Deep Correlation for Matching Images and Text

Fei Yan Krystian Mikolajczyk
Centre for Vision, Speech and Signal Processing, University of Surrey
Guildford, Surrey, United Kingdom. GU2 7XH
{f.yan, k.mikolajczyk}@surrey.ac.uk

Abstract

This paper addresses the problem of matching images and captions in a joint latent space learnt with deep canonical correlation analysis (DCCA). The image and caption data are represented by the outputs of the vision and text based deep neural networks. The high dimensionality of the features presents a great challenge in terms of memory and speed complexity when used in DCCA framework. We address these problems by a GPU implementation and propose methods to deal with overfitting. This makes it possible to evaluate DCCA approach on popular caption-image matching benchmarks. We compare our approach to other recently proposed techniques and present state of the art results on three datasets.

1. Introduction

Automatically describing (resp. synthesising) visual data using (resp. from) natural language [5, 6, 11, 15, 16, 23, 25, 26, 29, 34, 35, 36, 42, 52] is one of the ultimate goals of computer vision (CV), natural language processing (NLP) and machine learning (ML). However, besides being extremely ambitious, this goal is also practically problematic due to the lack of means for quantifying progress with objective evaluation.

With the image-text parallel corpora that have become available recently [2, 14, 18, 31, 41, 53], the evaluation issue has been alleviated by changing the goal of description and synthesis to a cross-modal retrieval one. In such a scenario, given an image, the goal is to retrieve the gold textual description, and vice versa. Following this setting, various techniques have been proposed recently to learn a latent joint space for image and text. The majority of these techniques optimise either the canonical correlation objective [13, 18] or a structured objective [48] over typically shallowly learnt features, i.e., the features and the objective are decoupled. In contrast, the very recent work of [21] integrates into the deep learning [24, 28] framework an objective that maximises the alignment between fragments of the image and those of the text.

In this paper, we propose an alternative end-to-end learning scheme based on the deep canonical correlation analysis (DCCA) [1]. Our contributions can be summarised as follows:

First, we make non-trivial extensions to [1]. [1] evaluates DCCA on medium-sized problems with low feature dimensionalities, allowing training in a full batch mode using the L-BFGS method. Our feature dimensionalities are two orders of magnitude higher, which can better represent the data but limits the training to small batches and imposes overfitting problems. We propose specific steps to address these issues. The higher dimensionalities also make the singular value decomposition (SVD) required by DCCA much more computationally intensive. To address this problem we propose and discuss details of a GPU implementation with CULA libraries. The efficiency of our implementation is several orders of magnitude higher than CPU implementations.

Secondly, we advance the state of the art on the widely used benchmarks for image-text matching. The performance of the proposed learning scheme outperforms, or is on par with prior art[13, 21]. Our results show that canonical correlation is a very competitive objective, not only for shallowly learnt features, but also in the context of deep learning.

The remainder of this paper is organised as follows. In Section 2 we briefly review related work in the literature. In Section 3 the proposed end-to-end learning scheme is presented, where we discuss the overall architecture, the trace norm objective, as well as how to address complexity and overfitting issues. We then provide experimental evaluation on three benchmarks in Section 4. Finally, Section 5 concludes the paper.

2. Related Work

In this section, we review the existing work on sentence generation for visual data, and its proxy task of matching images and text, followed by a brief introduction to deep CCA.

978-1-4673-6964-0/15/$31.00 ©2015 IEEE

2.1. Describing visual data with natural language

Generating natural language description for image and video has become a popular research topic in recent years. Most techniques [11, 16, 25, 26, 29, 35, 52] adopt a content selection and surface realisation approach. Starting from the output of visual processing engines e.g. object classifiers, object detectors and attribute classifiers, image content that is being described is selected in the form of tuples such as subject-action-object triplets, object-preposition-object triplets, and object-action-preposition-scene triplets quadruplet. A surface realiser is then employed to produce captions as constrained by the lexicon and grammar. While [29] focuses on the investigation of surface realisation techniques, the work in [11, 16, 25, 26, 35, 52] differs primarily in the way the tuples of image content are generated.

Recently, maximum entropy based [10] and tree based [27] language models have been proposed for caption generation. More remarkably, end-to-end learning systems using long short term memory (LSTM) [17] and other recurrent neural networks (RNNs) have enjoyed great success and are becoming popular [4, 7, 20, 22, 32, 49, 50].

In parallel to image captioning, automatic video description is also receiving increasing attention [6, 15, 23, 42, 43, 47]. These techniques operate within the same paradigm of content selection and surface realisation. Compared to image description, typically video description systems additionally employ spatio-temporal methods for action recognition.

