Video and Language Description Targeted Person Search

LI, Shuang

A Thesis Submitted in Partial Fulfilment

of the Requirements for the Degree of

Master of Philosophy

in

Electronic Engineering

The Chinese University of Hong Kong

July 2018

Abstract

Person search aims at matching a target identity with a gallery of person images. It has many real-world applications such as cross-camera person tracking, home security system, and autonomous driving. In this thesis, three problems on person search are investigated, including video-based person search, natural language based person search, and identity-aware textual-visual matching.

In the first part of this thesis, we propose a diversity regularized spatiotemporal attention for video-based person search. Instead of averaging full frame features across time where people are often partially occluded, the spatiotemporal attention model learns to detect multiple diverse salient image regions to avoid features from being corrupted by occluded regions. A diversity regularization term is proposed to ensure the spatial attention models don't focus on the same body part.

Video-based person search cannot be applied to some real-world scenarios where only verbal descriptions are available; therefore, the second part addresses the problem of person search using natural language description. We collect a person description dataset and design a Gated Neural Attention model to capture word-image relations. The model consists of a visual sub-network and a language sub-network. The visual sub-network encodes certain human attributes and appearance patterns. The language sub-network is a recurrent neural network. It predicts the importance of each image region to the input word and computes the sentence-image affinity.

One of the problems pertaining to natural language based person search is the difficulty in balancing accuracy and efficiency in a single network. In the third part of this thesis, a two-stage framework is proposed to overcome this problem. The easy incorrect matchings (sentence and person images) are efficiently screened by a Cross-Modal Cross-Entropy loss in the stage-1 framework. The stage-2 framework further verifies hard matchings using a co-attention mechanism. This method can solve general textual-visual matching problems.

摘要

人物檢索通過圖像匹配在數據庫中找到目標人物。人物檢索廣泛應用在實際場景中，例如人物跟蹤，室內安防，以及自動駕駛。本文針對人物檢索的三個問題進行了詳細的分析，包括基於視頻的人物檢索，基於自然語言描述的人物檢索，以及基於個體的文字圖像匹配。

在論文的第一部分，我們提出了一個多樣性約束的時空註意力模型去解決視頻人物檢索。由於行人經常有部分區域被遮擋，時空註意力模型並沒有平均每一幀圖像的特徵作為視頻特徵，它學習去檢測多個顯著區域並分別提取每個區域的特徵從而避免特徵被遮擋的區域污染。一個多樣性約束項被提出來保證多個空間註意力模型檢測到人體的不衕部位。

基於視頻的人物檢索並不能解決全部實際問題，例如當隻有語言描述可以利用時，基於視頻檢索的方法將失效。論文的第二部分提出了基於自然語言描述的人物檢索。我們建立了一個描述人物的數據庫並且設計了一個深度神經網絡去捕捉文字圖像關係。該神經網絡由一個視覺子網絡和一個語言子網絡組成。其中視覺子網絡編碼特定的人物特徵和外觀模式。語言子網絡是一個卷積神經網絡，它通過預測每個圖像區域針對當前輸入文字的重要程度來計算文字圖像的相關性。

基於自然語言描述的人物檢索麵臨如何在單一網絡中平衡準確性和高效性的問題。在論文的第三部分，我們提出了一個雙階段的方法去解決這個問題。在第一階段，我們通過交叉熵損失函數過濾掉簡單的錯誤文字圖像對。第二階段利用聯合註意力模型針對剩下的複雜文字圖像對進行預測。這個方法可以用來解決基本的文字圖像匹配類問題。

Acknowledgments

I am really fortunate to be supervised by Prof. Xiaogang Wang, an absolutely devoted advisor, who gradually cultivates my ability to conduct in-depth research. He is a person with great aptitude, diligence, and persistence, who inspires me to work harder and think deeper. Without his help, I would never have published so many papers.

I was really lucky to work with so many talented people during my M.Phil. period. I would like to thank Hongsheng Li, Wanli Ouyang, Dahua Lin, Change (Cavan) Loy, and Xiaoou Tang. Their insightful suggestions and research attitude encourage me to conduct in-depth reflection for not only my research but also my life.

I must appreciate my colleague Tong Xiao, Wei Yang, Kai Kang, Jing Shao, Xiao Chu, Yantao Shen, Lu Sheng, Rui Zhao, Xingyu Zeng, Shuai Yi, Zhuoyi Zhao, Hongyang Li, Shuai Li, Kun Wang, Xihui Liu, Yixiao Ge, Hang Zhou, Peng Gao, Xiaoyang Guo, Yonglong Tian, Yuanjun Xiong, and Ruihui Wang. Thank all of my colleagues for supporting me so much during my study at The Chinese University of Hong Kong.

I would like to thank Leonid Sigal, Peter Carr, and Slawomir Bak, for offering me the great opportunity to work at Disney Research; Liang Lin, Zhe Lin, Xiaohui Shen, Brian Price, Bolei Zhou, Dayu Yue, Jiayun Wang, Junting Pan and Peter Mathews, for providing valuable suggestions during our collaborations; and Antonio Torralba and Huchuan Lu, for supporting and encouraging me to pursue my career aspiration as a professor.

Last but not least, my deepest gratitude to my parents, who tought me right from wrong and would always be there for me.

Declaration

I hereby declare that this dissertation is composed by myself and all the contents has not been submitted to this or any other universities for a degree. The materials of some chapters have been published in the following conference proceedings or journals.

- **Shuang Li***, Tong Xiao*, Bochao Wang, Liang Lin, and Xiaogang Wang, "Joint Detection and Identification Feature Learning for Person Search." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (***CVPR***), 2017.

- **Shuang Li**, Tong Xiao, Hongsheng Li, Bolei Zhou, Dayu Yue, and Xiaogang Wang, "Person Search with Natural Language Description." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (***CVPR***), 2017.

- **Shuang Li**, Tong Xiao, Hongsheng Li, Wei Yang, and Xiaogang Wang, "Identity-Aware Textual-Visual Matching with Latent Co-attention." In *Proceedings of the IEEE International Conference on Computer Vision* (***ICCV***), 2017.

- Wei Yang, **Shuang Li**, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang, "Learning Feature Pyramids for Human Pose Estimation." In *Proceedings of the IEEE International Conference on Computer Vision* (***ICCV***), 2017.

- **Shuang Li**, Slawomir Bak, Peter Carr, and Xiaogang Wang, "Diversity Regularized Spatiotemporal Attention for Video-based Person Re-identification." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (***CVPR***), 2018.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Artificial intelligence (AI) products have a significant impact on human society. Creating strong AI might be the biggest event in human history. AI motivates research in many areas, from economics and law to technical topics such as control and security. The role of computer vision to AI system is just like vision to human. Vision is the most important perception for people to sense the surrounding environment.

Person search, which aims at finding a specific person of interest from a gallery of images or videos, is a basic research problem for understanding computer vision. Human identification is tightly connected to AI systems. For example, the home security systems rely on person recognition to monitor house surroundings and warn housebreakers. Person detection and recognition are also applied to autonomous driving and smart phone apps.

Human can easily identify people through high-level attributes (face, gender, and age) and fine-grained details (cloth, pose, hair, accessories such as glasses, hairpin, and shoes). However, recognizing people is extremely difficult for computers. All things that the computer receives are digital signals. Computers need to understand the meaning of these numbers and transfer them to interpretable patterns which are also known as feature learning. Language-based person search is also necessary when there are only verbal descriptions of people's appearances available. However, learning good textual features is more difficult than learning visual features. Associating textual and visual patterns is also a challenging task.

