Learning Non-Metric Visual Similarity for Image Retrieval : A Reappraisal

Lukman Olagoke
pennsylvester2@gmail.com

ABSTRACT

An efficient similarity measure is an essential core of any similar image retrieval model since it is responsible for capturing the (dis)similarity of the perceptual stimuli from the acquired vectors of visual descriptors. However, it is not often clear how the commonly used standard metric measure on this feature vector space corresponds to the observed differences in the perceptual space. The reviewed paper argues that a non-metric visual similarity based on neural network performs better than standard metric distance in measuring visual similarity of images. The aim of this review is to study the impact of non-metric visual similarity and standard metric in similar image retrieval.

CCS CONCEPTS

• **Information systems** → *Similarity measures.*

KEYWORDS

deep neural networks, similarity measure, content based Image retrieval, visual similarity, metric learning, Convolutional Neural Network.

1 INTRODUCTION

The paper under review [5] focuses on content based image retrieval approach called search by example, which requires learning of features and similarity metrics necessary to distinguish between objects within the same category - i.e., *fine-grained object similarity* . Another approach would be *search by category*, where two images are considered similar if they belong to the same category -i.e.,*category level similarity* [11, 18].

Early approaches to fine-grained object retrieval requires the use of hand-crafted features as discussed in [4, 14]. These features are subsequently used to learn similarity [2, 3]. In this review, deep neural network especially Convolutional Neural Network (CNN) and its variants have been used for feature extraction as discussed in [7, 8, 19]. Subsequently, image similarity is learnt using these features.

Paper Review , August 11, Moscow.
2019. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 ARCHITECTURE SUMMARY

The non-metric visual similarity network proposed by the authors was trained on an existing CNN visual feature extractor called regional maximum activation of convolution (R-MAC) described in [19]. R-MAC was itself built on the pre-trained CNN model described in [16]. R-MAC effectively discards the fully connected layer of the pre-trained model and uses the Maximum Activation of Convolution technique for feature extraction. Summarily, the authors fed these visual descriptors into a similarity network called *SimNet*. The SimNet consisted of a set of fully connected layers with dimensions as shown in Table 1.

Table 1: Composition of SimNet

Layers	Size	Comments
Input layer	$1 \times K \times 2$	
First layer	$1 \times K \times 2 \times Ch$	Ch is number of channels
Hidden layers	$1 \times Ch \times 2 \times Ch$	
Output layer	$1 \times Ch \times 2 \times 1$	This is followed by ReLu

Training on Architecture

The R-MAC and SimNet networks were trained end-to-end using easy and difficult image pairs respectively. Four different configurations with different numbers of hidden layers and channels were tested. The training involved feeding two images into separate R-MAC architectures. The resultant features are then concatenated and fed into the SimNet for similarity computation. The training required that image pairs were labelled as similar pair (or not) and that an annotated similarity score is assigned to the pairs.

3 STRONG AND WEAK POINTS

Strong Points

The strong points of the paper are highlighted below:

- The authors were able to motivate the need for a non-metric similarity.
- The adaptability of the model - The SimNet could be adapted for use with other state-of-the art architectures.
- The results from the paper extended the state-of-the-art using well known data sets.

- The model training used both easy and hard data sample pairs in order to achieve good performance.
- The model was evaluated on both off-shelf and fine-tuned state-of-the-art models for comprehensive comparison.

In addition, the authors proposed a new loss function for training similarity :

$$\mathcal{L} = |s_{i,j} - l_{i,j}(sim(x_i, x_j) + \Delta) + (1 - l_{i,j})(sim(x_i, x_j) - \Delta)| \quad (1)$$

where $s_{i,j}$ is the similarity score. $sim(x_i, x_j)$ is the cosine similarity between the output feature vectors from R-MAC-SimNet architecture. Δ is the margin parameter. If I_i and I_j are 2 images, then:

$$l_{i,j} = \begin{cases} 1, & \text{if } I_i \text{ and } I_j \text{ are similar} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Weak Points

Training Evaluation.