2.2. Matching images and text

The main issue with description generation is the lack of automatic and objective evaluation metric. Automatic metrics such as BLEU [38] and ROUGE [30] are useful for measuring the fluency of the generated text [40], but not reliable for how accurately a caption describes an image or a video [18]. METEOR is more correlated with human judgements than BLEU and ROUGE but there is still a large gap [9]. On the other hand, human judgements are quite subjective and are expensive and time-consuming to collect.

The evaluation issue is alleviated by the ranking based formulation of the problem [13, 18, 21, 48, 51]. Assuming pairs of image and human-written description (caption) are available, a joint latent space is learnt, and the performance is evaluated essentially by how high the gold description is ranked among other candidates for a given image (i.e. the image annotation setting), and vice versa (i.e. the image retrieval setting). Existing techniques differ mainly in the way the latent space is learnt.

Canonical correlation analysis (CCA) and its kernel version (KCCA) maximise the correlation in the latent space. While [18] directly employs KCCA for matching images and captions, [13] builds two layers of CCA. The first layer transfers information from a large extra dataset with 1 million image-caption pairs, and the final latent space is learnt in the second layer of CCA.

In contrast to the correlation objective in CCA and KCCA, [48] maximises the margin between matched image-text pairs and non-matched ones, in the structured SVM (S-SVM) setting [45]. As a result, the learnt latent space is asymmetric: two spaces are learnt separately for matching images to text and matching text to images. Moreover, the S-SVM is computationally much more expensive to solve than CCA/KCCA, both speed-wise and memory-wise.

In [21], image and sentence fragments are extracted using object detectors and dependency tree relations, respectively. The objective optimised encodes the intuition that fragments in the image and those in the text should be aligned. [21] produces state of the art performance on popular benchmarks for image-text matching. Another two similar pieces of work deeply embedding visual and textual features are [12, 44], whose objective functions can be broadly seen as special cases of that in [21].

2.3. Deep canonical correlation analysis

In contrast to hand-crafted objectives, deep CCA (DCCA) [1] optimises the CCA objective in the deep learning framework. It uses the insight that the total correlation sought in CCA can be maximised by optimising a matrix trace norm, and the gradient of the trace norm with respect to features of the two modalities can be computed. This allows propagating the gradient down in a deep neural network, achieving end-to-end learning.

In [1] DCCA is evaluated on medium-sized problems and in terms of total correlation obtained in the learnt latent space, which as noted in the paper is not the final goal of real-world applications. Before DCCA can be applied to problems whose features are two orders of magnitude higher and achieve improved matching performance, various issues such as time complexity, memory complexity, and overfitting must to be addressed.

3. Deep Correlation for Matching Images and Text

In this section, we show how we address the issues in DCCA in order to produce state of the art performance on the task of matching images and text. We first present the overall architecture of the proposed network, followed by an introduction to the trace norm objective. We then discuss how the complexity and overfitting issues are addressed.

3.1. Architecture of the network

The architecture of the proposed network is illustrated in Figure 1. The image branch (top row in Figure 1) of the net-

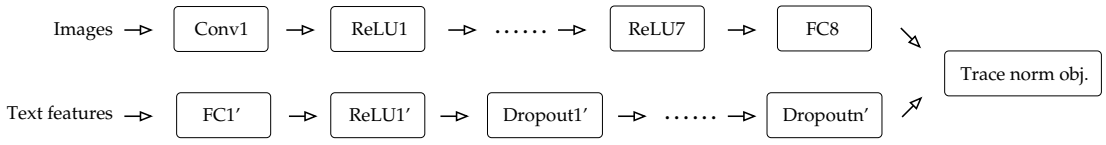


Figure 1. Architecture of the proposed network.

work is essentially the deep convolutional neural network (CNN) proposed in [24]. This network and its variants have achieved immense success in a wide range of vision problems [8, 39]. The text branch (bottom row in Figure 1) of the network consists of n stacked triplets of fully connected (FC) layer, rectified linear unit (ReLU) layer, and dropout layer.

As will be detailed later, the total correlation obtained in CCA is equal to a matrix trace norm. In the trace norm objective (TNO) layer, the gradient of the trace norm with respect to X and Y are computed and propagated backward, where $X \in \mathbb{R}^{d_x \times m}$ is the output of the FC8 layer and $Y \in \mathbb{R}^{d_y \times m}$ is that of the Dropoutn′ layer, d_x and d_y are the dimensionalities, and m is the batch size.

We employ simple term frequency-inverse document frequency (TF-IDF) representation to build text features that are fed into the FC1′ layer. First, all captions are tokenised and lemmatised using the linguistic analyser of [37]. We keep the top d_y most frequent lemmatised words and build a d_y dimensional TF-IDF histogram $\mathbf{t}$ for each caption, with the i^{th} dimension:

$$t_i = a_i \log \frac{B}{b_i + 1} \tag{1}$$

where a_i is the term frequency of the i^{th} lemmatised word i.e. the number of its occurrences in the caption, b_i is the document frequency of the lemmatised word i.e. the number of training captions where it appears, and B is the total number of training captions. The FC layers in the text part of the network FC1′, ..., FCn′ each have d_y neurons. As a result, the input features Y to the TNO layer has a dimensionality of d_y.