The rapid development of deep learning methods provides better solutions to artificial intelligence problems. Instead of manually designing task-specific feature representations [3–6], deep learning methods [7–10] automatically learn them from the raw input data. The success of deep learning promotes more challenging visual tasks. Some

conventional image-based problems such as image classification and object detection have been extended to video level or combined with natural language processing. This motivates us to solve the video and sentence description based person search problems.

However, video-based and natural language based person search are not easy. First, people are often partially occluded in some video frames, which can corrupt the extracted features. On the other hand, a person's pose will change over time. Assembling an effective representation of a person from these various glimpses is difficult. Second, existing person search methods focus on searching persons with image-based or attribute-based queries. But image-based person search requires at least one photo of the query person being given. Attributes have limited capability of describing persons' appearance and they are expensive to be collected. Collecting attributes requires workers to go through the long list of attributes to find the corresponding ones. Language-based person search is able to enrich the person search task. New benchmark and approaches are in urgent demands. Thrid, encoding natural language is more difficult than image or video. It lacks enough supervised data and effective models for learning good features. The most commonly used textual encoding architecture, RNN, tend to miss important words appearing at the beginning of the sentence. The RNN is also unstable to sentence structures.

In this dissertation, we address these three challenges respectively, in order to make person search applicable to real-world scenarios.

In Chapter 2 we introduce some basic concepts about deep learning, including convolutional neural networks, recurrent neural networks, and some deep learning applications.

In Chapter 3 we present the background of person search, reviewing the person search categories and commonly used methods, datasets, and evaluation metrics.

In Chapter 4 we address the first challenge by proposing a spatiotemporal attention model for video-based person search. The network learns multiple spatial attention models and employs a diversity regularization term to ensure multiple models do not discover the same body part. Features extracted from local image regions are organized by spatial attention models and are combined using temporal attention models.

In Chapter 5 we collect a large-scale person description dataset with rich language annotations and person samples from various sources. We also design an effective gated

neural attention mechanism to capture the optimal word-image relations.

In Chapter 6 we propose a two-stage framework and extend natural language based person search to general textual-visual matching. The stage-1 network is able to embed textual and visual features to the same space efficiently, screen easy incorrect matchings, and provide initial training point for the stage-2 training. The stage-2 network adopts a spatial attention and a semantic attention to associate words and image regions. The two-stage model achieves both efficiency and accuracy.

Finally, we conclude this dissertation in Chapter 7.

Chapter 2

Deep Learning Basics

Deep learning is part of machine learning methods which learns data representations automatically. Categorized by the type of input data, learning can be supervised, semi-supervised, and unsupervised. This chapter introduces some basic concepts of deep learning.

Convolutional Neural Networks, or CNNs, are a family of neural networks for processing non-sequential data, such as static images. Much as ordinary neural networks that are made of neurons and have learnable weights and biases, convolutional neural networks utilize certain properties of the inputs, which vastly reduce the number of parameters in the network.

Recurrent Neural Networks, or RNNs, is a type of neural network that is specialized for processing sequential data, such as video, sentence, and audio. Different from convolutional neural networks, which accept a fixed-sized vector as input and output a fixed-sized vector, RNNs are scaleable to sequences with variable length because the recurrent transformations can be applied multiple times. RNNs have many important applications, such as image captioning, machine translations, and video classification.

2.1 Convolutional Neural Network

Regular neural networks consist of a series of hidden layers with each layer contains a set of neurons. Neurons in each layer are independent to each other, but they have connections to all neurons in the previous layer. Thus the full-connected structure cannot be used for processing large images. For example, there would be $120,000$ weights for a single neural network with only one hidden layer to encode images of size $200 \times 200 \times 3$.

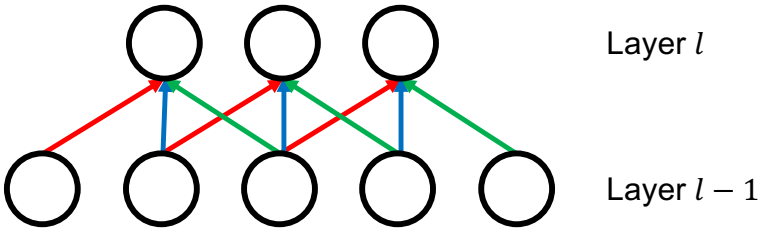


Figure 2.1: *Illustration of convolutional neural networks with shared weights.*

In convolutional neural networks, neurons in a layer will only be connected to a small region of the layer before it. In other words, the hidden neurons in layer l are from a subset of neurons in layer $l-1$ as shown in Figure 2.1. Neurons in layer l are only connected to 3 adjacent neurons in layer $l-1$. CNNs learn "filters" to encode the local visual patterns. In addition, each filter can be applied multiple times. In Figure 2.1, weights of the same color are identical which allows the visual patterns to be detected regardless of their locations in the input images. By sharing weights, CNNs can greatly reduce the number of parameters and increase learning efficiency. Convolutional neural networks have better generalization capacity on vision problems than regular neural networks.

A typical convolutional neural network is a sequence of linear and non-linear operators. The output of the previous layer serves as the input of the next layer. A simple CNN model for classification could have the architecture as shown in Figure 2.2. The neural network mainly consists of four types of layers: Convolutional Layer, Pooling Layer, ReLU Layer, and Fully-Connected Layer. Some layers have parameters and others don't. Next, we will introduce these layers.

2.1.1 Convolutional Layer

Convolutional operation is the main workhorse in CNNs. The use of filters in convolutional layers greatly reduces the model size. For example, a filter on the first layer might have size $3 \times 5 \times 5$ (5 pixels width and height for capturing the spatially local information, and 3 because images have 3 color channels). During the forward propagation, we slide the filter across the whole image and compute the inner products of the filter weight and the input at each position. A general form of the convolutional

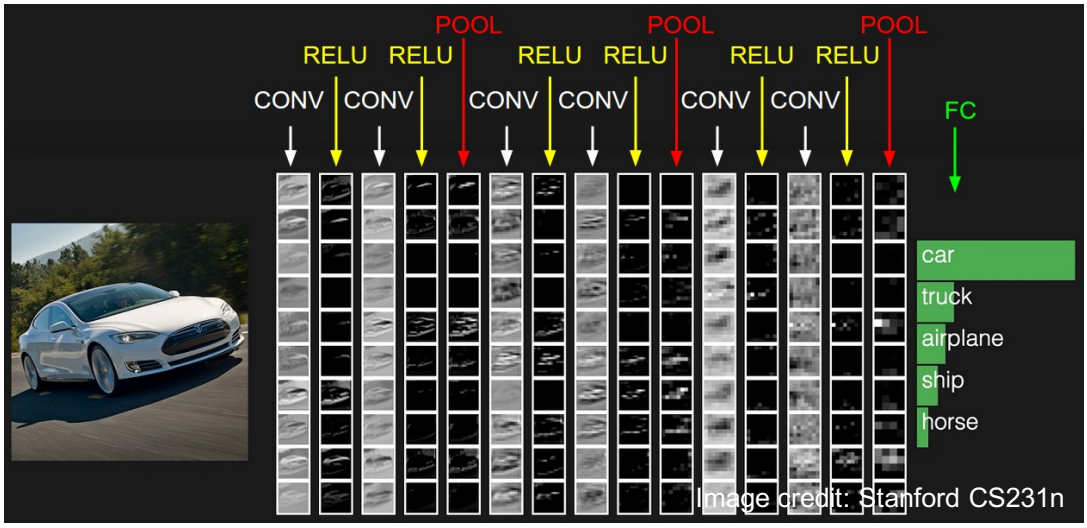


Figure 2.2: *An example CNN architecture for image classification. Source: Stanford CS231n*

operation can be written as follows:

$$y(i,j) = b + \sum_{c=1}^{C} \sum_{p=-K}^{K} \sum_{q=-K}^{K} w(c,p,q)x(c,i+p,j+q), \tag{2.1}$$

where $x \in \mathbb{R}^{C\times H\times W}$ is the input with spatial size $H \times W$ and C channels, $w \in \mathbb{R}^{C\times(2K+1)\times(2K+1)}$ is the parameter matrix of a filter. $b \in \mathbb{R}$ is the bias. $2K+1$ is the filter size. $y \in H \times W$ is an output feature map.

