SCORE-BASED LEARNING FOR RELEVANCE PREDICTION IN IMAGE SIMILARITY SEARCH

Alberto Oliveira, Anderson Rocha

Institute of Computing, University of Campinas (Unicamp)

ABSTRACT

Predicting the performance of queries when labels are not present has been a recurring problem faced in information retrieval systems. Beyond its clear importance, it can also be applied to aid post-retrieval optimization approaches such as re-ranking or rank-aggregation. However, most post-retrieval performance prediction approaches to retrieval systems rely on generating a single effectiveness value of performance for queries. We propose an alternative method to assess the performance of systems reliant on similarity search, which consists of predicting the individual relevance of ranked results according to the distribution of similarity scores of a given query compared to instances in a collection. The idea is that relationships between the ith ranked score and other scores of the rank can be leveraged to generate features which, in turn, are used to classify ranked objects according to their relevance to the query. We propose a positional classification scheme, in conjunction with simple and fast score-based features to predict the relevance of the *top-10* results of a similarity search rank. Our results in nine scenarios, comprising three different large image datasets, show good prediction accuracy for the *top-10* results, with the advantage of being amenable suitable to deploy at query time.

Index Terms— Relevance Prediction, Query Performance Prediction, Similarity Search, Information Retrieval, Machine Learning

1. INTRODUCTION

Several post-query optimization approaches applied to information retrieval systems, such as *rank-aggregation* [1] or *re-ranking* [2], either require, or can be greatly improved by, a correct estimation of the query's performance. In a testing scenario, however, only information pertaining to the query itself can be used for such, since it is likely that no labels are present. Performance estimation through analysis of query-related information is a well-established problem in Information Retrieval, commonly refered to as *Query Performance Prediction* (QPP) [3]. Its main goal is to derive a measure that reflects the overall success of a query. The *clarity score* [3] is a staple on QPP and query-difficulty estimation. Two of the most popular measures for QPP, the Weighted Information Gain [4] and the Normalized Query Commitment [5], are both computed from distributions of similarity scores. Infering the value of a retrieval effectiveness measure, such as mAP [6], is another common approach.

Recently, several works have been proposed for post-retrieval QPP [7, 8, 9, 10, 11, 12, 13]. Zhang et al. [13] proposed a classification strategy, which classifies queries in three categories (easy, medium, hard) according to features extracted from the statistics of the distribution of scores. *NeuralQPP*, from Zamani et al. [12], uses three neural components, one for $top\text{-}k$ scores, one for term distribution, and one for representation of documents in the semantic space. Using weak supervision, the authors train their three neural components for QPP. Sun et al. [11] use contextual information from ranked lists of results to create a feature-matrix, and a convolutional neural network for classification. Their work focus is, like ours, on content-based image retrieval tasks.

Those methods, however, lack any *positional* information about the quality of results. In this work, we explore an alternative formulation of the QPP problem, which we dub *Relevance Prediction*, aiming at quality assessment of individual results within a rank resultant from similarity search. We show that such prediction for similarity search, which is commonly used in Information Retrieval Systems, is not only feasible, but works well enough with minimal overhead added to query efficiency.

Behind the concept of relevance prediction is the hypothesis that both the rank and distribution of scores produced by a similarity search engine hold clues to the performance of the system, and those clues can be exploited to determine which of the ranked objects are likely relevant to the query at hand. This is closely related to the problem of Meta Recognition [14, 15], deciding whether the output of a classifier is a match or non-match, with two important differences: (1) in relevance prediction we are concerned with multiple outputs instead of the top only; and (2) the prediction is applied to objects of the same nature as the probe. A $rank\text{-}k$ Relevance Prediction system is concerned with finding relevant/non-relevant labels for the $top\text{-}k$ ranked objects for a certain query of a retrieval system.

Inspired by the work of Scheirer et al. [15], we introduce an approach to $rank\text{-}k$ Relevance Prediction based on learning features extracted from rank scores which employs multiple positional classifiers to predict the relevance of the $top\text{-}k$ results in a rank. We evaluate the proposed method in three different datasets, each of which with three variations, covering a wide range of similarity search setups, such as different descriptors, metric used, or query perturbations. Altogether, our results show that the proposed methods obtain good and consistent results between the many tested scenarios, especially considering the simplicity of the descriptors employed. Furthermore, both tecnhiques are fast, since they require only feature extraction and testing, thus being easy to deploy at query time.