3.2. Trace norm objective

Given two sets of m random vectors $X \in \mathbb{R}^{d_x \times m}$ and $Y \in \mathbb{R}^{d_y \times m}$, let their covariances be Σ_{xx} and Σ_{yy} respectively, and let the cross covariance be Σ_{xy}. Canonical correlation analysis (CCA) seeks pairs of linear projections that maximise the correlation of the two views:

$$\begin{aligned}(\mathbf{w}_x^*, \mathbf{w}_y^*) &= \underset{\mathbf{w}_x, \mathbf{w}_y}{\text{argmax}}\ \text{corr}(\mathbf{w}_x^T X, \mathbf{w}_y^T Y) \\ &= \underset{\mathbf{w}_x, \mathbf{w}_y}{\text{argmax}} \frac{\mathbf{w}_x^T \Sigma_{xy} \mathbf{w}_y}{\sqrt{\mathbf{w}_x^T \Sigma_{xx} \mathbf{w}_x \mathbf{w}_y^T \Sigma_{yy} \mathbf{w}_y}}\end{aligned} \tag{2}$$

Since the objective is invariant to scaling of $\mathbf{w}_x$ and $\mathbf{w}_y$, the projections are constrained to have unit variance:

$$(\mathbf{w}_x^*, \mathbf{w}_y^*) = \underset{\mathbf{w}_x^T \Sigma_{xx} \mathbf{w}_x = \mathbf{w}_y^T \Sigma_{yy} \mathbf{w}_y = 1}{\text{argmax}} \mathbf{w}_x^T \Sigma_{xy} \mathbf{w}_y \tag{3}$$

Assembling the top projection vectors into the columns of projection matrices W_x and W_y, the CCA objective can be written as:

$$\begin{aligned}&\max_{W_x, W_y}\ \text{tr}(W_x^T \Sigma_{xy} W_y) \\ &\text{s.t. : } W_x^T \Sigma_{xx} W_x = W_y^T \Sigma_{yy} W_y = I\end{aligned} \tag{4}$$

Let $\bar{X}$ and $\bar{Y}$ be the centred data matrices respectively:

$$\bar{X} = X - \frac{1}{m} X \mathbf{1}, \quad \bar{Y} = Y - \frac{1}{m} Y \mathbf{1} \tag{5}$$

In practice, the covariance matrices are estimated as:

$$\begin{aligned}\Sigma_{xx} = \tfrac{1}{m-1} \bar{X} \bar{X}^T + \lambda_x I, \quad \Sigma_{yy} = \tfrac{1}{m-1} \bar{Y} \bar{Y}^T + \lambda_y I \\ \Sigma_{xy} = \tfrac{1}{m-1} \bar{X} \bar{Y}^T\end{aligned} \tag{6}$$

where $\lambda_x I$ and $\lambda_y I$ are regularisers to ensure the positive definiteness of Σ_{xx} and Σ_{yy}.

Define $T = \Sigma_{xx}^{-1/2} \Sigma_{xy} \Sigma_{yy}^{-1/2}$, and let U_k and V_k be the matrices of the first k left- and right- singular vectors of T respectively. It is shown in [33] that the optimal objective value is the sum of the top k singular values of T, and the optimum is attained at

$$(W_x^*, W_y^*) = (\Sigma_{xx}^{-1/2} U_k, \Sigma_{yy}^{-1/2} V_k) \tag{7}$$

When $k = d_x = d_y$, the total correlation objective in Eq. (4) is equal to the trace norm of T:

$$\text{corr}(X, Y) = ||T||_{\text{tr}} = \text{tr}((T^T T)^{1/2}) \tag{8}$$

Moreover, let the singular value decomposition (SVD) of T be $T = UDV^T$, it is shown in [1] that the gradient of the total correlation with respect to X is given by:

$$\frac{\partial \text{corr}(X, Y)}{\partial X} = \frac{1}{m-1} (2 \nabla_{xx} \bar{X} + \nabla_{xy} \bar{Y}) \tag{9}$$

where

$$\nabla_{xx} = -\tfrac{1}{2} \Sigma_{xx}^{-1/2} U D U^T \Sigma_{xx}^{-1/2} \tag{10}$$

$$\nabla_{xy} = \Sigma_{xx}^{-1/2} U V^T \Sigma_{yy}^{-1/2} \tag{11}$$

Similarly, the gradient with respect to Y is

$$\frac{\partial \text{corr}(X,Y)}{\partial Y} = \frac{1}{m-1}(2\nabla_{yy}\bar{Y} + \nabla_{yx}\bar{X}) \quad (12)$$

where

$$\nabla_{yy} = -\tfrac{1}{2}\Sigma_{yy}^{-1/2} V D V^T \Sigma_{yy}^{-1/2} \quad (13)$$

$$\nabla_{yx} = \Sigma_{yy}^{-1/2} V U^T \Sigma_{xx}^{-1/2} \quad (14)$$

The gradients are computed in the TNO layer of Figure 1, and propagated down along the two branches of the network.