The filters will generate different feature maps, and thus they serve as pattern detectors to find different types of visual features such as shapes, edges, or colors in the input image.

2.1.2 Fully-Connected Layer

Fully-connected layer is a linear mapping:

$$y = Wx + b, \tag{2.2}$$

where $x \in \mathbb{R}^d$ is the input vector, $W \in \mathbb{R}^{m\times d}$ and $b \in \mathbb{R}^m$ are the weight matrix and bias vector respectively. Neurons in the fully-connected layer are connected to all neurons in the previous layer, and thus fully-connected layers have more parameters than convolutional layers. FC layers are often added to the end of a neural network to predict class probabilities.

2.1.3 ReLU Layer and Softmax Layer

The ReLU activation function has become popular in the past few years. It has the mathematical form:

$$\text{ReLU}(x) = \begin{cases} x & \text{when } x \geq 0, \\ 0 & \text{when } x < 0. \end{cases} \tag{2.3}$$

The ReLU layer thresholds a matrix of activation at zero. It is very easy to be implemented. However, the ReLU function could cause neurons never activate on any datapoint again because of the $x < 0$ part.

Leakey ReLU modifies the ReLU function to fix the "dying ReLU" problem. The Leakey ReLU adopts a slope in the negative region:

$$\text{LeakeyReLU}(x) = \begin{cases} x & \text{when } x \geq 0, \\ \alpha x & \text{when } x < 0, \end{cases} \tag{2.4}$$

where α is a small constant.

Softmax is another commonly used non-linear function. It squashes the output of each item to be in $(0, 1)$ and forces the sum of the output to be 1. The mathematical form is shown as follows:

$$\text{softmax}(x)_i = \frac{\exp(x_i)}{\sum_{j=1}^{d} \exp(x_j)}. \tag{2.5}$$

The outputs of softmax layer equal to the categorical probability distribution and can be sent to the loss function with target label to compute losses.

2.1.4 Pooling Layer

Pooling layer is another important operator. It is commonly inserted between two convolutional layers for reducing the parameter size. Pooling is a form of down-sampling along the spatial dimensions. The most common form is the max pooling with filters of size 2×2.

Max-pooling divides the input feature map into a set of non-overlapping grids and outputs the maximum value in each grid. The depth dimension remains unchanged, but the spatial size would decrease following the formulation: $H_o = (H - F)/F + 1$, $W_o = (W - F)/F + 1$. H and W are the input height and width, F is the filter size,

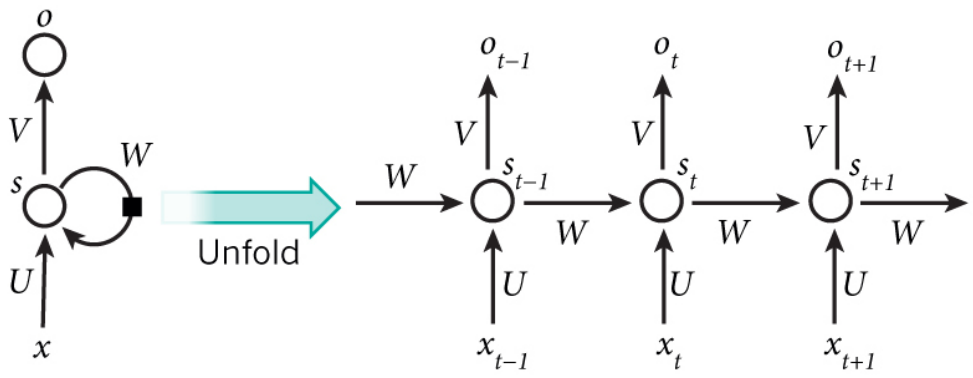


Figure 2.3: *A recurrent neural network and the unfolding form. Source: Nature*

and S is the stride of down-sampling along both the width and height.

2.2 Recurrent Neural Network

The idea of RNNs is to utilize sequential information. A typical RNN model is shown in Figure 2.3. The RNN is unrolled into a full network. In the forward propagation, we send the sequential data to the neural network one by one and get an output at each time step (this is not necessary, for some specific tasks, we may only care about the final output). The RNN shares parameters across all steps to reduce the parameter size.

Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are the most commonly used RNNs. LSTMs and GRUs have the same architecture as RNNs, but they use different ways to compute the hidden state s_t. x_t and o_t are the input and output at time step t in Figure 2.3. U, W, V are transformation matrices. Next, we will take the machine translation as an example to introduce LSTM and GRU models.

2.2.1 Long Short-Term Memory

RNN models encode sequential data and utilize previous information to the present task. However, RNNs tend to focus on the most recent inputs and forget the beginning ones. When the input sequence is too long, the gap between the relevant information and point where it is needed becomes very large which makes the RNNs cannot connect that information.

Long short-term memory networks are designed by Hochreiter *et al.* [11] to avoid

the long-term dependency problem. There are three gates and one memory cell in LSTMs. The gates control the flow direction of information while the memory cell preserves knowledge of previous steps. At each word, the LSTMs update the memory cell c_t and output a hidden state h_t in the following way:

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \\ f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \\ c_t &= f_t \odot c_{t-1} + i_t \odot h(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \\ s_t &= o_t \odot h(c_t), \end{aligned} \tag{2.6}$$

where $\odot$ represents the element-wise multiplication, W and b are parameters to be learned, *sigma* and h are the sigmoid function and tanh function respectively. The forget gate f is a sigmoid function, which looks at the hidden state of the last step h_{t-1} and the input word x_t, and outputs a scalar to indicate which value to be thrown away from the cell state. The input gate i and output gate o decide which information to be updated and output respectively. c is a function of the old subjects and the new input.

2.2.2 Gated Recurrent Unit

GRU model is first introduced by Kyunghyun *et al.* [12]. It has a simpler structure and fewer parameters than LSTM. The implementation is shown below:

$$\begin{aligned} z_t &= \sigma(W_{xz}x_t + W_{hz}h_{t-1} + b_z), \\ r_t &= \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r), \\ h_t &= h(W_{xh}x_t + W_{hh}(r_t \odot s_{t-1}) + b_h), \\ s_t &= (1 - z_t) \odot s_{t-1} + z_t \odot h_t, \end{aligned}$$

A GRU has two gates, a reset gate r and an update gate z. The reset gate decides how to combine the new and old information. The update gate determines which value to be kept. Different from LSTMs, which have three gates and an internal memory cell c_t, GRUs don't have the output gate. The reset gate is applied directly to the previous hidden state. Because of the fewer parameters, GRUs need less time and data

to generalize when training.

