Martínez-Cortés, Tomás, González-Díaz, Iván, Díaz-de-María, Fernando. (2018). Automatic learning of image representations combining content and metadata. *2018 25th IEEE International Conference on Image Processing (ICIP)*, 7-10 Oct. 2018, Athens, Greece. Pp.: 1972-1976.

DOI: https://doi.org/10.1109/ICIP.2018.8451566

AUTOMATIC LEARNING OF IMAGE REPRESENTATIONS COMBINING CONTENT AND METADATA

Tomás Martínez-Cortés, Iván González-Díaz, Fernando Díaz-de-María

Department of Signal Theory and Communications, Universidad Carlos III, Leganés (Madrid), Spain

ABSTRACT

Content-based image representation is a very challenging task if we restrict to their visual content. However, associated metadata (such as tags or geolocation) become a valuable source of complementary information that may help to enhance the current system performance. In this paper, we propose an automatic training framework that uses both image visual contents and metadata to fine tune deep Convolutional Neural Networks (CNNs), providing better image descriptors adapted to certain locations, such as cities or regions. Specifically, we propose to estimate some weak labels by combining visual- and location-related information and incorporate them to a novel loss-function over pairs of images. Our experiments on a landmark discovery task show that this novel training procedure enhances the performance up to a 55% over well-established CNN-based models and is free from overfitting.

Index Terms— CNN, metadata, loss function, weak labels

1. INTRODUCTION

In 2017, Facebook's users generated a total of 300 million photos per day[1]. The amount of new multimedia content has grown exponentially for the past decade, and it is now so staggering that storing, managing, indexing and organizing user-generated files efficiently is one of the main technological challenges for the industry. During the last few years, the scientific community has tackled this problem by means of novel computer vision techniques aiming to automatically obtain content-based image descriptors which are distinctive, compact, and allow an efficient search [1][2][3].

Convolutional Neural Networks (CNNs) [4] have shown their superior performance in a variety of tasks and also in this particular field. Modern deep classification models, such as residual networks [5], have achieved human-like performance on the ImageNet challenge [6], where a thousand object categories are recognized in a set of a few million images. These networks are able to learn the common visual patterns of the objects belonging to the same category, i.e., objects of the same category are described by similar feature vectors regardless of the intra-class variability. More recently, some research works such as [7] have developed Recurrent Neural Networks (RNN) that combine object recognition and language models to generate natural language-based descriptions of the image contents in a human-like way. Describing the image content using natural language is particularly useful for presenting humans with captions or doing content-based searches from textual queries.

This work has been partially supported by the National Grants TEC2014-53390-P and TEC2017-84395-P of the Spanish Ministry of Economy and Competitiveness. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the TITAN X GPU used for this research.

[1]https://zephoria.com/top-15-valuable-facebook-statistics/

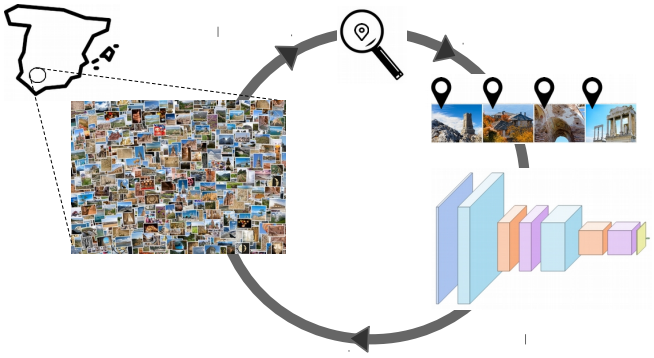

Fig. 1: The proposed system: first, all images available from a region are gathered; then, we select the geolocated and use them to train a location-adapted CNN; finally, we employ this model to describe the whole set of images.