The remainder of this paper is organized as follows: Section 2 gives a formal description of the Relevance Prediction problem while Section 3 presents our approach to solving it. Section 4 describes our evaluation strategy and presents the obtained results. Finally, Section 5 concludes this work and discusses what we expect for future work in this field.

The research for this paper was financially supported by the São Paulo Research Foundation (FAPESP), DéjàVu grant #2017/12646-3; by the Coordenao de Aperfeioamento de Pessoal de Nvel Superior (CAPES), DeepEyes grant; and by the Conselho Nacional de Desenvolvimento Cientfico e Tecnolgico (CNPq), grant #304472/2015-8.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

2. RELEVANCE PREDICTION

Information Retrieval (IR) systems are tasked with satisfying some information need posed by their users. In practical terms, this is often retrieving, from a collection of objects $\mathcal{C}$, the most similar objects to some query object o_q. We use *object* as a general term for multimedia data sources, such as text, images, or videos. IR systems are frequently modeled around the similarity search problem. For a set of points P in a metric space $\mathcal{M}$ and given a query point $q \in \mathcal{M}$, the nearest-neighbors similarity search problem consists in finding the $P_q \subseteq P$, such that $|P_q| = k$ and P_q are the closest points to q in $\mathcal{M}$. In IR, similarity search is applied as a mean to retrieve the most similar objects to a query object o_q, forming a rank $R_q \subseteq \mathcal{C}$, such that the objects in R_q are ordered according to their similarity to o_q.

Relevance is a pivotal concept within the study of IR systems, often used to denote if a retrieved document satisfies the information need presented by a user. An example of such is: consider an IR system for retrieving pictures of buildings. Considering a query picture of a certain building, relevant information for this particular query are images from the collection featuring the same building. Clearly, the notion of relevance is encoded within the objective of the IR system itself. Performance of IR systems is commonly measured by the amount and ordering of relevant objects within a rank.

While Query Performance Prediction is an indirect way to assess the performance of an IR system without relevance labels, Relevance Prediction is concerned with predicting the relevance labels themselves. Thus, the objective of a $rank\text{-}k$ relevance prediction system is to find a sequence of labels $\mathbf{P}^q = \{p_1, p_2, ..., p_k\}$, such that $p_i = 1$ if the ith ranked object is predicted as relevant, and $p_i = 0$ otherwise. A clear advantage of this formulation is the positional information obtained when good predictions are made. Nevertheless, it is also a more difficult problem since relevance is such an intrinsic characteristic of IR systems, and thus hard to predict.

3. LEARNING-BASED RELEVANCE PREDICTION

Inspired by the work of Scheirer et al. [15], which presented a method for learning-based failure detection in classification systems, we propose a learning-based relevance prediction approach based on simple and fast features extracted from an ordered set of similarity scores. Our focus were descriptors which do not add significant overhead to the retrieval procedure, in such way that any decision stemming from the prediction can be quickly performed in sequential order. Additionally, all features are independent to the metric space used, requiring only that a single measurement of similarity is available. Despite this, both the classification scheme and features are easy to extend to a multiple similarity score scenario.

Our approach employs k independent classifiers, one for each of the $top\text{-}k$ positions. The features, however, are not computed independently. A feature computed for the third rank position, for example, still depends on the similarity value of the remaining $i \in \{1, 2, 4, ..., k\}$ positions of the rank. The features of Section 3.1 were used in conjunction with a *Support Vector Machines* [16] classifier, using an radial basis function kernel [17], in a 5x2 cross validation scheme consisting of a two-fold division, in which each set is used as training and test once, repeated for five rounds. The number of positive (relevant) and negative (non-relevant) examples for the classifier is directly tied to the performance of the ranking system, specifically for the $top\text{-}k$ positions. If the ranking system has very high performance for the $top\text{-}k$ positions, it is possible that some of those positions have no non-relevant examples for training at all. A simple workaround is to employ a one-class SVM classifier instead of a two-class one when no negative examples are present within our training set.

3.1. Features From Similarities

Following the definitions in Section 2, suppose that, for some query q, we have a totally ordered set of similarity scores $S_q = \{s_1, s_2, ..., s_n\}$. Bellow, we define three score-based features used alongside with relevance predictors.