2.3 Deep Learning Applications

2.3.1 Image Classification

Deep convolutional neural networks have led to great success for large-scale image classification. AlexNet [7] roses the interest in CNNs. It is considered to be the breakthrough method when winning the ImageNet challenge of 2012. The network consists of 5 convolutional layers, 3 fully-connected layers, and some dropout layers and max-pooling layers. It also contains data augmentation and ReLU non-linear activation functions. VGG Net [8] replaces the 7×7 filters in AlexNet with 3×3 filters. The intuition is that 2 consecutive 3×3 filters and a 5×5 filter have the same receptive field, and thus 3 consecutive 3×3 filters give a receptive field of a 7×7 filter. The introduction of smaller filters reduce the number of parameters and the deep models can be designed deeper.

GoogLeNet [13] has several inception modules and eliminates all fully-connected layers which save lots of parameters. Microsoft Resnet [14] wins the 2015 ImageNet challenge. The Residual block is represented as follows:

$$y = f_\theta(x) + x, \tag{2.7}$$

He *et al.* show the model introduces neither extra parameter nor computation complexity, and can be designed very deep.

2.3.2 Object Detection

Object detection aims at identifying the location of objects in an image. Early methods [15] generate regions of interest or region proposals and send them to CNN models to learn feature representations. In [15], Girshick *et al.* propose the Fast R-CNN which significantly improves the training and testing speed and accuracy. Instead of extracting features of each candidate bounding box, they forward the image only once and the features of each bounding box are generated by a RoI pooling layer. However, the detection accuracy of Fast R-CNN depends on the performance of the region proposal module.

Faster R-CNN [16] combines the proposal detection and Fast R-CNN in a single deep neural network. They introduce a novel Region Proposal Network that shares convolutional layers with down-stream detection networks. The RPN predicts object bounds and objectness scores at each position simultaneously which enables nearly cost-free region proposals. Faster R-CNN realizes the real-time object detection.

2.3.3 Generative Adversarial Networks

Generative adversarial networks (GANs) are deep neural networks comprised of a generator (G) and a discriminator (D). The generator takes random inputs and generates new data instances to fool the discriminator. The discriminator tries to classify whether the input data belonging to the actual training dataset or not. The generator and discriminator play the following minimax game:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[logD(x)] + \mathbb{E}_{x\sim p_z(z)}[log(1-D(G(z)))], \tag{2.8}$$

DCGAN [17] introduces deep convolutional neural networks to generative adversarial models and shows that the deep generative networks can learn good representations of images. Wasserstein GAN [18] provides comprehensive theoretical analysis of GAN models.

Chapter 3

Person Search Background

Person search aims at finding a particular person from the gallery of pedestrian images. Because of its important applications in solving real-world problems, person search has received much attention in the past few years. Solving this problem is not easy. The complex variations of human poses, image resolutions, lighting conditions, camera viewpoints, occlusions, and background clutter make the person search more challenging. In Section 3.1, we introduce existing person search categories. In Section 3.2 and Section 3.3, we discuss some basic knowledge of person search, including the commonly used methods and datasets. Section 3.4 contains several evaluation methods.

3.1 Person Search Categories

Existing works of person search focus on automatically learning features with convolutional neural networks [19, 20] and learning distance metrics [21–23]. Considering the type of input, person search can be categorized into image-based person search, video-based person search, and natural language based person search.

Image-based Person Search. Image-based person search, or person re-identification, usually uses [5, 24–30] convolutional neural networks to learn features. Ahmed *et al.* [20] send a pair of images to a specifically designed CNN model with a binary classification loss for person re-identification. Ding *et al.* [31] control the distances within the same person to be smaller than distances between different people using a triplet loss function.

Video-based Person Search. Video-based person search is an extension of image-based person search. The input is video sequences. Some people treat videos as a set of images and regard the video-based person search as multi-shot learning [32–34]. Others extract the temporal information using RNN models [35–37] or Optical Flow [35, 38].

In [35], McLaughlin *et al.* introduce a RNN model to encode temporal information. Chung *et al.* [38] present a two stream convolutional neural network to learn spatial and temporal information separately.

Natural Language based Person Search. Image-based and video-based person search have major limitations in practice. They require at least one photo of the queried person being given. But in many criminal cases, there might be only verbal descriptions of the suspects' appearance available. To close the gap, natural language descriptions are used as queries to search people. It does not require a person photo to be given as in those image-based or video-based query methods.

3.2 Person Search Methods

3.2.1 Identity Feature Learning

Existing person search methods usually take the classification loss or triplet loss as the supervision to learn feature representations. There are two types of classification settings as shown in Figure 3.1 (a) and (b). The first type of methods treat the person search as a classification problem with the softmax loss function, *i.e.* each person corresponds to a class in the training stage [6,30,39,40]. In the testing stage, we extract outputs before the last classification layer as person features and the similarity of two persons is evaluated by their feature distance. Even though this type of methods are easy to be implemented, there is a mismatch between the training and testing — the training is a classification problem while the testing is a verification problem.

The second type of person search methods take two images as input and predict whether they contain the same person or not [20,41] (Figure 3.1 (b)). The networks have a binary classification loss. We use the value of being class 1 as their matching score. However, methods with binary classification loss are time-consuming during the testing stage. To evaluate the performance of the proposed method, we need to construct N^2 image pairs and extract their features. N is the number of testing samples. This may lead to longer evaluation time, especially on large-scale datasets.

For methods using triplet loss functions [42,43] (Figure 3.1 (c)), their input consists of one anchor, one positive sample, and one negative sample. We train the model to minimize distances within the same person and maximize distances between different people.

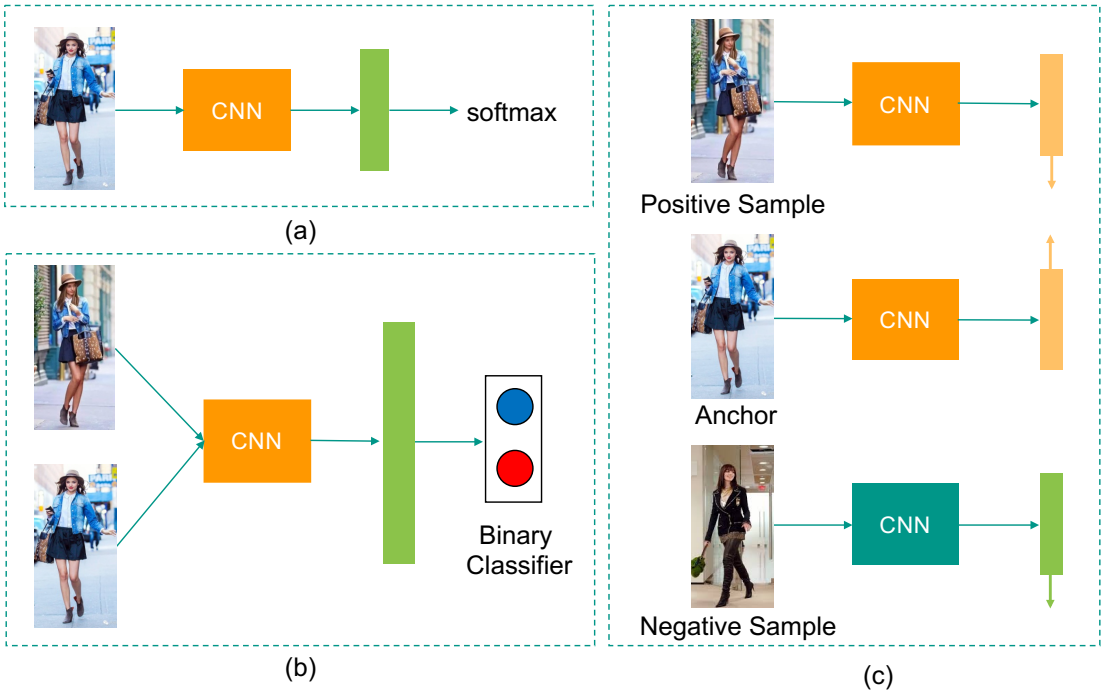


Figure 3.1: *Identity feature learning using different loss functions.*

A new loss function, *e.g.* Online Instance Matching, is proposed by Xiao *et al.* [28] recently. This loss function aims at solving the inefficiency of learning a large classification matrix in methods with the classification loss. The OIM is non-parametric. The gradients directly operate on the features without the transformation by a classifier matrix.