- There is an issue with using OASIS [3] in the evaluation of results. OASIS is not based on CNN architecture and CNN architectures are known to have good representation power [6]. How do we decide if the improvement in performance is due to the R-MAC representational power or the added SimNet or both? Evaluation of the OASIS could have been done using OASIS versus OASIS-SimNet architecture. It is then possible to attribute the improvement in performance to the SimNet.
- OASIS is applicable to online large scale retrieval. The authors did not mention or show that the end-to-end architecture could be adapted for large scale retrieval. As noted in [19], the aim of R-Mac was not large scale retrieval but to present the representational power of the activation layers of CNN. How could R-MAC-SimNet be adapted to large scale retrieval and then consequently compared to an architecture such as OASIS for fair comparison?
- It would have been interesting to incorporate SimNet into deeprank model discussed in [21] by replacing the euclidean distance of the deeprank model with the SimNet to test for any improvement in performance. Deeprank uses CNN architecture for feature representation and it outperforms OASIS. If deeprank-SimNet architecture performs better than the deep rank architecture: then one could attribute the improvement in performance to SimNet. The same performance test could also be applied to the end-to-end architecture of [9].

Loss Computation and distance measure.

- The architecture uses cosine similarity. It is known that the cosine similarity equals the euclidean distance when the features are l_2 normalized. Note that for l_2 normalized vectors $\mathbf{x}$ and $\mathbf{y}$, then

$$\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1$$

It follows that :

$$\begin{aligned} \|\mathbf{x} - \mathbf{y}\|_2^2 &= (\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y}) \\ &= \mathbf{x}^T\mathbf{x} - 2\mathbf{x}^T\mathbf{y} + \mathbf{y}^T\mathbf{y} \\ &= 2 - 2\mathbf{x}^T\mathbf{y} \\ &= 2(1 - \cos \angle(\mathbf{x}, \mathbf{y})) \end{aligned} \quad (3)$$

Thus we are back to the case of the euclidean distance which is a metric as defined by the properties mentioned in [5] itself.
- The loss function in equation 3 is defined as an absolute value. It is well known that it is not differentiable everywhere (specifically at zero). This is very easy to observe in the case where $\Delta = 0$ and images are similar ($l_{i,j} = 1$). In such case, we have that:

$$\mathcal{L} = |s_{i,j} - (sim(x_i, x_j)|$$

It was not stated how this would be tackled or was tackled. An alternative would be to use max-margin loss or propose certain heuristics at the zero point or use sub-gradient method [12].

Metric similarity and perceptual representation.

- First, as mentioned in [23] perceptual space is the space "out there" that we can often feel with our body and is characterised by three-dimensionality. Conceptual space is formulated in terms of abstract continuum [23], defined by mathematics and physics. A psychological space is space-as-experienced and can not be readily measured using any of the sense data that are used to define perceptual or conceptual space.
- Secondly, as discussed in [13] object similarity is intimately connected with the idea of geometric representation of stimuli in perceptual space. However, [20] opined that the closeness of stimuli and (object) similarity in a geometric representation can not be effectively captured by metrics that are based on segmental addition (eg euclidean distance metric).
- Thirdly , as discussed in [13], similarities based on Shepard law of universal similarity [15] are not affected by this criticism in [20]. Hence, it could be argued that given the transformations induced on the image space by the neural network, the metric distance between the psychological space and conceptual space representations decreases exponentially as discussed

in [13, 15]. The sketch of the argument is shown below:

d_p = distance induced on the psychological l_p space

The Shepard law says that the distance between the psychological lp norm and the measured similarity is exponentially decreasing:$K(x, y) = \exp(-d_p(x, y)^q)$

$$\begin{aligned} \bar{d_p} &= \|\phi(x) - \phi(y)\|^2 \\ &= \phi(x)^T\phi(x) - 2\phi(x)^T\phi(y) + \phi(y)^T\phi(y) \\ &= 2 - 2K(x, y) \\ &= 2 - 2\exp(-d_p(x, y)^q) \end{aligned} \tag{4}$$

In the above, q is an hyperparameter, x, y are feature vectors, and ϕ represent transformation on this feature space by neural net (say the final hidden layer). In essence we expect the difference in dissimilarity between the real measurement and geometrical distance to decrease exponentially.