- **Delta a Feature (Δ_a):** This feature, computed for position i of the rank, is the vector:

$$\vec{V}^i_{\Delta_a} = \langle (s_1 - s_i), ..., (s_{i-1} - s_i), (s_{i+1} - s_i), ..., (s_a - s_i) \rangle \tag{1}$$

  such that a is an adjustable parameter that controls the length of the feature vector, such that $|\vec{V}^i_{\Delta_a}| = a - 1$. This feature explores the changes between sequential scores, centered at the target position, to describe whether the object at the position is relevant or not.

- **Shift DCT b Feature ($sDCT_b$):** This feature consists of computing the *Discrete Cosine Transform* (dct) [18] of the scores from rank position i to position $i + (b - 1)$:

$$\vec{V}^i_{sDCT_b} = dct(\{s_i, s_{i+1}, ..., s_{i+b-1}\}) \tag{2}$$

  again, b is an adjustable parameter that control the length of the feature vector, such that $|\vec{V}^i_{sDCT_b}| = b$. This feature has also been employed by Scheirer et al [15], with the rationale that such transform has been shown to be a good way to represent the information within a score series [19].

- **Fusion Feature (FS_{ab}):** This feature is the concatenation of the two aforementioned features into a single feature vector, or:

$$\vec{V}^i_{FS_{ab}} = \vec{V}^i_{\Delta_a} \frown \vec{V}^i_{sDCT_b} \tag{3}$$

  such that $\frown$ is the **concatenation** operator. By directly combining both features into a single feature vector we expect to explore any complementarity existent between both aforementioned features. The values of a and b control the length of the individual feature vectors, such that $|\vec{V}^i_{FS_{ab}}| = (a - 1) + b$

3.2. Similarity-Search Implementation

Ranking with similarity search was performed in three image datasets, Places365 [20], Vggfaces [21], and Imagenet [22]. While the first dataset was considered in its entirety, the other two were sampled. We adapted the three datasets from being originally classification problems to retrieval problems instead, by considering as relevant other images from the same class as the query. Image descriptions are generated globally from deep neural networks, tuned to the datasets in question. Finally, we generate ranks by performing nearest-neighbors similarity search using the image descriptors extracted. Table 1 summarizes each search scenario performed, alongside some additional information about the adopted image sets.

4. EVALUATION & DISCUSSION

4.1. Evaluation Setup

With nine different search senarios, we aimed at covering variations such as search performance, similarity metrics, and descriptors. Fur-

Table 1. Summary of Relevance Prediction Scenarios

dataset	type	size	# of queries	alias	descriptor	metric	query postprocess?	P@10
vggfaces[21]	faces	∼262k	7,866	VGGF VGG16-L2Sq	vgg16(1x2622)	L2 Squared	no	95.4%
				VGGF VGG16-L2Sq + Pert	vgg16(1x2622)	L2 Squared	yes	66.7%
				VGGF VGG16-Cos	vgg16(1x2622)	Cosine	no	96.3%
places365[20]	scenes	∼330k	3,650	P365 VGG16-L2Sq	vgg16(1x365)	L2 Squared	no	41.8%
				P365 VGG16-Cos	vgg16(1x365)	Cosine	no	44.1%
				P365 R152-L2Sq	Resnet(1x365)	L2 Squared	no	38.4%
imagenet[22]	objects	500k	3,000	INET Rv2-L2Sq	ResnetV2(1x1536)	L2 Squared	no	78.8%
				INET Rv2-Canb	ResnetV2(1x1536)	Canberra	no	75.9%
				INET Rv2-Cheb	ResnetV2(1x1536)	Chebyshev	no	75.9%

thermore, one of the datasets had their queries postprocessed to simulate perturbations in the acquisition step, see *VGGF VGG16-L2Sq + Pert* on Table 1. In evaluating this many search scenarios, our objective was showing that it is possible to develop relevance prediction methods with consistent accuracy by computing features from scores, regardless of the approach used to generate them.

As outlined in Section 2, our work is concerned with $rank\text{-}k$ binary relevance prediction of a query. Thus, for some query q, the output of our system is a sequence of binary labels $\mathbf{P}^q = \{p_1, p_2, ..., p_k\}$ such that $p_i = 1$ if the ith element of the rank is predicted as relevant, and $p_i = 0$ otherwise. In addition, the groundtruth labels $\mathbf{G}^q = \{g_1, g_2, ..., g_k\}$ such that $g_i \in \{0, 1\}$ are available. We have that $g_i = 1$ if the ith element of the rank is relevant to the query, and $g_i = 0$ otherwise. On account of space constraints, we show only results for $k = 10$.