Nevertheless, in some scenarios, we are more interested in identifying instances of specific objects rather than general categories (detecting a particular car make and model, instead of just cars). For such a task, it is common to employ image matching techniques which put more emphasis on highly discriminative capabilities. For the particular problem of content-based image retrieval, the work in [8] proposed a new CNN-based architecture called *deep-retrieval*, which outperforms most of the classic image matching methods at a fraction of their computational cost. Although it achieves impressive results in several well-known datasets for retrieval, its performance is still limited due to the fact that even images that do not represent the same object yet share certain visual patterns (e.g. various churches of the same style may share certain architectonic elements, several car models of the same make usually share many elements and external finishes). Overcoming this issue is not straightforward and would require some degree of supervision to help retrieval systems to infer which elements or details are actually discriminating. However, annotating datasets is costly and might become impractical in many scenarios.

Alternatively, many public image and video repositories provide metadata (date, GPS, tags, titles, etc.) associated with the contents that may help to improve the automatic description of the images. In [9], the authors proposed a system that finds out complementary information about landmarks from Singapore by combining content-based CNN descriptions with GPS coordinates. However, they propose a supervised approximation which requires labeling thousands of images by hand, jeopardizing its practical application. Automatic photo geotagging is also an active field [10][11][12], where coordinates for a new image are estimated from those exhibiting the most similar content in a reference geotagged repository. Similarly, user-

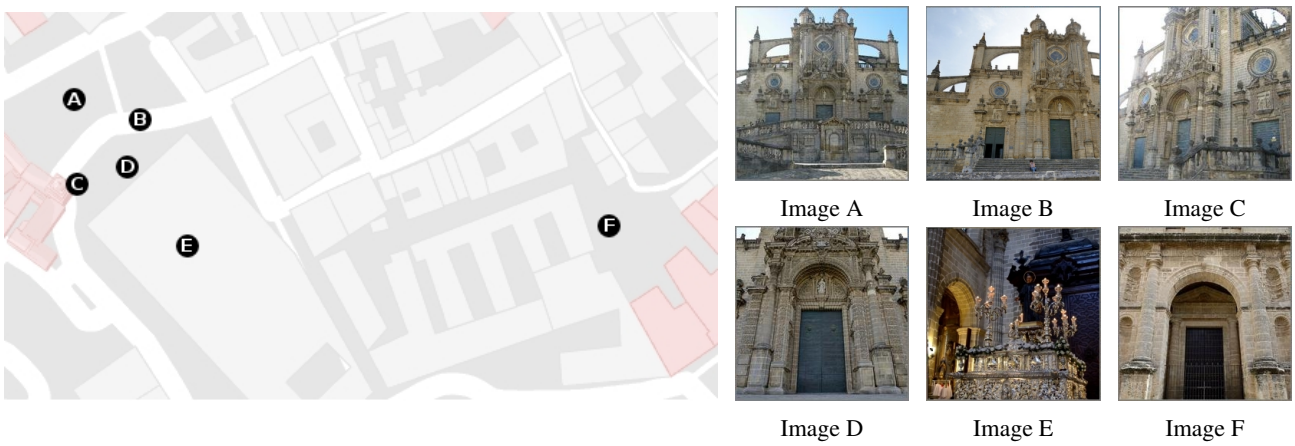

Fig. 2: (left) Geo-location of six photos shown on a map of Jerez (Spain). (right) Corresponding photos.

provided tags or titles have been also used to retrieve representative views of landmarks worldwide [13]. The success of these methods is strongly related to the availability and quality of metadata associated with contents. Nevertheless, only a fraction of the users sharing their photos in public repositories actually provide these metadata. Taking GPS coordinates as an illustrative example, a simple search in *Flickr* with the tag *London* yields 16% of geo-located images. Similar values are achieved for *Berlin*, 18%, and *Madrid*, 15.1%. Furthermore, GPS coordinates, user tags or even titles are often noisy, which dramatically limits their usability [14].

In this paper, we propose a novel training framework that relies on both, image content and metadata in the form of geo-location, to automatically learn location-adapted deep models that provide properly tuned image descriptions for those visual contents found in that location. Avoiding any kind of supervision (beyond that inferred from the available, noisy metadata), we learn our models from the subset of images with GPS coordinates, and then apply them to the whole set of visual contents. For this purpose, we propose a specifically tailored cost function that makes use of weak labels estimated from image descriptors and geo-location. We use this loss function to fine tune a baseline model aiming to suitably represent images from a particular city or region, thus boosting the system performance in subsequent tasks such as landmark discovery or image retrieval.