- Lastly, as hypothesized in [25], perceptual similarity is not a special function but a consequence of visual representation tuned to be predictive about important structures in the world. Representations that are effective at semantic prediction tasks are also representations in which euclidean distance is highly predictive of perceptual similarity judgments.

4 FOLLOW UP STUDY TO THE PAPER

General consideration

- The paper introduces the similarity network. However, it will be worth while to focus on end-to-end architecture that seamlessly incorporates visual descriptor architecture and prediction in one pipeline. The impact of the similarity network could be tested by evaluating against standard metric distances.
- Models such as [9, 21, 25] have provided a general framework from which one could get an intuition. One would simply use a supervised learning method that given any 2 image feature vectors, predicts the similarity score. In fact, [25] has a prototype similarity model. The truth value could be generated using the *structural similarity index* as discussed in [22] or the perceptual or difference Hash as discussed in [24]. The challenge would be doing the supervision on the CNN architecture.

Proposed Models

First Model. A new model, shown in figure 1, is proposed. The intuition for this architecture comes from [5, 8, 25]. The architecture in fig1 is similar to [8]. The region proposal network from [8] is retained. However, the important differences are discussed below:

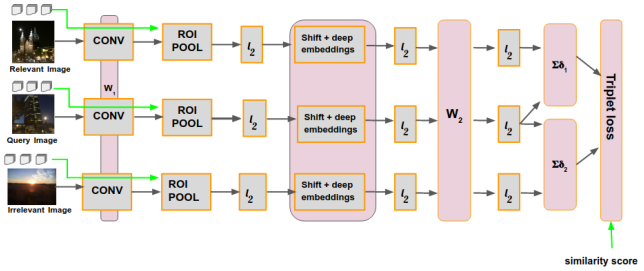

Figure 1: Proposed architecture. End-to-End similarity computation and margin maximization. $\mathbf{W}_1, \mathbf{W}_2$ are shared weights. $\Sigma\delta_1$ are the difference computation

- The CNN architecture extracts feature stack which are then normalized in the channel dimension.
- The activations are scaled channel-wise with the vector $\mathbf{W}_2$ which is shared across the triplet architecture.
- Given a feature output of L layers, if we designate $y_q^l, y_i^l \in \mathcal{R}^{H_l, W_l, C_l}$ for layer l,then the distance becomes the average spatial summed across each channel given as:

$$\Sigma\delta_1 = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \|y_q^l - y_i^l\|_2^2$$

- For any pair of images or regions x_1, x_2 we evaluate their structural similarity score [22] and difference hash [24]. Then the similarity score becomes:

$$\text{similarity\_score}(x_1, x_2) = \frac{\text{structural similarity index}(x_1, x_2)}{\text{difference Hash}(x_1, x_2)}$$

- The loss function is computed as below:

$$\mathcal{L}(I_q, I_r, I_i) = \max(0, \text{similarity\_score} + \Sigma\delta_1 - \Sigma\delta_2)$$

q, i, r signify query,relevant and irrelevant images. The sub-gradients computation follows as defined in [1].

In general a Siamese architecture (2 input network) could also have been an excellent proposal. In this case one needs to train on easy and difficult sample pairs in order to enhance performance.