3.3. GPU implementation

Assuming for now $d_x = d_y = d$, the most computationally expensive operation for computing the gradients in Eq. (9) and Eq. (12) is the SVD of the $d \times d$ matrix T. [1] considers cases where d is in the order of 10^1 and implements the TNO layer on a CPU. For the application of image-text matching, however, it is observed that the number of features required to encode the rich information in image and text is in the order of 10^3 [8, 13, 18, 39].

To make DCCA practically applicable to our application, we implement the TNO layer on a GPU with the CUBLAS[1] and CULA[2] libraries. Both libraries are based on CUDA[3]. While CUBLAS is a GPU-accelerated version of the complete standard BLAS[4] library, which provides basic linear algebra subroutines, CULA can be broadly seen as the GPU version of the LAPACK[5] library, and provides more sophisticated linear algebra routines such as solving systems of simultaneous linear equations, eigenvalue problems, and singular value problems.

In our GPU based implementation of the TNO layer, the SVD of matrix T is solved using the CULA library. For comparison we also implement the layer using several CPU based linear algebra libraries. Figure 2 compares the time needed to solve an SVD with various libraries, where OpenCV[6], LAPACK[7], Eigen[8] are CPU based. It is clear from Figure 2 that when d is in the order of 10^3, CULA is typically two to three orders of magnitude faster. For example, when $d = 4096$, CULA takes only 16.3 seconds, while LAPACK, OpenCV and Eigen take 4922.6, 6714.5 and 22971.1 seconds, respectively.

Other linear operations such as matrix multiplication and the Cholesky decomposition required for matrix inversion also get a significant speedup with CULA and CUBLAS.

[1] https://developer.nvidia.com/cublas
[2] http://www.culatools.com/
[3] http://www.nvidia.com/object/cuda_home_new.html
[4] http://www.netlib.org/blas/
[5] http://www.netlib.org/lapack/
[6] http://opencv.org
[7] http://www.netlib.org/lapack
[8] http://eigen.tuxfamily.org

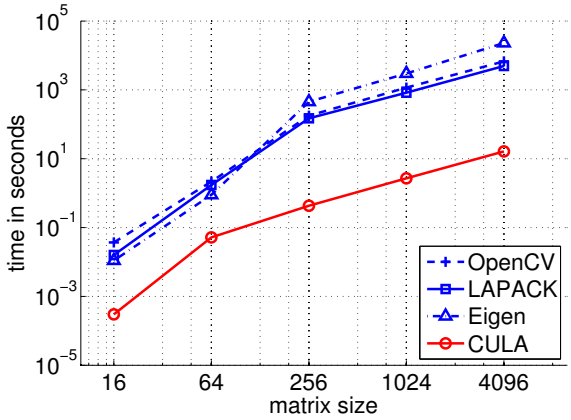


Figure 2. Log-log plot of speed of SVD solvers. The GPU based CULA solver is two to three orders of magnitude faster than CPU based ones when matrix is 4096×4096.

Overall each iteration for a batch of size $m = 100$ takes approximately 26.5 seconds to complete. Since typically thousands of iterations are needed for the network to converge, it is clear that the migration from CPU to GPU is a crucial step for DCCA to be practically employed in our problem.

Both CPU and GPU versions of the TNO layer are implemented under the deep learning framework Caffe [19]. Time reported in this section is measures on a single core of an AMD Opteron 6262 HE CPU and an Nvidia Tesla M2090 GPU using single precision.

3.4. Addressing overfitting

The GPU implementation brings huge efficiency improvements but at the same time imposes a limit on the batch size. The CCA correlation loss is a batch objective, that is, it can not be computed by assembling losses of individual points, and is sensitive to the batch size. It is observed in [1] that training in full batch mode produces much better results than in small batches. However, a GPU with 6GB of memory limits the batch size m to 100 when d is set to 4096. This is only a fraction of the training data for datasets we consider, and as a result adds to the already existing issue of overfitting in deep neural networks.

