000 examples in the training and reference set and 100 instances in the validation and query set, which were identical.

Figures 6 provide a comparison regarding the change in descriptive space between training with large margin loss case versus training with cross entropy only. Cross entropy loss gathers the data to some extent but does not increase cluster distances. In contrast, large margin loss increases data class compactness and cluster space. The impact increases with the degree of initial class overlap.

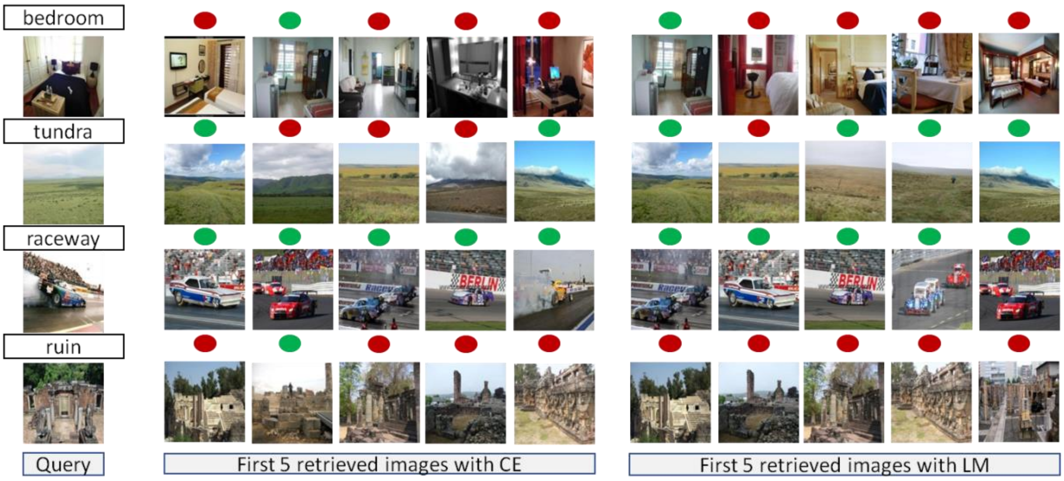
bedroom
tundra
raceway
ruin
Query
First 5 retrieved images with CE
First 5 retrieved images with LM

Fig.5. Examples of first 5 images retrieved with CE and LM. Red bullets mark retrieved images with a different class compared to query image. Green bullets describe correctly retrieved images

When problems are more difficult due to easier to confuse classes, the spacing is improved sometimes by large margin leading to results superior to cross entropy. Yet the results is not guaranteed as for instance in Fig 6 (which shows samples from the validation/query dataset), where the dark orange and dark blue classes are too similar even for large margin. The results obtained are presented in Table 2. LM outperformed CE in all metrics except Top 5 error rate for easily confused classes. Now, mAP is up 4%, accuracy is up 3%, and AUC is up over 7%. A graphic representation of the precision-recall curves for LM and CE for the case with inseparable data and initial case with the entire Places dataset can be seen in Fig. 7.

Table 2

Experimental results on (mAP-mean average precision; AUC – area under curve; CE- cross entropy, LM- large margin, Acc- Accuracy)

Scenario/Metric [%]	mAP-5-query	mAP-10-query	Top-1-err-rate	Top-5-err-rate	Acc	AUC – PR curve
CE-non-separable data	55.83	56.00	43.61	12.69	67.20	37.25
LM-non-separable data	59.75	58.97	40.60	13.45	70.08	44.75

Even if the data are overlapping, the proposed method describes the embeddings space more efficiently, though it can't reach the ideal delimitation. Figure 8 shows the difference in reference set descriptor representation (the one in which the retrieval results are sought). Large margin loss samples are more compact and farther apart. Border examples (between classes) have a lower risk of being misclassified. Large margin loss reduces the cluster of overlapping points in the left figure. Dark blue and orange classes that almost completely overlap remain difficult to separate.