3.2.2 Feature Comparison

Metric learning [6, 24, 44–47] is commonly used in early person search methods to compute the distance of two images. Suppose x_1 and x_2 are the feature representations of two persons, metric learning replaces conventional L2-distance by learning a distance function $d(x_1, x_2)$. Metric learning can be understood as learning a linear transformation on features, and thus it can be merged to deep learning models through fully-connected layers. The combination of a simple deep neural network and an L2-distance metric is a popular way to solve the person search problem.

Feature maps contain spatial information, but two feature maps are usually not well aligned. In the same position of two images, one could be a person's head and the other could be just the background. Comparing the distance of two feature maps

Dataset	Time	#ID	#Image	#Cameras	Label	Sequences
VIPeR [49]	2007	632	1,264	2	hand	×
QMUL iLIDS [50]	2009	119	476	2	hand	×
GRID [51]	2009	250	1,275	8	hand	×
3DPeS [52]	2011	192	1,011	8	hand	×
CUHK01 [53]	2012	971	3,884	2	hand	×
CUHK02 [54]	2013	1,816	7,264	10	hand	×
CUHK03 [19]	2014	1,467	13,164	2	hand/DPM	×
CUHK-SYSU [55]	2016	8,432	18,184	-	hand	×
Market-1501 [56]	2015	1,501	32,668	6	hand/DPM	×
DukeMTMC [57]	2017	1,812	36,441	8	hand	×
DukeMTMC4ReID [57]	2017	1,852	46,261	8	Doppia	×
PRID2011 [58]	2011	934	24,541	2	hand	✓
iLIDS-VID [59]	2014	300	42,495	2	hand	✓
MARS [60]	2016	1,261	1,191,003	6	DPM+GMMCP	✓

Table 3.1: *Statistics of some commonly used datasets for person search.*

is also an important strategy to evaluate their similarity. Li *et al.* [19] compute the correlation between the strips on two feature maps to estimate the similarity. In [20], Ahmed *et al.* change the correlation from strips to local rectangle regions and improve the accuracy significantly.

Bilinear pooling models also show impressive performance on fine-grained recognition tasks. They capture second-order statistics of convolutional features in a translationally invariant manner. Bilinear pooling methods have repeatedly shown to produce state-of-the-art results. However, they are not widely adopted because their high dimensional features are impractical for subsequent analysis. Gao *et al.* [48] propose two compact bilinear pooling methods to reduce the feature dimension. The compact bilinear pooling allows efficient end-to-end training.

3.3 Datasets

A number of datasets have been released for research purpose. We list several the most commonly used image-based person search datasets and video-based datasets in Table 3.1. VIPeR [49] is the most popular dataset, containing 632 identities and 1,264 images. 10 random train/test splits are predefined for stable evaluations. GRID [51] is captured by 8 disjoint cameras in an underground station. Each identity has two images from different camera views. The image quality of this dataset is fairly poor. QMUL iLIDS [50] and iLIDS-VID [59] are captured at the airport. They have scenarios with heavy occlusion and pose variance. CUHK01 [53], CUHK02 [54], and CUHK03 [19] are

collected from CUHK campus. In most existing datasets, the ground truth pedestrians are manually annotated. However, as the dataset size increases, CUHK03 [19] and Market-1501 [56] introduce pedestrian detectors, such as DPM, to generate annotations. DukeMTMC4ReID [57] uses the Doppia detector. Samples in CUHK-SYSU [55] are mainly from street snaps and movies. Different from previous image-based datasets with cropped images, CUHK-SYSU provides whole scene images. This could trigger researchers to design methods that do pedestrian detection and person re-identification simultaneously. The problem setting of CUHK-SYSU is closer to real-world applications.

PRID2011 [58], iLIDS-VID [59], and MARS [60] are three video-based person search datasets. PRID2011 consists of person videos from two camera views. There are 385 and 749 identities in camera A and camera B respectively. Only the first 200 people appear in both cameras. The sequence length ranges from 5 to 675 frames. iLIDS-VID contains 600 image sequences of 300 persons. For each person, we have two videos with the length of each video sequence varying from 23 to 192 frames. The average duration is 73 frames. MARS dataset is the largest video-based person search benchmark with 1,261 identities and around 20,000 video sequences. They adopt DPM detector [61] and GMMCP tracker [62] to extract the person sequence. Zheng *et al.* [60] set 6 cameras to collect data in the university campus. Each identity is captured by at least 2 cameras. This dataset also contains 3,248 distractor sequences.

3.4 Evaluation Metrics

Cumulative Matching Characteristics (CMC) curve is usually used to evaluate person search algorithms. Taking the single-gallery-shot as an example, where there is only one matching identity for each query, the algorithm ranks all the gallery samples according to their similarities to the query. The CMC top-k accuracy is

$$\text{cmc}_k = \begin{cases} 1 & \text{if the top-}k\text{ ranked gallery samples contains the correct matching,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

The final CMC curve is the average value over all queries. This evaluation metric is acceptable to single-gallery-shot datasets. However, when it comes to the multi-gallery-shot setting, where each query has multiple ground truths existing in the gallery set, people define different criterions on different datasets.

Zheng *et al.* [56] propose to use mean average precision (mAP) in the Market-1505 dataset. They compute the area under the precision-recall curve, *i.e.* average precision (AP). The mAP is the mean value of AP. In CUHK03, Li *et al.* randomly sample one instance for each gallery identity and compute the CMC curve. The process is repeated multiple times to generate the multi-gallery-shot CMC curve.

Chapter 4

Diversity Regularized Spatiotemporal Attention for Video-based Person Search

Video-based person search matches video clips of people across non-overlapping cameras. Most existing methods tackle this problem by encoding each video frame in its entirety and computing an aggregate representation across all frames. In practice, people are often partially occluded, which can corrupt the extracted features. Instead, we propose a new spatiotemporal attention model that automatically discovers a diverse set of distinctive body parts. This allows useful information to be extracted from all frames without succumbing to occlusions and misalignments. The network learns multiple spatial attention models and employs a diversity regularization term to ensure multiple models do not discover the same body part. Features extracted from local image regions are organized by spatial attention model and are combined using temporal attention. As a result, the network learns latent representations of the face, torso and other body parts using the best available image patches from the entire video sequence. Extensive evaluations on three datasets show that our framework outperforms the state-of-the-art approaches by large margins on multiple metrics.

We propose a new spatiotemporal attention scheme that effectively handles the difficulties of video-based person search. Instead of directly encoding the whole image (or a predefined decomposition, such as a grid), we use multiple spatial attention models to localize discriminative image regions, and pool these extracted local features across time using temporal attention. Our approach has several useful properties:

- Spatial attention explicitly solves the alignment problem between images, and avoids features from being corrupted by occluded regions.