Considering the nature of our output is akin to a binary-classification problem output, we can quantify the number of *True Positives* (TP), *True Negatives* (TN), *False Positives* (FP), and *False Negatives* (FN). For evaluation purposes, we employ the commonly used Normalized Accuracy (*nACC*), which takes into account both TP and TN. With it, we evaluate whether our system correctly predicts the ith position of a rank as either relevant or non-relevant. Below is the definition of nACC:

$$nACC = \frac{\frac{TP}{TP+FN} + \frac{TN}{TN+FP}}{2} \tag{4}$$

Table 2 shows a toy example to clarify this evaluation approach. This example considers that we have five queries, for which we want to predict the first three positions of their rank, that is, we have a $top\text{-}3$ relevance prediction. The columns under $\mathbf{G}$, g_1, g_2, and g_3 depict the groundtruth of the $top\text{-}3$ position of the rank, respectively. Under $\mathbf{P}$, the columns p_1, p_2, and p_3 depict the predicted relevances for the $top\text{-}3$ positions of the rank, respectively. Because we measure the nACC positionally, we compare column g_1 with column p_1, column g_2 with column p_2, and column g_3 with column p_3. This results in three measurements of nACC:

$$\begin{cases} \alpha_1 = nACC(\{1,0,1,1,0\}, \{1,1,1,1,0\}) = 0.750 \\ \alpha_2 = nACC(\{0,0,0,1,0\}, \{1,0,0,1,0\}) = 0.875 \\ \alpha_3 = nACC(\{1,0,1,1,1\}, \{0,0,1,0,0\}) = 0.625 \end{cases} \tag{5}$$

4.2. Results

Figure 1 depicts results for the normalized accuracy evaluation of each of the $top\text{-}10$ positions in the evaluated ranks. For the results reported, we used $a = 20$ and $b = 20$. While adjusting the values of a and b could potentially lead to better results, we do not explore this optimization in this paper. For simplification, we omit the subscript, referring to the features as Δ, $sDCT$, and FS in the discussion below. The curves show a consistent behavior for all positions of the rank, except for the first one, which sees an spike close to 1.0 normalized accuracy. Overall, the first position of the rank is the easiest to predict, on account of the large score difference between it and other scores of the rank when the object in that position is a match. Thus, it is expected that features which take into account relationships among scores obtain better results at predicting the relevance of this particular position of the rank.

Table 2. Example of a $top\text{-}3$ prediction with five sample queries

Queries	G			P		
	g_1	g_2	g_3	p_1	p_2	p_3
Query 01	1	0	1	1	1	0
Query 02	0	0	0	1	0	0
Query 03	1	0	1	1	0	1
Query 04	1	1	1	1	1	0
Query 05	0	0	1	0	0	0

Another trend observed in the results is that, besides the first position of the rank, prediction among positions two to ten tend to be consistent, without large accuracy spikes. This fact hints towards the structure of the rank being somewhat indistinguishable between the different positions, and thus a method that classifies well enough position i of the rank should also work well enough for any other position j, at least for the nine setups tested herein.

Between the different three different datasets tested, there seems to be a tendency toward better classification in datasets were ranking performed at least reasonably well. It is likely that our methods struggle more to predict non-relevant entries.

Considering the variations within the Places365 dataset, changing either the distance metric or the descriptor employed had little impact on the measured accuracy, mostly impacting the prediction of the first position. In the Vggfaces dataset, adding query perturbations mostly impacted the Δ feature, decreasing its accuracy across all positions of the rank. It also had a small impact on the prediction of the $top\text{-}1$ result. With the Imagenet results, we observe minor impact the metric space used has on the prediction accuracy.

The three features employed had similar results, with the FS feature showing the most consistent results among the three options, although by a small margin. Since both Δ and $sDCT$ features are fast to compute, combining both to generate the FS feature has little impact on the efficiency of the method and should be the preferable approach.