The remainder of the paper is organized as follows: Section 2 describes our learning framework, including the custom loss function and the process to combine image features and metadata. Section 3 describes our experiments and discuss the results. Finally, Section 4 draws our conclusions and outlines further work.

2. SYSTEM DESCRIPTION

Our system departs from an initial model that was originally trained for a different task, such as classification or retrieval. We refer to this model as **baseline model** and to our proposed location-adapted networks as **location-CNNs**. Figure 1 summarizes our proposal. First, we gather from Flickr [2] all images from a particular location (city, region, etc.) using geo-location and textual tags. Then, we use the subset of geo-located images to train a location-adapted CNN using a cost function that works with soft labels derived from image visual descriptors and GPS coordinates. Finally, the learned model is used to describe the whole set of photos from that particular place, enabling subsequent user-end applications such as landmark discovery, image retrieval and annotation.

[2]www.flickr.com

2.1. Cost Function

The training of the location-CNNs is carried out using a cost function that aims to learn visual descriptors that bring together pairs of images showing related views of the same landmark, while pushing away other pairs. Considering the example shown in Fig. 2, we would like our network to provide similar visual descriptors for the images in pairs (A,B) and (B,C), and push away the descriptors of pairs (C,E) and (D,F). This kind of cost functions working with pairs of images have been previously proposed in the context of pure matching tasks [15][16]. Nevertheless, we have developed a novel version of those functions that computes the loss for the i-*th* pair as follows:

$$L_i = \frac{1}{2} y_i d_{V_i}^2 + \frac{1}{2} (1 - y_i) \, max \left(0, m - d_{V_i}^2\right) \qquad (1)$$

where $d_{V_i}^2$ is the square Euclidean distance between the visual features of the i-*th* pair, computed with our location-CNN model; $m \geq 0$ is a margin that avoids that very dissimilar pairs keep contributing to the loss; and $y_i \in [0, 1]$ is a soft label that signals whether the generated visual descriptors (of a pair of images) need to get closer ($y_i \geq 0.5$) or further away ($y_i < 0.5$).

To ensure convexity, both the soft labels (y_i) and the margin (m) must be fixed during learning. To that end, we compute the margin and the soft labels for every pair of images using the descriptors provided by the baseline model. The margin is set to the average distance of all the feature pairs, while the soft labels are estimated using the baseline model features and the GPS coordinates as detailed on the next section.

2.2. From Images and Metadata to Soft Labels

The soft label y_i for any pair of images is estimated as a function of their feature-based distance (computed with the baseline model) and their spatial distance. Referring again to the previous example in Fig. 2, images (A,B,C,D,E) are spatially close to each other; however, this fact does not guarantee that their contents are visually related. For instance, image E shows a completely different scene than (A,B,C,D); therefore its descriptor should not be forced to be

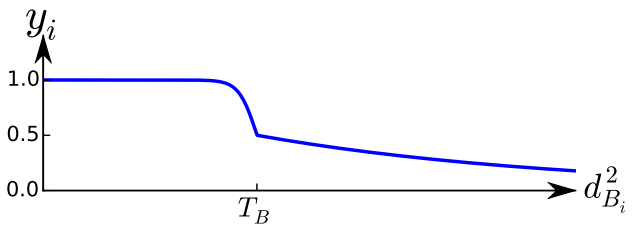

Fig. 3: Piece-wise function defined by equation 3 for $d_{S_i} \leq T_S$ as a function of $d^2_{B_i}$. The left hand side of T_B is almost flat giving positive pairs large weights. The right hand side decreases slowly to give near-the-threshold false negatives small weights in the cost function.

similar to those of the rest. The opposite holds for images (D,F); although they are visually similar, they were taken hundreds of meters away from each other and, in fact, belong to different buildings.