To address the overfitting issue, dropout layers $\text{ReLU1}', ..., \text{ReLUn}'$ are inserted in the network, and the training data is augmented with mirrored versions of the images. In addition, the training data is also augmented with 10 copies of itself and is shuffled. Although the copies are identical to the original data, due to the batch nature of the CCA objective, they still provide new batches for the network to learn from.

It is also crucially important to have a good initialisation for the millions of parameters in the network of Figure 1. For the image part of the network, we initialise by

- A girl in a white dress runs down a country road.
- a girl walks on a dirt street.
- A little girl in a white dress and no shoes walks down a dirt road in America.
- A woman in a white dress is walking along a long straight road.
- A young girl with a white dress walks down a dirt road with trees and fields on both sides of the road.

- A woman wearing a white shirt, tan pants, and boots is standing beside a vendor cart that contains various beverages.
- Lady at the beach standing next to her ice cream cart, numerous people is lounging in the background.
- A treat vendor is standing in the sun while others sit in the shade under umbrellas.
- A black woman wearing a blue hat and white t-shirt is standing at a snow cone stand.
- Woman with a vending cart in the middle of a beach.

- the ruins of a city with many green areas that was built on several terraces; behind it a very significant, rugged mountain; slight waft of mist; there is a wooded mountain range in the background;

Table 1. Example image-caption pairs. Top: Flickr8K; middle: Flickr30K; bottom: IAPR TC-12. IAPR TC-12 captions tend to be more detailed than those of Flickr datasets. The example images have been resized to 256 by 256.

pre-training the AlexNet model [19, 24] and transferring the learnt weights. For the text part of the network, we initialise the weights of the fully connected layers (FC1$'$, ..., FCn$'$) to identity matrices, and the biases to zeros. This ensures that the search for optimal parameters always starts from the "safe" point of the TF-IDF features defined in Eq. (1).

4. Experiments

In this section we evaluate the DCCA learning scheme on three image-text parallel datasets, namely Flickr8K, Flickr30K, and IAPR TC-12. On each dataset we compare the performance of our scheme with the state of the art reported, following the experimental protocols and evaluation metrics used. On all datasets, the batch size is set to $m = 100$ and the dimensionalities of the input of both modalities to the TNO layer are set to $d = 4096$.

4.1. Flickr8K

The Flickr8K dataset [18] consists of 8000 images from the Flickr.com website, which focus on people or animal performing actions. Using a crowdsourcing service, five captions were generated by different annotators for each image. The annotators were asked to describe the actors, objects, scenes and activities that were shown in the image, i.e., information that could be obtained from the image alone. An example image-caption pair is shown in the top row of Table 1.

The dataset is split into predefined training, validation, and test sets with 6000, 1000, and 1000 pairs respectively. In [18] the five captions are pooled into one for the training set, and for the validation and test sets only caption two is used. We dub this setting protocol I. In contrast the protocol in [21] (protocol II) keeps all five captions for the test set. For the task of image annotation i.e. image-to-text retrieval, only the highest ranked caption among the five ground truth captions is considered.

In addition to these two existing protocols, we also introduce protocol III, where we pool the five captions into one for train, validation and test sets. The text representation for the validation and test sets in protocol III is richer and matches the training set better, it is therefore easier than protocols I and II.

For each test image the 1000 captions in the test set are ranked according to their cosine similarity to the image in the learnt latent space. This ranked list allows to define automatic and objective metrics that measure how well images and captions are matched. Moreover, such a framework can be trivially extended to perform the symmetric task of image retrieval using captions. We follow the common practice on this dataset and report in Table 2 the average recall of the gold item at position 1, 5, 10 of the ranked list (R@1, R@5, R@10), and the median rank (MR) of the gold item, for both image annotation and image retrieval tasks. Note that in contrast to the recalls, a lower median rank indicates a better performance.

Table 2 shows that under both protocols I and II the proposed scheme outperforms competing methods on most metrics. The only exception is that under protocol I, the

		Image annotation				Image retrieval			
		R@1	R@5	R@10	MR	R@1	R@5	R@10	MR
Protocol I	KCCA [18]	8.3	21.6	30.3	34	7.6	20.7	30.1	38
	transfer CCA [13]			**48.8**					
	Deep Fragment [21]	9.3	24.9	37.4	21	8.8	27.9	41.3	17
	DCCA	**13.6**	**32.9**	46.4	**13**	**12.1**	**31.6**	**44.8**	**14**
Protocol II	DeViSE [12]	4.8	16.5	27.3	28	5.9	20.1	29.6	29
	SDT-RNN [44]	6.0	22.7	34.0	23	6.6	21.6	31.7	25
	Deep Fragment [21]	12.6	32.9	44.0	14	9.7	29.6	42.5	15
	DCCA	**17.9**	**40.3**	**51.9**	**9**	**12.7**	**31.2**	**44.1**	**13**
Protocol III	DCCA	28.2	56.1	69.8	4	26.3	54.0	67.5	5

Table 2. Performance on Flickr8K.