- Although many discriminative image regions correspond to body parts, accessories like sunglasses, backpacks and hats; are prevalent and useful for identification. Because these categories are hard to predefine, we employ an unsupervised learning approach and let the neural network automatically discover a set of discriminative object part detectors (spatial attention models).
- We employ a novel diversity regularization term based on the Hellinger distance to ensure multiple spatial attention models do not discover the same body part.
- We use temporal attention models to compute an aggregate representation of the features extracted by each spatial attention model. These aggregate representations are then concatenated into a final feature vector that represents all of the information available from the entire video.

4.1 Related Work

Person search, or person re-identificaiton, matches images of pedestrians in one camera with images of pedestrians from another, non-overlapping camera. This task has drawn increasing attention in recent years due to its importance in applications, such as surveillance [63], activity analysis [51] and tracking [64]. It remains a challenging problem because of complex variations in camera viewpoints, human poses, lighting, occlusions, and background clutter.

Person search was first proposed for multi-camera tracking [63, 65]. Gheissari *et al.* [66] designed a spatial-temporal segmentation method to extract visual cues and employed color and salient edges for foreground detection. This work defined the image-based person search as a specific computer vision task.

Image-based person search. Image-based person search mainly focuses on two categories: extracting discriminative features [5, 22, 27, 29, 30] and learning robust metrics [21, 24–26, 67]. In recent years, researchers have proposed numerous deep learning based methods [19, 20, 28, 31, 68] to jointly handle both aspects. Ahmed *et al.* [20] input a pair of cropped pedestrian images to a specifically designed CNN with a binary verification loss function for person search. In [31], Ding *et al.* minimize feature distances between the same person and maximize the distances among different people by employing a triplet loss function when training deep neural networks. Xiao *et*

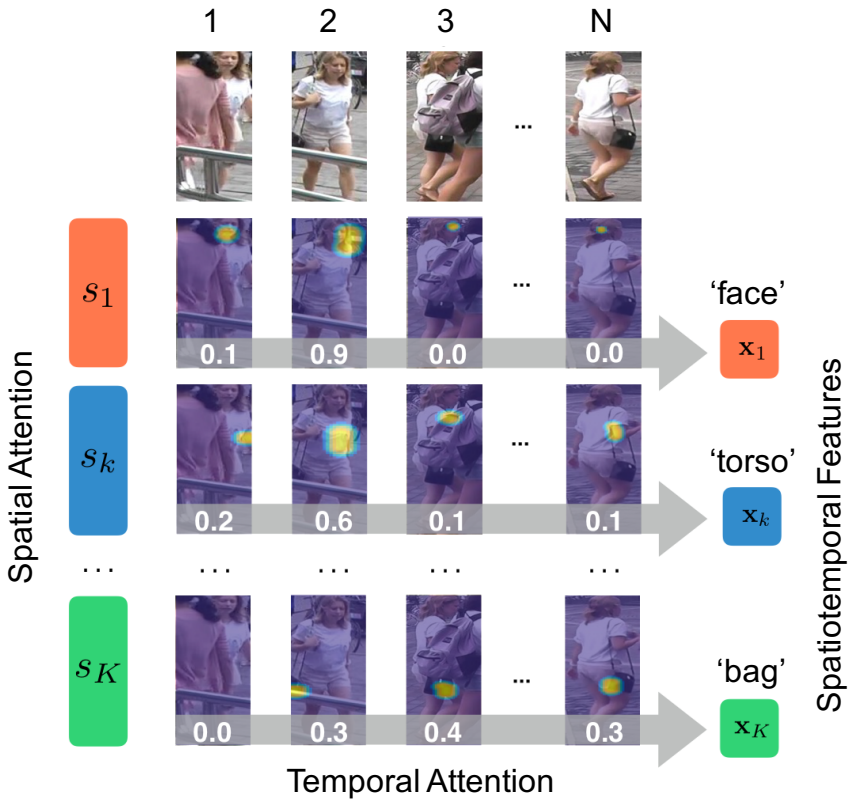


Figure 4.1: ***Spatiotemporal Attention.*** *In challenging video search scenarios, a person is rarely fully visible in all frames. However, frames in which only part of the person is visible often contain useful information. For example, the face is clearly visible in the frames 1 and 2, the torso in frame 2, and the handbag in frames 2, 3 and N. Instead of averaging full frame features across time, we propose a new spatiotemporal approach which learns to detect a set of K diverse salient image regions within each frame (superimposed heatmaps). An aggregate representation of each body part is then produced by combining the extracted per-frame regions across time (weights shown as white text). Our spatiotemporal approach creates a compact encoding of the video that exploits useful partial information in each frame by leveraging multiple spatial attention models, and combining their outputs using multiple temporal attention models.*

al. [28] jointly train the pedestrian detection and person re-identification in a single CNN model. They propose an Online Instance Matching loss function which learns features more efficiently in large scale verification problems.

Video-based person search. Video-based person search [35, 36, 59, 69–71] is an extension of image-based approaches. Instead of pairs of images, the learning algorithm is given pairs of video sequences. In [69], You *et al.* present a top-push distance learning model accompanied by the minimization of intra-class variations to optimize the matching accuracy at the top rank for person search. McLaughlin *et al.* [35] introduce an RNN model to encode temporal information. They utilize temporal pooling to select

the maximum activation over each feature dimension and compute the feature similarity of two videos. Wang *et al.* [59] select reliable space-time features from noisy/incomplete image sequences while simultaneously learning a video ranking function. Ma *et al.* [71] encode multiple granularities of spatiotemporal dynamics to generate latent representations for each person. A Time Shift Dynamic Time Warping model is derived to select and match data between inaccurate and incomplete sequences.

Attention models for person search. Attention models [72–74] have grown in popularity since [72]. Zhou *et al.* [36] combine spatial and temporal information by building an end-to-end deep neural network. An attention model assigns importance scores to input frames according to the hidden states of an RNN. The final feature is a temporal average pooling of the RNN's outputs. However, if trained in this way, corresponding weights at different time steps of the attention model tend to have the same values. Liu *et al.* [75] proposed a multi-directional attention module to exploit the global and local contents for image-based person search. However, jointly training multiple attentions might cause the mode collapse. The network has to be carefully trained to avoid attention models focusing on similar regions with high redundancy. In this paper, we combine spatial and temporal attentions into spatiotemporal attention models to address the challenges in video-based person search. For spatial attention, we use a penalization term to regularize multiple redundant attentions. We employ temporal attention to assign weights to different salient regions on a per-frame basis to take full advantage of discriminative image regions. Our method demonstrates better empirical performance, and decomposes into an intuitive network architecture.

4.2 Method Overview

In this chapter, we investigate the problem of video-based person search, which is a generalization of the standard image-based search task. Instead of matching image pairs, the algorithm must match pairs of video sequences (possibly of different durations). A key challenge in this paradigm is developing a good latent feature representation of each video sequence.

Existing video-based person search methods represent each frame as a feature vector and then compute an aggregate representation across time using average or maximum pooling [36,69,76]. Unfortunately, this approach has several drawbacks when applied to

datasets where occlusions are frequent (Fig. 4.1). The feature representation generated for each image is often corrupted by the visual appearances of occluders. However, the remaining visible portions of the person may provide strong cues for person search. Assembling an effective representation of a person from these various glimpses should be possible. However, aggregating features across time is not straightforward. A person's pose will change over time, which means any aggregation method must account for spatial misalignment (in addition to occlusion) when comparing features extracted from different frames.