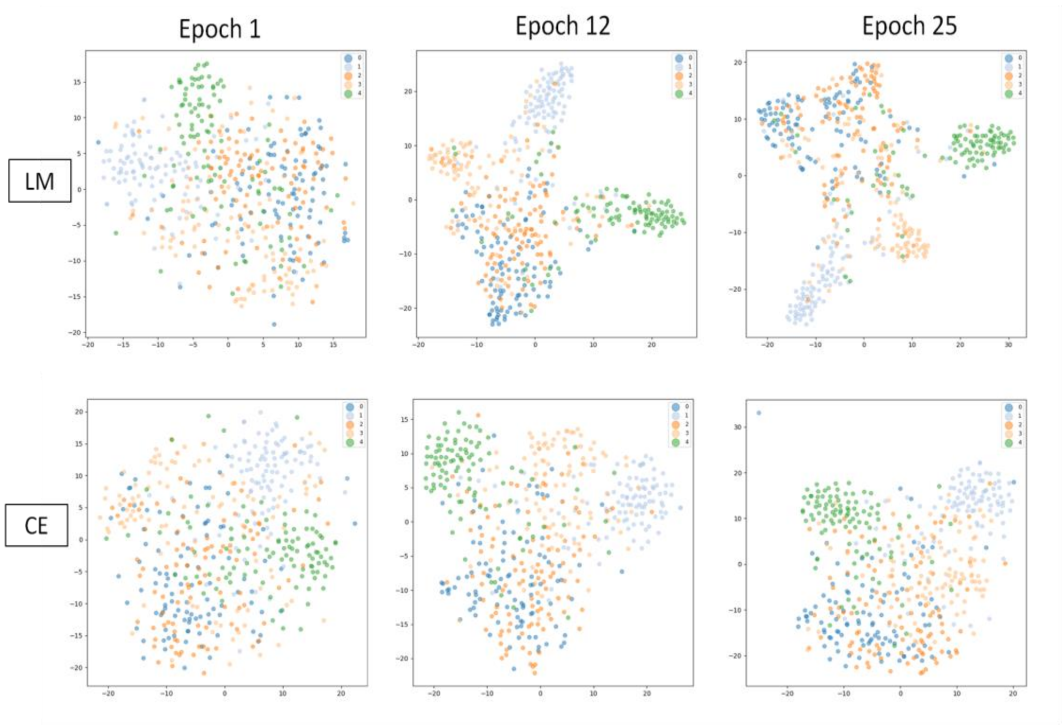

Fig.6. The modification of the descriptive space during the training process for the case with non-separable data (LM –up, CE- bottom)

The experiments were run on a Nvidia GeForce GTX 1080 video card. For the entire dataset and a batch of 128 images, the training time for an epoch with Cross Entropy only, is about 50 minutes. Adding the Large Margin loss increases the training time per epoch by 35 seconds. Referring only to a single batch, the temporal addition is approximately 2.5 ms due to the forward and backward step for the Large Margin.

The centers are trainable parameters of the network and change directly through the gradients provided by the large margin component. However, if the position of the centroids is explicitly computed for each given batch the time for an iteration can increase with 10ms or even more in comparison with Cross Entropy, which can lead to an increase of 4-5 hours in the total training time (90 epochs). It should be mentioned that the previously specified times were obtained when the network was trained from scratch.

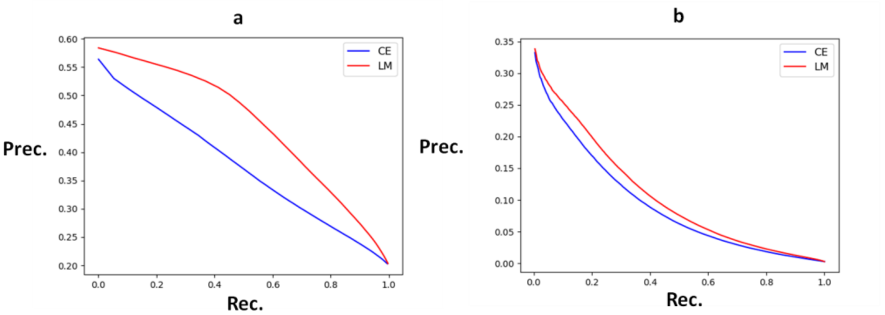

Fig.7. PR- curves for LM and CE. a- non separable data scenario; b- initial scenario with the entire Places365 database

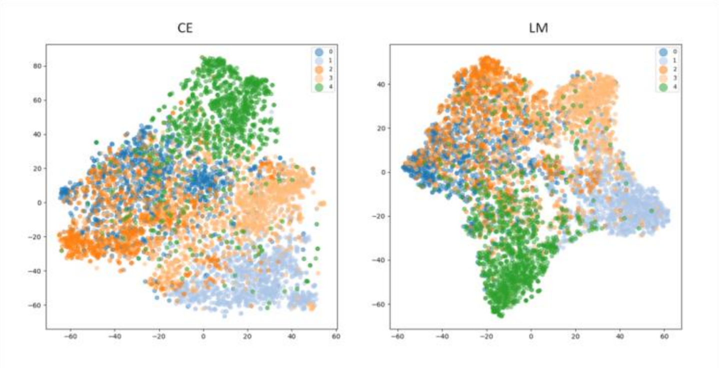

Fig. 8. The descriptors representation associated with the reference set with CE (left) and LM (right) for non-separable data scenario

4. Conclusions

The field of convolutional networks contributed to obtaining some useful descriptors for image retrieval problems. In the experiments presented throughout the paper, the use of a large margin type loss for a more efficient representation of the embeddings space was demonstrated. Introduced during training, this loss improves accuracy and retrieval.