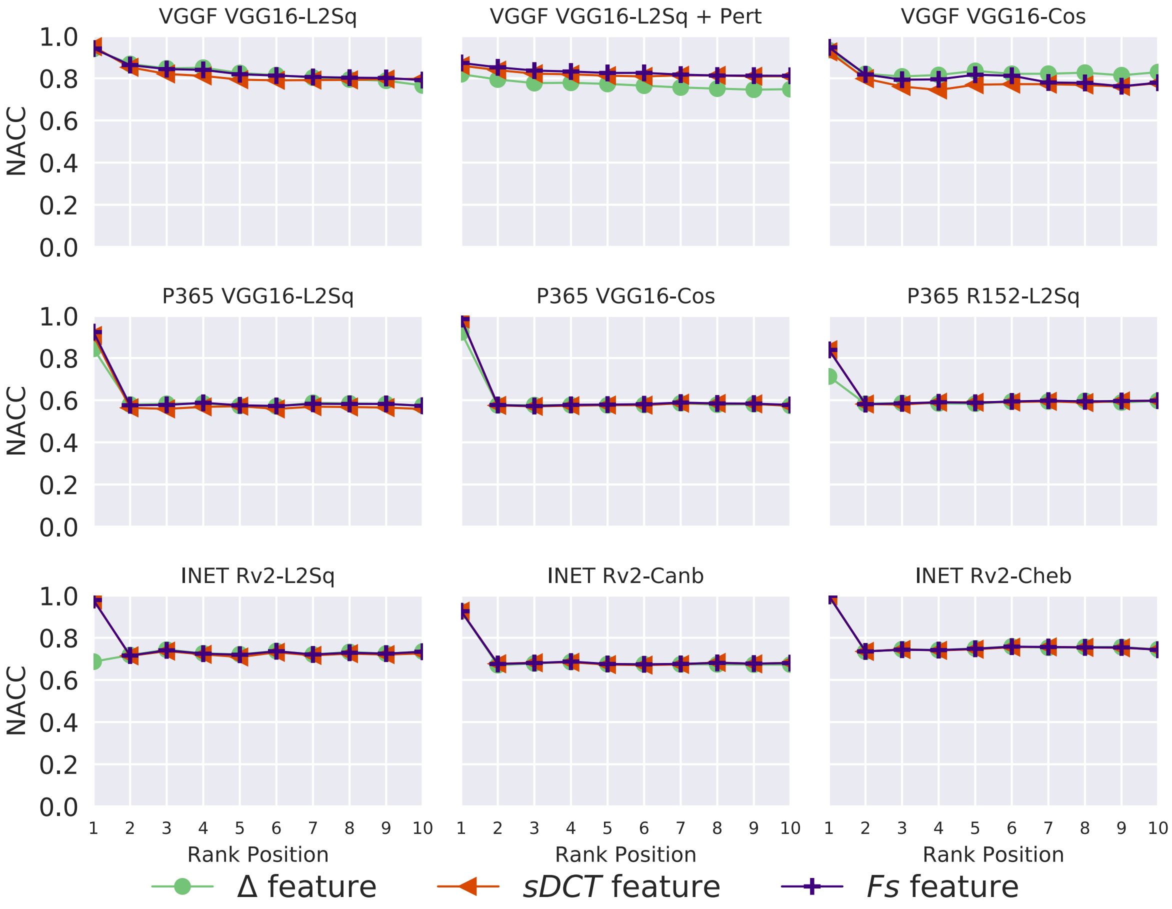

Fig. 1. Normalized accuracy (nACC) curves for the three types of score-based features. Nine experiments are depicted in this figure, covering the scenarios of Table 1. At $x = i$, each curve shows the nACC at predicting the relevance of the ith element of the rank, using each of the proposed features. For all features we have $a = 20$ and $b = 20$.

Our results show the viability of using score-based features to predict the relevance of results from a similarity search engine. Furthermore, there are a few points such as the small amount of training samples and imbalance between relevant and non-relevant samples that could be further improved in order to achieve even better predictions.

5. CONCLUSIONS

This work presented a classification framework for Relevance Prediction in retrieval systems relying on similarity searches. Closely related to the query performance prediction problem, relevance prediction is concerned with predicting the relevance (or lack thereof) of the *top-k* ranked results from a similarity query. Our proposal is to employ k classifiers, such that the ith one is used to predict the ith position of the rank, in conjunction to features extracted from relationships between the scores in the rank. Our features were designed to be fast to compute, and feasible to apply at query time, and utilize score differences or transformations on sequential scores. To evaluate our proposal, we devised nine different testing scenarios, spanning three different image retrieval datasets, focusing on a wide range of methods to generate similarity scores, since those are the core of our method.