We propose a new deep learning architecture (Fig. 4.2) to better handle video search by automatically organizing the data into sets of consistent salient subregions. Given an input video sequence, we first use a restricted random sampling strategy to select a subset of video frames (Sec. 4.3). Then we send the selected frames to a multi-region spatial attention module (Sec. 4.4) to generate a diverse set of discriminative spatial gated visual features—each roughly corresponding to a specific salient region of a person (Sec. 4.4.1). The overall representation of each salient region across the duration of the video is generated using temporal attention (Sec. 4.5). Finally, we concatenate all temporal gated features and send them to a fully-connected layer which represents the latent spatiotemporal encoding of the original input video sequence. An OIM loss function, proposed by Xiao *et al.* [28], is built on top of the FC layer to supervise the training of the whole network in an end-to-end fashion. However, any traditional loss function (like softmax) could also be employed. We demonstrate the effectiveness of our approach on three challenging video-based person search datasets. Our technique out performs the state-of-the-art methods under multiple evaluation metrics.

4.3 Restricted Random Sampling

Previous video-based person search methods [35, 36, 71] do not model long-range temporal structure because the input video sequences are relatively short. To some degree, this paradigm is only slightly more complicated than image-based search since consecutive video frames are highly correlated, and the visual features extracted from one frame do not change drastically over the course of a short sequence. However, when input video sequences are long, any search methodology must be able to cope with significant visual changes over time, such as different body poses and angles relative to

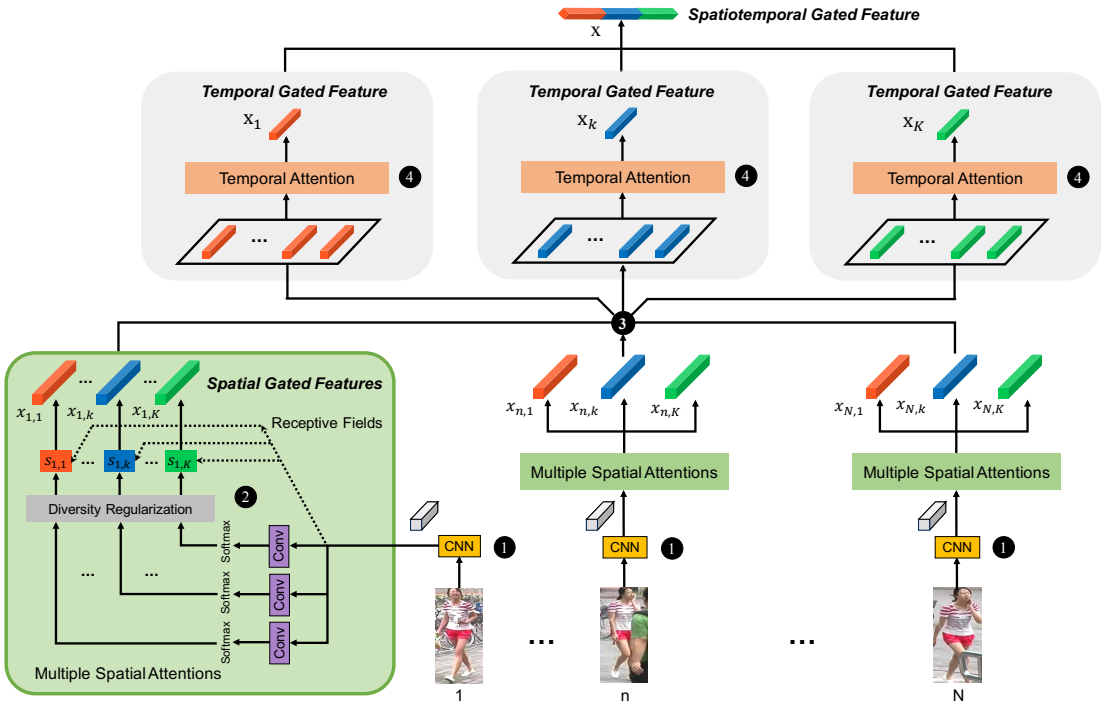


Figure 4.2: ***Spatiotemporal Attention Network Architecture***. *The input video is reduced to N frames using restricted random sampling. (1) Each image is transformed into feature maps using a CNN. (2) These feature maps are sent to a conventional network followed by a softmax function to generate multiple spatial attention models and corresponding receptive fields for each input image. A diversity regularization term encourages learning spatial attention models that do not result in overlapping receptive fields per image. Each spatial attention model discovers a specific salient image region and generates a spatial gated feature (Fig. 4.3). (3) Spatial gated features from all frames are grouped by spatial attention model. (4) Temporal attentions compute an aggregate representation for the set of features generated by each spatial attention model. Finally, the spatiotemporal gated features for all body parts are concatenated into a single feature which represents the information contained in the entire video sequence.*

the camera.

Wang *et al.* [77] proposed a temporal segment network to generate video snippets for action recognition. Inspired by them, we propose a restricted random sampling strategy to generate compact representations of long video sequences that still provide good representations of the original data. Our approach enables models to utilize visual information from the entire video and avoids the redundancy between sequential frames. Given an input video $\mathbf{V}$, we divide it into N chunks $\{C_n\}_{n=1,N}$ of equal duration. From each chunk C_n, we randomly sample an image I_n. The video is then represented by the ordered set of sampled frames $\{I_n\}_{n=1,N}$.

4.4 Multiple Spatial Attention Models

We employ multiple spatial attention models to automatically discover salient image regions (body parts or accessories) useful for person search. Instead of pre-defining a rigid spatial decomposition of input images (*e.g.* a grid structure), our approach automatically identifies multiple disjoint salient regions in each image that consistently occur across multiple training videos. Because the network learns to identify and localize these regions (e.g. automatically discovering a set of object part detectors), our approach mitigates registration problems that arise from pose changes, variations in scale, and occlusion. Our approach is not limited to detecting human body parts. It can focus on any informative image regions, such as hats, bags and other accessories often found in person search datasets. Feature representations directly generated from entire images can easily miss fine-grained visual cues (Fig. 4.1). Multiple diverse spatial attention models, on the other hand, can simultaneously discover discriminative visual features while reducing the distraction of background contents and occlusions. Although spatial attention is not a new concept, to the best of our knowledge, this is first time that a network has been designed to automatically discover a diverse set of attentions within image frames that are consistent across multiple videos.

As shown in Fig. 4.2, we adopt the ResNet-50 CNN architecture [14] as our base model for extracting features from each sampled image. The CNN has a convolutional layer in front (named $conv1$), followed by four residual blocks. We exploit $conv1$ to $res5c$ as the feature extractor. As a result, each image I_n is represented by an 8×4 grid of feature vectors $\{\mathbf{f}_{n,\ell}\}_{\ell=1,L}$, where $L = 32$ is the number of grid cells, and each feature is a $D = 2048$ dimensional vector.

Multiple attention models are then trained to locate discriminative image regions (distinctive object parts) within the training data. For the k^{th} model, $k \in (1, \ldots, K)$, the amount of spatial attention $s_{n,k,\ell}$ given to the feature vector in cell ℓ is based on a response $e_{n,k,\ell}$ generated by passing the feature vector through two linear transforms and a ReLU activation in between. Specifically,

$$e_{n,k,\ell} = (\mathbf{w}'_{s,k})^{\mathsf{T}} \max(\mathbf{W}_{s,k}\mathbf{f}_{n,\ell} + \mathbf{b}_{s,k}, 0) + b'_{s,k}, \tag{4.1}$$

where $\mathbf{w}'_{s,k} \in \mathbb{R}^d$, $\mathbf{W}_{s,k} \in \mathbb{R}^{d \times D}$, $\mathbf{b}_{s,k} \in \mathbb{R}^d$ and $b'_{s,k} \in \mathbb{R}$ are parameters to be learned

for the k^{th} spatial attention model. The first linear transform projects the original feature to a lower $d = 256$ dimensional space, and the second transform produces a scalar value for each feature/cell. The attention for each feature/cell is then computed as the softmax of the responses

$$s_{n,k,\ell} = \frac{\exp(e_{n,k,\ell})}{\sum_{j=1}^{L} \exp(e_{n,k,j})}. \tag{4.2}$$

The set $\mathbf{s}_{n,k} = [s_{n,k,1}, \ldots, s_{n,k,L}]$ of weights defines the *receptive field* of the k^{th} spatial attention model (part detector) for image I_n. By definition, each receptive field is a probability mass function since $\sum_{\ell=1}^{L} s_{n,k,\ell} = 1$.