To evaluate the model, the pipeline in figure 2 is proposed. The following salient points about this architecture:

- A Siamese architecture is assumed.
- Given a query image, to generate similar image, select any image at random from the data base.
- Compute the similarity score. If the similarity score is above a defined threshold, we refer to this as a positive sample. Re-rank the images in the data base such that images with close difference hash values to positive samples are the once that will be tested for comparison with query image. Continue to evaluate, each time selecting and re-ranking till only best samples are at

the top. Select best top-k ($k \in \mathcal{N}$) from the collected best samples.

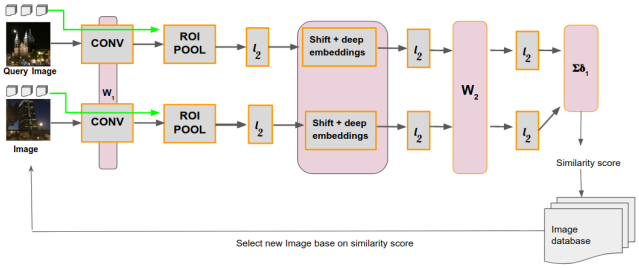

Figure 2: Proposed architecture. End-to-End similarity model evaluation.

Second Model. The second model proposed is based on [21]. A multilayer perceptron (MLP) is added to the architecture as shown below. The aim of the MLP layer is to take a linear combination of the feature vectors and compute a score. This score is compared to the pre-computed similarity score (as given in first model). The difference in similarity score is incorporated in the computation of max-margin loss.

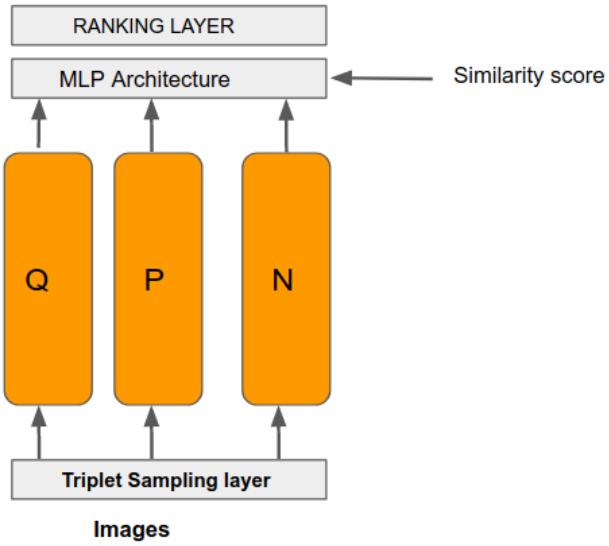

Figure 3: Proposed architecture based on deep rank model. An MLP layer has been added.

5 CONCLUSION

The authors have introduced a new idea for the robust computation of similarities given any pair of visual descriptors. This idea could be incorporated in state-of-the-art architecture in image retrieval. Overall, the idea presented by the authors is a crucial one that still needs to be revisited and studied.

REFERENCES

[1] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. 2018. Automatic differentiation in machine learning: a survey. *Journal of machine learning research* 18, 153 (2018).

[2] Y-Lan Boureau, Francis Bach, Yann LeCun, and Jean Ponce. 2010. Learning mid-level features for recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Citeseer, 2559–2566.

[3] Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. 2010. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research* 11, Mar (2010), 1109–1135.

[4] Navneet Dalal and Bill Triggs. 2005. Histograms of oriented gradients for human detection.

[5] Noa Garcia and George Vogiatzis. 2017. Learning Non-Metric Visual Similarity for Image Retrieval. *CoRR* abs/1709.01353 (2017). arXiv:1709.01353 http://arxiv.org/abs/1709.01353

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 580–587.

[7] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. 2017. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision* 124, 2 (2017), 237–254.

[8] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. 2017. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision* 124, 2 (2017), 237–254.

[9] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. 2017. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision* 124, 2 (2017), 237–254.

[10] Gregory Griffin, Alex Holub, and Pietro Perona. 2007. Caltech-256 object category dataset. (2007).

[11] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. IEEE, 1735–1742.