		Image annotation				Image retrieval			
		R@1	R@5	R@10	MR	R@1	R@5	R@10	MR
Protocol I	transfer CCA [13]			**32.8** / 32.4					
	DCCA			32.5					
Protocol II	DeViSE [12]	4.5	18.1	29.2	26	6.7	21.9	32.7	25
	SDT-RNN [44]	9.6	29.8	41.1	16	8.9	29.8	41.1	16
	Deep Fragment [21]	16.4	**40.2**	**54.7**	**8**	10.3	**31.4**	**44.5**	**13**
	DCCA	**16.7**	39.3	52.9	**8**	**12.6**	31.0	43.0	15
Protocol III	DCCA	27.9	56.9	68.2	4	26.8	52.9	66.9	4

Table 3. Performance on Flickr30K.

transfer CCA of [13] has an edge (48.8 vs. 46.4) on R@10 for image annotation, the only metric reported in the paper. Note however that [13] uses an additional large dataset with 1 million image-caption pairs for training. In Table 2, the performance of DeViSE [12] and SDT-RNN [44] is taken from [21], where the original code is modified to work on the Flickr8K dataset.

4.2. Flickr30K

Flickr30K [53] is an extension of Flicker8K with 31784 images each paired with five captions, and the captions were annotated in a similar style as in Flicker8K. An example image-caption pair can be found in the middle row of Table 1.

As for the case of Flicker8K, two evaluation protocols exist in the literature. Protocol I [13] uses 25000 pairs for training, 3000 for validation and 3000 for test. It pools the five captions into one only for the training set, and uses only one caption for validation and test sets. Protocol II [12, 21, 44] adopts a 28000/1000/1000 split for the sets. It keeps five captions separately for the test set, and evaluate image annotation in a similar fashion as for Flickr8K dataset. We also introduce protocol III, where we use a 28000/1000/1000 split and pool five captions into one for all three sets. As in Flickr8K, average recalls R@1, R@5, R@10 and median rank MR are used as evaluation metrics for both tasks of image annotation and image retrieval.

The results in Table 3 indicate that under protocol I the transfer CCA in [13] achieves an R@10 score of 32.8 and 32.4 when using Flickr1M and SBU1M as additional training data respectively, where Flickr1M and SBU1M each contain 1 million image-caption pairs. Our method has an R@10 score of 32.5, which is on par with [13] but does not use additional data for training. Note that in [13] only the R@10 score for the image annotation task is reported.

On the other hand, when no extra data is used for training, the performance of the proposed learning scheme is comparable to that of [21], which is the state of the art on this dataset under protocol II. Essentially, the two approaches adopt different philosophies for matching images and text. [21] breaks an image into objects and a sentence into dependency tree relations, and maximises the explicit alignment between the image fragments and text fragments. In contrast, our TF-IDF based text features and CNN based visual features capture global properties of the two modalities respectively. The alignment of the fragments in image and text is implicitly taken care of by the CCA correlation objective.

4.3. IAPR TC-12

The IAPR TC-12 benchmark [14] consists of 20000 still natural images taken from locations around the world and

	Image annotation			Image retrieval		
	P@1	P@5	MAP	P@1	P@5	MAP
structured SVM [48]	0.086	0.070	0.050	0.035	0.029	0.035
DCCA	**0.302**	**0.114**	**0.426**	**0.295**	**0.120**	**0.415**

Table 4. Performance on IAPR TC-12.

comprising an assorted cross-section of still natural images. Each image is associated with a text caption in up to three different languages (English, German and Spanish). In this paper we consider only the pairing of the images and the English captions. An example pair is shown in the bottom row of Table 1. Compared to Flickr8K and Flickr30K, there is only one caption for each image, but the captions tend to be more detailed. The average length of the captions is 28.2 words, as opposed to 12.9 and 14.4 words in Flickr8K and Flickr30K respectively.

To our knowledge [48] is the only work that uses the IAPR TC-12 dataset for image-text matching. Following the evaluation protocol and metrics in [48], we split the dataset into a training set of 18000 pairs and a test set of 2000 pairs, and report in Table 4 precision at position 1, 5 of the ranked list (P@1, P@5) and the mean average precision (MAP). Note that P@k and average precision (AP) are closely related to R@k and median rank metrics used for Flickr8K and Flickr30K datasets. More specifically, $\text{P@k} = \text{R@k}/k$, and $\text{AP} = 1/\text{Rank}$.

The performance of [48] and the proposed learning scheme is shown in Table 4. Table 4 demonstrates the advantage of our method, improving the scores by up to an order of magnitude. In the table the performance of structured SVM [48] is read from Figure 4 of the paper, as numerical values are not provided.