Our results in all scenarios show that the proposed classification approach achieves good and consistent results in predicting the relevance of the *top-10* results from similarity searches. The first position of the rank is particularly easy to predict, since its score, when the top ranked object is relevant, usually differs greatly from the remaining entries of the rank. However, our results in the remaining positions are also good, and consistent. Between the different features proposed, or their fusion into a single feature, the latter option obtained the best results in most scenarios, although without a large difference from the other features. We have also observed that the relevance prediction is likely related to the retrieval performance of the tested scenario. As future work, we aim at creating training sets more balanced with non-relevant examples since, for some positions, there is a large imbalance. Moreover, we intend to expand our set of features to contain more score-based features, as well as features based on rank structure. Exploring alternative classification frameworks, such as a single classifier for all *top-k*, is another suitable way to further extend our current method.

6. REFERENCES

[1] Pushpa B Patil and Manesh B Kokare, “Relevance feedback in content based image retrieval: A review,” *Journal of Applied Computer Science & Mathematics*, , no. 10, 2011.

[2] Tao Mei, Yong Rui, Shipeng Li, and Qi Tian, “Multimedia search reranking: A literature survey,” *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 38, 2014.

[3] Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft, “Predicting query performance,” in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2002, pp. 299–306.

[4] Yun Zhou and W Bruce Croft, “Query performance prediction in web search environments,” in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2007, pp. 543–550.

[5] Anna Shtok, Oren Kurland, David Carmel, Fiana Raiber, and Gad Markovits, “Predicting query performance by query-drift estimation,” *ACM Transactions on Information Systems (TOIS)*, vol. 30, no. 2, pp. 11, 2012.

[6] Ronan Cummins, “On the inference of average precision from score distributions,” in *ACM International Conference on Information and Knowledge Management*. ACM, 2012, pp. 2435–2438.

[7] Anna Shtok, Oren Kurland, and David Carmel, “Query performance prediction using reference lists,” *ACM Transactions on Information Systems (TOIS)*, vol. 34, no. 4, pp. 19, 2016.

[8] Haggai Roitman, Shai Erera, and Bar Weiner, “Robust standard deviation estimation for query performance prediction,” in *ACM SIGIR International Conference on Theory of Information Retrieval*. ACM, 2017, pp. 245–248.

[9] Haggai Roitman, Shai Erera, Oren Sar-Shalom, and Bar Weiner, “Enhanced mean retrieval score estimation for query performance prediction,” in *ACM SIGIR International Conference on Theory of Information Retrieval*. ACM, 2017, pp. 35–42.

[10] Stefano Mizzaro, Josiane Mothe, Kevin Roitero, and Md Zia Ullah, “Query performance prediction and effectiveness evaluation without relevance judgments: Two sides of the same coin,” in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2018, pp. 1233–1236.

[11] Shaoyan Sun, Wengang Zhou, Qi Tian, Ming Yang, and Houqiang Li, “Assessing image retrieval quality at the first glance,” *IEEE Transactions on Image Processing*, 2018.

[12] Hamed Zamani, W. Bruce Croft, and J. Shane Culpepper, “Neural query performance prediction using weak supervision from multiple signals,” in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2018, pp. 105–114, ACM.

[13] Zhongmin Zhang, Jiawei Chen, and Shengli Wu, “Query performance prediction and classification for information search systems,” in *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. Springer, 2018, pp. 277–285.

[14] Walter J Scheirer, Anderson Rocha, Ross J Micheals, and Terrance E Boult, “Meta-recognition: The theory and practice of recognition score analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1689–1695, 2011.

[15] Walter J Scheirer, Anderson de Rezende Rocha, Jonathan Parris, and Terrance E Boult, “Learning for meta-recognition,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1214–1224, 2012.

[16] Corinna Cortes and Vladimir Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[17] Jean-Philippe Vert, Koji Tsuda, and Bernhard Schölkopf, “A primer on kernel methods,” *Kernel methods in computational biology*, vol. 47, pp. 35–70, 2004.

[18] Nasir Ahmed, T₋ Natarajan, and Kamisetty R Rao, “Discrete cosine transform,” *IEEE transactions on Computers*, vol. 100, no. 1, pp. 90–93, 1974.

[19] Terry Riopka and Terrance Boult, “Classification enhancement via biometric pattern perturbation,” in *International Conference on Audio-and Video-Based Biometric Person Authentication*. Springer, 2005, pp. 850–859.

[20] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba, “Places: A 10 million image database for scene recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[21] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al., “Deep face recognition.,” in *BMVC*, 2015, number 3 in 1, p. 6.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.