Following these intuitions, we conclude that for a pair of images to be assigned a $y_i \geq 0.5$ they have to be close in terms of both, features and GPS coordinates. For that purpose, we need to set up two thresholds, T_B related to the distance of visual features computed with the baseline network, and T_S associated with the spatial distance. In order to obtain a robust solution that is aware of the actual data statistics, we derive the threshold T_B from the distribution of square visual distances $d^2_{B_i}$ computed using the baseline model. Specifically, assuming a Gaussian distribution:

$$T_B = \mu_B - k\sigma_B \tag{2}$$

where μ_B is the average and σ_B the standard deviation. We have found experimentally that a suitable range for k is from 2.0 to 2.5. Concerning the spatial distance, a simple threshold of $T_S = 300$ meters performs well in our training sets and generalizes properly on unseen locations.

Once we have settled both thresholds, we use the following piece-wise function to compute the soft label y_i based on visual $d^2_{B_i}$ and spatial d_{S_i} distances:

$$y_i = \begin{cases} 0 & \text{if } d_{S_i} > T_S \\ \frac{1}{1+\exp\left(d^2_{B_i} - T_B\right)} & \text{if } d_{S_i} \leq T_S, d^2_{B_i} \leq T_B \\ \exp\left(\frac{\ln(1/2) d^2_{B_i}}{T_B}\right) & \text{if } d_{S_i} \leq T_S, d^2_{B_i} > T_B, \end{cases} \tag{3}$$

where it should be noted that given $d_{S_i} \leq T_S$, i.e., the images of the i-*th* pair are close enough according to their geolocation, the threshold T_B on the visual distance decides between $y_i \geq 0.5$ and $y_i < 0.5$ as illustrated in figure 3.

Let us gain some insight into eq. (3) by discussing its main advantages: (a) the soft labels y_i allow us to put more or less emphasis on certain pairs of images, thus producing stronger gradients for high-confidence pairs and weaker gradients for more doubtful cases (e.g. equation 1 generates zero gradients for $y_i = 0.5$); (b) the piece-wise function allows to establish asymmetric behaviors at both sides of the threshold T_B. In particular, we have observed that, by setting a conservative threshold, image pairs with distances below T_B almost always show the same visual scene, and the variations in the distance are usually due to factors like different viewpoints or varying illuminations. Hence, we have designed a flat curve on this piece of the equation to ensure a similar contribution for all of them. However, if the threshold is conservative, we may yet find related image pairs with distances above T_B. In that case, in order to avoid pushing them too away, a slowly decreasing slope seems to be more appropriate (see Fig. 3).

Table 1: Training and test sets statistics

	Training sets			Test sets	
	Images	Pos pairs	Neg pairs	Images	Landmarks
Jerez	1000	1.5k	250k	1000	19
Madrid	4000	30k	4M	3900	15
Rome	4000	30k	4M	3100	14

3. EXPERIMENTS AND RESULTS

In this section, we describe the data sets, the experimental setup, the evaluation metrics and the final results. Although the main goal of our approach is to provide location-adapted visual features enabling subsequent higher-level tasks, in this paper, we have focused our assessment on the specific task of automatic landmark discovery. For that purpose, our computed visual descriptors are used to feed a clustering algorithm which is in charge of discovering relevant clusters associated with landmarks. In order to separate the analysis of the proposed learning framework from the potential influence of the parameters of the clustering algorithm, we have used a *k-means* algorithm with a pre-defined number of clusters (the number of landmarks we aim to discover). It should be noted that other clustering approaches could also be used in a more realistic setup.

3.1. Datasets, experimental setup and evaluation metrics

Our dataset contains Flickr images from three different cities in Europe: Rome (Italy), Madrid and Jerez de la Frontera (Spain). For each city, a train set of geo-located images has been gathered within a $10km$ radius around the center of each city. Additionally, we have used a list of generic keywords that helps to retrieve images relevant to our task, namely: *landmark, monument, building, park or art*. Finally, we filter out the results allowing only one image per user in order to avoid duplicates. The test set has been built by searching for a predefined list of famous landmarks in each city, and manually cleaning the retrieved results. In order to provide a fair analysis, an image is never present in both sets. Table 1 summarizes the number of images and landmarks per city in the corresponding training and test sets, as well as the resulting number of positive and negative sample image pairs.