[48] employs the latent Dirichlet allocation (LDA) [3] to build image and text representations. The structured SVM in [48] constructs a joint space of outer product, and is computationally expensive both in term of time and memory. For instance, when the dimensionality d of both modalities is 1000, approximately 100GB of memory are required, and training can take several days for certain values of the regularisation parameter C[9]. As a result, the dimensionality i.e. the number of topics learnt in LDA is limited, and is set to 100 in their experiments. The low dimensional representations may not be expressive enough, leading to the suboptimal performance of [48].

In contrast, our scheme benefits from the efficient GPU implementation, and allows to use higher dimensional representations. As a result, the rich information in image and text is captured.

[9]Code is available at http://researchweb.iiit.ac.in/ yashaswi.verma/

4.4. Discussions

Overall, on all of the three widely used benchmarks for image-text matching, the proposed learning scheme has exhibited state of the art performance. This confirms that canonical correlation is a powerful objective not only for shallowly learnt features, but also in the context of deep learning. On the other hand, however, it is not yet known how to employ canonical correlation for text generation rather than image-text matching. The very recent work [4, 7, 20, 22, 32, 49, 50] that uses LSTM and other variants of RNN for text generation is advantageous in this respect. Recently there has also been progress on objective metric for image description evaluation [46].

In Table 5 and Table 6 qualitative results for three random test examples in the Flickr8K dataset are shown, with the five top ranked and the gold captions/image for each query image/caption.

5. Conclusions

In this paper we proposed an image-caption matching approach based on deep canonical correlation analysis (DCCA). We have made DCCA applicable to high dimensional image and text representations and large datasets by resolving non-trivial complexity and overfitting issues. We have demonstrated the achieved speedup of several orders of magnitude and compared our approach with competing techniques on standard benchmarks for image-text matching. The performance of the proposed learning scheme outperforms, or is on par with prior art.

Acknowledgements

This work has been supported by EU Chist-Era EPSRC EP/K01904X/1 Visual Sense project.

References

[1] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *ICML*, 2013.

[2] T. Berg, A. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web images. In *ECCV*, 2010.

[3] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *JMLR*, 3(1):234–278, 2003.

[4] X. Chen and C. Zitnick. Learning a recurent visual representation for image caption generation. In *arXiv:1411.5654 [cs.CV]*, 2014.

- A bull stands in a field, next to a red car.
- A black and white dog is catching a Frisbee in the yard.
- A black and white dog catches flying discs thrown by a man.
- A dog is jumping to catch a Frisbee and casts a perfect shadow.
- a black and white dog jumping in the air to catch a Frisbee
- . . .
- **A dog is jumping in the air trying to catch a red Frisbee.**

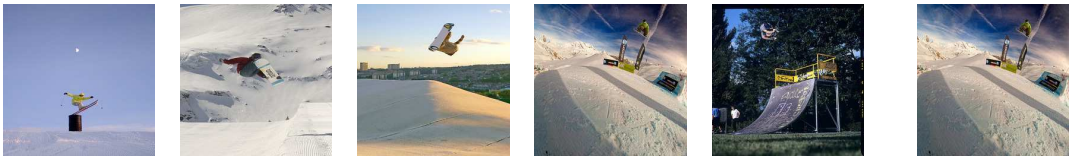
- A boy is making a splash in a swimming pool.
- Two boys are in midair jumping into an inground pool.
- **Killer whales perform for a crowd.**
- A child, held by his mother, slides down a water slide.
- A bunch of dogs and kids playing in a swimming pool.

- A girl twirls in a pink dress.
- A little girl wearing a pink shirt is sitting at the table and drinking a milkshake
- A girl is cold after coming out of the pool and is covered by a towel.
- A girl making a sad face standing in front of her tricycle
- A toddler is making a splash inside a blue paddling pool.
- . . .
- **A young girl is in a teal colored netted skirt and wearing a shirt with a peace sign.**

Table 5. Query image, the five top ranked captions retrieved (from top to bottom), and the gold caption (in boldface). In the three random examples the rank of the gold caption is 30, 3, and 24 respectively. The images have been resized to 256 by 256.

Children participate in a sport on a green field while in uniforms.

A snowboarder in bright green performing a jump at a competition.



A girl in a bikini wears a sign saying ”free hugs”.

Table 6. Query caption, the five top ranked images retrieved (from left to right), and the gold image (in column 6). In the three random examples the rank of the gold image is 30, 4, and 137 respectively. The images have been resized to 256 by 256.

[5] B. Coyne and R. Sproat. Wordseye: An automatic text-to-scene conversion system. In *SIGGRAPH*, 2001.

[6] P. Das, C. Xu, R. Doell, and J. Corso. A thousand frames in just a few words: Lingual description of videos through latent topic and sparse object stitching. In *CVPR*, 2013.