[12] Michael Held, Philip Wolfe, and Harlan P Crowder. 1974. Validation of subgradient optimization. *Mathematical programming* 6, 1 (1974), 62–88.

[13] Frank Jäkel, Bernhard Schölkopf, and Felix A Wichmann. 2008. Similarity, kernels, and the triangle inequality. *Journal of Mathematical Psychology* 52, 5 (2008), 297–303.

[14] David G Lowe et al. 1999. Object recognition from local scale-invariant features.. In *iccv*, Vol. 99. 1150–1157.

[15] Roger N Shepard. 1957. Stimulus and response generalization: A stochastic model relating generalization to distance in psychological space. *Psychometrika* 22, 4 (1957), 325–345.

[16] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[17] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[18] Graham W Taylor, Ian Spiro, Christoph Bregler, and Rob Fergus. 2011. Learning invariance through imitation. In *CVPR 2011*. IEEE, 2729–2736.

[19] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. 2015. Particular object retrieval with integral max-pooling of CNN activations. *arXiv preprint arXiv:1511.05879* (2015).

[20] Amos Tversky and Itamar Gati. 1982. Similarity, separability, and the triangle inequality. *Psychological review* 89, 2 (1982), 123.
[21] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. 2014. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1386–1393.
[22] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.
[23] John Welwood. 1977. On psychological space. *The Journal of Transpersonal Psychology* 2 (1977), 97–118.
[24] Christoph Zauner. 2010. Implementation and benchmarking of perceptual image hash functions. (2010).
[25] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 586–595.

A EXPERIMENTAL DETAILS

In order to proof the concepts discussed in this paper, a very brief training and evaluation model was done on the caltech dataset 101 [10] which contains 250 categories of objects. From each category 9 objects were selected.

Training

Each object is transformed to a feature output vector of size 1x2048 using a pre-trained ResNet model [17]. This is followed by randomly selecting 2 pairs of feature vectors and computing the square of their differences. The square of their differences is used as the input into an MLP similarity network.

The target value was computed using the structural similarity index divided by the difference hash as described in the first proposed model. In summary, the MLP takes in square of the difference between any pair of image feature vector and output the target value.

Evaluation

This simple model was evaluated against an exactly similar model but which uses the euclidean distance as a measure of similarity instead of the MLP similarity network. To evaluate the model, one selects a query image randomly and let the network output the top-10 similar objects. It should be noted that feature vector is first extracted from the query image. Afterwards, the query feature vector is compared against the database of feature vectors using euclidean distance and the MLP network as similarity measure respectively.

It should be stated that the computation was carried out on a personal computer without GPU capability. This means the MLP architecture might not be as robust as expected but it still looks very promising.

The query image is shown below:

Figure 4: Query image for model evaluation

The retrieved image using euclidean distance as metric:

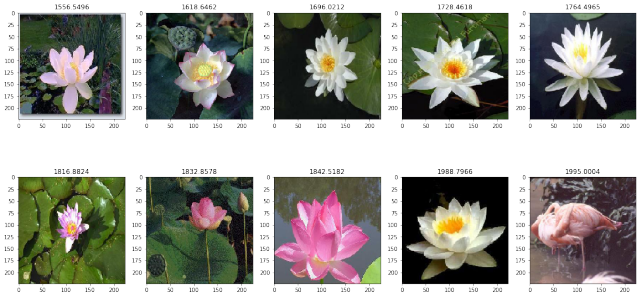

Figure 5: Evaluation of Model using euclidean distance

The retrieved image using the MLP similarity network:

Figure 6: Evaluation of Model using MLP similarity network

As usual with any deep neural network model, the parameters still needs to be tuned to achieve a better result. In addition more samples and training epochs will be needed which the time constraint will not permit me. However, it is a step in the right direction. The code for the model is on my Github reporistory. [1]

[1] https://github.com/adderbyte/content_based_image_retrieval