Using the corresponding training sets, we have trained three independent CNNs using ResNet50 as our *baseline network*: **JerezNet**, **MadridNet** and **RomeNet**. All the networks were initialized with the weights of ResNet50 trained on ImageNet and were trained for 10 epochs with 1000 batches per epoch using Batch Gradient Descend (BGD) with 40 pairs of images per batch. We have used a sampling strategy to build the batches that ensures that at least 10% of the pairs are positive ($y_i \geq 0.5$) and the rest are negative ($y_i < 0.5$). Additionally, a particular image is only included once per batch, allowing us to safely use BGD instead Stochastic Gradient Descend (SGD), as it is a common practice when working with

Table 2: Average and standard deviation of the Rand, Jaccard and Fowlkes indexes as a result of comparing the proposed models: JerezNet, MadridNet and RomeNet; with the baseline model ResNet50

		Rand	Jaccard	Fowlkes
Jerez	ResNet50	0.9071	0.3203	0.4944
		±0.0201	±0.0145	±0.0208
	JerezNet	0.9242	0.3834	0.5627
		±0.0201	±0.0323	±0.0254
Madrid	ResNet50	0.9254	0.3817	0.5467
		±0.0051	±0.0239	±0.0322
	MadridNet	0.9547	0.5911	0.7551
		±0.0189	±0.0261	±0.0257
Rome	ResNet50	0.9174	0.3728	0.5692
		±0.0157	±0.0298	±0.0182
	RomeNet	0.9345	0.5473	0.6625
		±0.0177	±0.0138	±0.0125

pairwise loss functions. For the parameters update, we have employed 0.9 for the momentum term, 10^{-5} as learning rate and 10^{-3} as weight decay. The affine layers of the original model have been removed and we have kept the output of the last average pooling as our features. The models were trained using the open deep learning library PyTorch[3] on a NVIDIA TITAN XP GPU.

Once our networks have been trained, in order to assess our models in the task of automatic landmark discovery, we have generated the visual descriptors of images in the test set, and used K-means with the pre-defined number of landmarks to cluster these descriptors. The results are then compared with the ground-truth using three classical clustering evaluation metrics: the Rand [17], Jaccard [18] and Fowlkes-Mallows [19] indexes. These indexes are based on counting pairs of images whose members lie in the same or different clusters when comparing the ground truth and the estimated labels, and range from zero to one with one meaning a perfect clustering. It is worth noting that, for the sake of stability and statistical significance, we have repeated the clustering process ten times to account for differences due to K-means initialization.

3.2. Results

Table 2 shows the averages and standard deviations of the Rand, Jaccard and Fowlkes-Mallows indexes obtained in our experiments, using either visual descriptors generated by the baseline network Resnet50 or by our proposed location-adapted CNNs (JerezNet, MadridNet, RomeNet). Results show that for all the evaluation indexes and test sets, the location-adapted CNNs provide a notable improvement over the baseline. The large difference between the Rand and the other two metrics is due to the nature of the measures. The Rand index is highly biased towards true negatives, i.e., pairs of images whose members were not in the same cluster neither in the ground truth nor in the estimated labels, which are the vast majority for any reasonable sized database. The other two indexes neglect true negatives, providing more stable results over different dataset sizes and number of clusters.

[3]http://pytorch.org/

Table 3: Average and standard deviation of the Rand, Jaccard and Fowlkes indexes as a result of clustering images from Madrid and Rome using JerezNet.