The method was tested in several scenarios, but the best improvement was with non-separable classes. In this scenario, accuracy gained 3% and retrieval metrics such as mean average precision and area under precision-recall curve doubled. Performance gains on retrieval tasks can be much higher even if accuracy gains are weak or nonexistent. Due to the data complexity, there are still limitations. Some separable features in a first phase may move to other classes or overlap with other samples, but the large margin loss is more efficient overall. Considering these factors, the proposed method is a good retrieval alternative.

Acknowledgement

The research is part of the "IMOPEDIA - Innovative Artificial Intelligence Systems in the field of real estate portals" project. Project co-financed from the European Regional Development Fund through the Competitiveness Operational Program, SMIS 129132, call POC/524/2/2, action 2.2.1, call 2. This work was also partially supported by the Ministry of Innovation and Research, UEFISCDI, by the project TRANSLATE, TE 66/2020, PN-III-P1-1.1-TE-2019-0543.

REFERENCES

[1] *Zheng, L., Yang, Y. and Tian, Q., 2017*. SIFT meets CNN: A decade survey of instance retrieval. IEEE PAMI, 40(5), pp.1224-1244.

[2]. *Dalal, Navneet, and Bill Triggs*. "Histograms of oriented gradients for human detection." 2005 IEEE (CVPR'05). Vol. 1. Ieee, 2005.

[3]. *Ojala, Timo, Matti Pietikainen, and Topi Maenpaa*. "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns." IEEE Transactions on pattern analysis and machine intelligence 24.7 (2002): 971-987.

[4]. *Lowe, David G*. "Distinctive image features from scale-invariant keypoints." International journal of computer vision 60.2 (2004): 91-110.

[5]. *Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool*. "Surf: Speeded up robust features ECCV. Springer, Berlin, Heidelberg, 2006.

[6] *Calonder, M., Lepetit, V., Strecha, C., & Fua, P.* (2010, September). Brief: Binary robust independent elementary features. ECCV (pp. 778-792). Springer, Berlin, Heidelberg.

[7] *Zhou, W., Newsam, S., Li, C., & Shao, Z.* (2017). Learning low dimensional convolutional neural networks for high-resolution remote sensing image retrieval.Remote Sensing, 9, 489

[8] *Yue-Hei Ng, J., Yang, F., & Davis, L. S*. (2015). Exploiting local features from deep networks for image retrieval. In Proceedings of the IEEE CVPR (pp. 53-61)

[9] *Jégou, H., Douze, M., Schmid, C., & Pérez, P*. (2010, June). Aggregating local descriptors into a compact image representation. In 2010 IEEE CVPR (pp. 3304-3311).

[10] *Wen, Y., Zhang, K., Li, Z., & Qiao, Y*. (2016, October). A discriminative feature learning approach for deep face recognition. ECCV (pp. 499-515). Springer, Cham

[11] *Cai, J., Meng, Z., Khan, A. S., Li, Z., O'Reilly, J., & Tong, Y*. (2018, May). Island loss for learning discriminative features in facial expression recognition.2018 13th IEEE ICAF & Gesture Recognition (FG 2018) (pp. 302-309).

[12] *Ye, Y., Zhang, T., & Yang, C.* (2019, July). Fisher Loss: A More Discriminative Feature Learning Method in Classification. In *2019 IEEE/ASME (AIM)* (pp. 746-751). IEEE.

[13] *Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., & Wei, Y.* (2020). Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF CVPR* (pp. 6398-6407

[14] *F. Schroff, D. Kalenichenko, and J. Philbin*, "Facenet: A unified embedding for face recognition and clustering," in Proceedings of the IEEE conference on CVPR 2015, pp. 815–823.

[15] *Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., & Torralba, A. (2010, June).* Sun database: Large-scale scene recognition from abbey to zoo. In 2010 IEEE CVPR (pp.3485-3492).

[16] *Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., & Torralba, A. (2017*). Places: A 10 million image database for scene recognition. IEEE PAMI, 40(6), 1452-1464

[17] *Van der Maaten, L., & Hinton, G. (2008)*. Visualizing data using t-SNE. Journal of machine learning research, 9(11)