[7] J. Donahue, L. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and de-

scription. In *arXiv:1411:4389 [cs.CV]*, 2014.

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. arXiv:1310.1531 [cs.CV], 2013.

[9] D. Elliott and F. Keller. Comparing automatic evaluation measures for image description. In *ACL*, 2014.

[10] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. Platt, C. Zitnick, and G. Zweig. From captions to visual concepts and back. In *arXiv:1411.4952 [cs.CV]*, 2014.

[11] A. Farhadi, M. Hejrati, M. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences for images. In *ECCV*, 2010.

[12] A. Frome, G. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

[13] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *ECCV*, 2014.

[14] M. Grubinger, P. Clough, H. Muller, and T. Deselaers. The iapr benchark: A new evaluation resource for visual information systems. In *International Conference on Language Resources and Evaluation*, 2006.

[15] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*, 2013.

[16] A. Gupta, Y. Verma, and C. Jawahar. Choosing linguistics over vision to describe images. In *AAAI*, 2012.

[17] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[18] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013.

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093 [cs.CV], 2014.

[20] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. Technical Report, 2014.

[21] A. Karpathy, A. Joulin, and L. Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, 2014.

[22] R. Kiros, R. Salakhutdinov, and R. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *arXiv:1411:2539 [cs.LG]*, 2014.

[23] N. Krishnamoorthy, G. Malkarnenkar, R. Mooney, K. Saenko, and S. Guadarrama. Generating natural language video descriptions using text-mined knowledge. In *AAAI*, 2013.

[24] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012.

[25] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. Berg, and T. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.

[26] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012.

[27] P. Kuznetsova, V. Ordonez, T. Berg, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions. *TACL*, 2:351–362, 2014.

[28] Y. LeCun, B. Boser, J. Denker, D. Henerson, R. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[29] S. Li, G. Kulkarni, T. Berg, A. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *CoNLL*, 2011.

[30] C. Lin. ROUGE: a package for automatic evaluation of summaries. In *Workshop on Text Summarization Branches Out*, 2004.

[31] T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Zitnick, and P. Dollar. Microsoft coco: Common objects in context. In *arXiv:1405.0312 [cs.CV]*, 2014.

[32] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Explain images with multimodal recurrent neural networks. arXiv:1410.1090 [cs.CV], 2014.

[33] K. Mardia, J. Kent, and J. Bibby. *Multivariate Analysis*. Academic Press, 1979.

[34] R. Mason and E. Charniak. Nonparametric method for data-driven image captioning. In *ACL*, 2014.

[35] M. Mitchell, X. Han, J. Dodge, A. Mensch, A. Goyal, A. Berg, K. Yamaguchi, T. Berg, K. Stratos, and H. Daume. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012.

[36] V. Ordonez, G. Kulkarni, and T. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[37] L. Padro and E. Stanivlosky. Freeling 3.0: Towards wider multilinguality. In *Language Resources and Evaluation Conference*, 2012.

[38] K. Papineni, S. Roukos, T. Ward, and W. Zhu. BLEU: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[39] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. arXiv:1403.6382 [cs.CV], 2014.

[40] E. Reiter and A. Belz. An investigation into the validity of some metrics for automatically evaluating natural lanugage generation systems. *Computational Linguistics*, 35(4):529–338, 2009.

[41] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele. A dataset for movie description. In *arXiv:1501.0253 [cs.CV]*, 2015.

[42] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *ICCV*, 2013.

[43] A. Senina, M. Rohrbach, W. Qiu, A. Friedrich, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele. Coherent multi-sentence video description with variable level of detail. In *arXiv:1403.6173 [cs.CV]*, 2014.

[44] R. Socher, A. Karpathy, Q. Le, C. Manning, and A. Ng. Grounded compositional semantics for finding and describing images with sentences. In *ACL*, 2014.

[45] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6:1453–1484, 2005.

[46] R. Vedantam, C. Zitnick, and D. Parikh. Cider - consensus-based image description evaluation. In *arXiv:1411.5726 [cs.CV]*, 2014.

[47] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In *arXiv:1412.4729 [cs.CV]*, 2014.

[48] Y. Verma and C. Jawahar. Im2text and text2im: Associating images and texts for cross-modal retrieval. In *British Machine Vision Conference*, 2014.

[49] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *arXiv:1411:4555 [cs.CV]*, 2014.

[50] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell - neural image caption generation with visual attention. In *arXiv:1502.03044 [cs.LG]*, 2015.

[51] F. Yan and K. Mikolajczyk. Leveraging high level visual information for matching images and captions. In *ACCV*, 2014.

[52] Y. Yang, C. Teo, H. D. III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011.

[53] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014.