		Rand	Jaccard	Fowlkes
Madrid	ResNet50	0.9254	0.3817	0.5467
		±0.0051	±0.0239	±0.0322
	JerezNet	0.9287	0.3867	0.5522
		±0.0153	±0.0238	±0.0301
Rome	ResNet50	0.9174	0.3728	0.5692
		±0.0157	±0.0298	±0.0182
	JerezNet	0.9105	0.3646	0.5525
		±0.0054	±0.0179	±0.0223

3.3. Ablation Study

As an ablation study, we have also tested the ability of the learned models to deal with unseen city landmarks or even other cities. In other words, we would like to check if our models are overfitting the previously seen data and would therefore perform badly on unseen scenes. This would become a significant weakness if we do not have geo-located images of a particular landmark of interest in our training set. Table 3 shows the results achieved by JerezNet when tested in Madrid or Rome. It can be seen that, in this scenario, both the baseline and JerezNet offer very similar performances. This is a very important result since it proves that our model adapts to the trained location, but does not over-fit on the training data. Consequently, it is not necessary to see all the interesting places from a city during training in order to deploy a useful model since, even for those unseen landmarks, the model would perform at least as well as a general CNN such as Resnet50.

4. DISCUSSION

In this paper, we have proposed a novel training framework that relies on image content and metadata to learn location-adapted deep models, that provide tuned image descriptors for specific visual contents. Our networks, which depart from an initial model originally learned for a different task, are trained by means of a custom pairwise loss function using weak labels based on available image metadata. Our experiments on a landmark discovery task show that the proposed location-CNNs achieve an improvement of up to a 55% over the baseline model (Jaccard index on Madrid test set). This implies that the network has successfully learned the visual clues and peculiarities of the region for which it was trained, and generated image descriptors that are better location-adapted. In addition, for those landmarks that were not present on the training set or even other cities, our proposed models performed at least as well as the baseline network used as initialization, which demonstrates the absence of overfitting. Further work will explore other metadata and scenarios where specialized networks are necessary to outperform existing models.

5. REFERENCES

[1] J. H. Su, C. Y. Chin, J. Y. Li, and V. S. Tseng, "Efficient big image data retrieval using clustering index and parallel computation," in *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, Nov 2017, pp. 182–187.

[2] Y. Cao, M. Long, J. Wang, and S. Liu, "Deep visual-semantic quantization for efficient image retrieval," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 916–925.

[3] Xuchao Lu, Li Song, Rong Xie, Xiaokang Yang, and Wenjun Zhang, "Deep hash learning for efficient image retrieval," in *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, July 2017, pp. 579–584.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., pp. 1097–1105. Curran Associates, Inc., 2012.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 770–778.

[6] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li, "Imagenet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014.

[7] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 664–676, April 2017.

[8] Albert Gordo, Jon Almazán, Jérôme Revaud, and Diane Larlus, "Deep image retrieval: Learning global representations for image search," *CoRR*, vol. abs/1604.01325, 2016.

[9] L. F. D'Haro, R. E. Banchs, C. K. Leong, L. G. M. Daven, and N. T. Yuan, "Automatic labelling of touristic pictures using cnns and metadata information," in *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*, Aug 2017, pp. 292–296.

[10] H. J. Kim, E. Dunn, and J. M. Frahm, "Predicting good features for image geo-localization using per-bundle vlad," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 1170–1178.

[11] D. M. Chen, G. Baatz, K. Kser, S. S. Tsai, R. Vedantham, T. Pylvninen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk, "City-scale landmark identification on mobile devices," in *CVPR 2011*, June 2011, pp. 737–744.

[12] Yannis S. Avrithis, Yannis Kalantidis, Giorgos Tolias, and Evaggelos Spyrou, "Retrieving landmark and non-landmark images from community photo collections," in *ACM Multimedia*, 2010.

[13] Lyndon S. Kennedy and Mor Naaman, "Generating diverse and representative image search results for landmarks," in *Proceedings of the 17th International Conference on World Wide Web*, New York, NY, USA, 2008, WWW '08, pp. 297–306, ACM.

[14] Lyndon S. Kennedy, Shih fu Chang, and Igor V. Kozintsev, "To search or to label?: Predicting the performance of search-based automatic image classifiers," in *in ACM MIR*, 2006.

[15] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006, vol. 2, pp. 1735–1742.

[16] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Dec 2016, pp. 378–383.

[17] William M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

[18] Asa Ben-Hur, Andre Elisseeff, and Isabelle Guyon, "A stability based method for discovering structure in clustered data," vol. 2002, pp. 6–17, 02 2002.

[19] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 553–569, 1983.