For each image I_n, we generate K spatial gated visual features $\{\mathbf{x}_{n,k}\}_{k=1,K}$ using attention weighted averaging

$$\mathbf{x}_{n,k} = \sum_{\ell=1}^{L} s_{n,k,\ell} \mathbf{f}_{n,\ell}. \tag{4.3}$$

Each gated feature represents a salient part of the input image (Fig. 4.3). Because $\mathbf{x}_{n,k}$ is computed by pooling over the entire grid $\ell \in [1, L]$, the spatial gated feature contains no information about the image location from which it was extracted. As a result, the spatial gated features generated for a particular attention model across multiple images are all roughly aligned—*e.g.* extracted patches of the face all tend to have the eyes in roughly the same pixel location.

Similar to fine-grained object recognition [78], we pool information across frames to created an enhanced variant

$$\widehat{\mathbf{x}}_{n,k} = E(\mathbf{x}_{n,k}) \tag{4.4}$$

of each spatial gated feature. The enhancement function $E()$ follows the past work on second-order pooling [79]. See the supplementary material for further details.

4.4.1 Diversity Regularization

The outlined approach for learning multiple spatial attention models can easily produce a degenerate solution. For a given image, there is no constraint that the receptive field generated by one attention model needs to be different from the receptive field of another model. In other words, multiple attention models could easily learn to detect

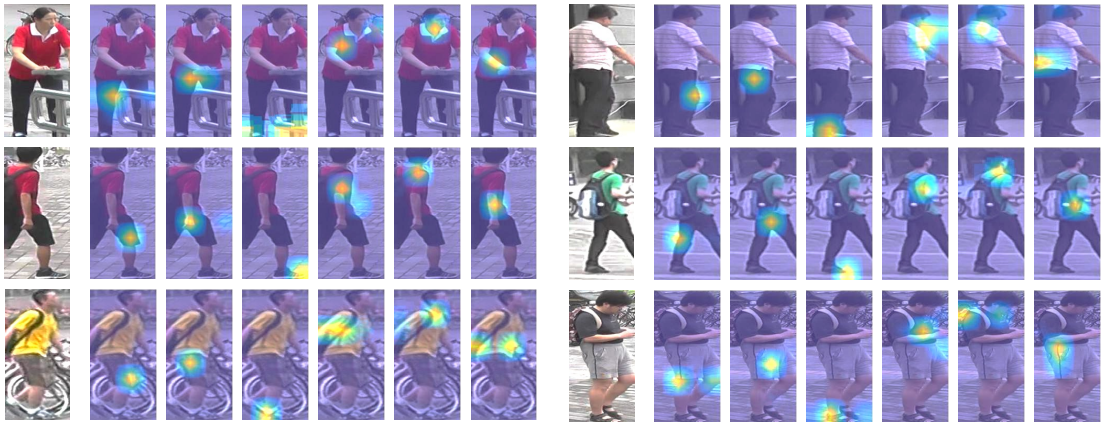


Figure 4.3: ***Learned Spatial Attention Models***. *Example images and corresponding receptive fields for our diverse spatial attention models when $K = 6$. Our methodology discovers distinctive image regions which are useful for person search. The attention models primarily focus on foreground regions and generally correspond to specifc body parts. Our interpretation of each is indicated at the bottom of each column.*

the same body part. In practice, we need to ensure each of the N spatial attention models focuses on different regions of the given image.

Since each receptive field $\mathbf{s}_{n,k}$ has a probabilistic interpretation, one solution is to use the Kullback-Leibler divergence to evaluate the diversity of the receptive fields for a given image. For notational convenience, we define the matrix $\mathbf{S}_n \in \mathbb{R}^{K \times L}$ as the collection of receptive fields generated for image I_n by the K spatial attention models

$$\mathbf{S}_n = [\mathbf{s}_{n,1}, \ldots, \mathbf{s}_{n,K}]. \tag{4.5}$$

Typically, the attention matrix has many values close to zero after the softmax() function, and these small values drop sharply when passed though the log() operation in the Kullback-Leibler divergence. In this case, the empirical evidence suggests the training process is unstable [80].

To encourage the spatial attention models to focus on different salient regions, we design a penalty term which measures the overlap between different receptive fields. Suppose $\mathbf{s}_{n,i}$ and $\mathbf{s}_{n,j}$ are two attention vectors in attention matrix $\mathbf{S}_n$. Employing the probability mass property of attention vectors, we use the Hellinger distance [81] to

measure the similarity of $\mathbf{s}_{n,i}$ and $\mathbf{s}_{n,j}$. The distance is defined as

$$H(\mathbf{s}_{n,i}, \mathbf{s}_{n,j}) = \frac{1}{\sqrt{2}} \sqrt{\sum_{\ell=1}^{L} (\sqrt{s_{n,i,\ell}} - \sqrt{s_{n,j,\ell}})^2}, \tag{4.6}$$

$$= \frac{1}{\sqrt{2}} \| \sqrt{\mathbf{s}_{n,i}} - \sqrt{\mathbf{s}_{n,j}} \|_2. \tag{4.7}$$

Since $\sum_{\ell=1}^{L} s_{n,k,\ell} = 1$:

$$H^2(\mathbf{s}_{n,i}, \mathbf{s}_{n,j}) = 1 - \sum_{\ell=1}^{L} (\sqrt{s_{n,i,\ell} s_{n,j,\ell}}). \tag{4.8}$$

To ensure diversity of the receptive fields, we need to maximize the distance between $\mathbf{s}_{n,i}$ and $\mathbf{s}_{n,j}$, which is equivalent to minimizing $1 - H^2(\mathbf{s}_{n,i}, \mathbf{s}_{n,j})$. We introduce $\mathbf{R}_n = \sqrt{\mathbf{S}_n}$ for notation convenience, where each element in $\mathbf{R}_n$ is the square root of the corresponding element in $\mathbf{S}_n$. Thus, the regularization term to measure the redundancy between receptive fields per image is

$$Q = \| (\mathbf{R}_n \mathbf{R}_n^\mathsf{T} - \mathbf{I}) \|_F^2, \tag{4.9}$$

where $\| \cdot \|_F$ denotes the Frobenius norm of a matrix and $\mathbf{I}$ is a K-dimensional identity matrix. This regularization term Q will be multiplied by a coefficient, and added to the original OIM loss.

Diversity regularization was recently employed for text embedding using recurrent networks [80]. In this case, the authors employed a variant

$$Q' = \| (\mathbf{S}_n \mathbf{S}_n^\mathsf{T} - \mathbf{I}) \|_F^2 \tag{4.10}$$

of our proposed regularization. Although Q and Q' have similar formulations, the regularization effects are very different. Q is based on probability mass distributions with the constraint $\sum_{\ell=1}^{L} \mathbf{s}_{n,k,\ell} = 1$ while Q' can be formulated on any matrix. Q' encourages $\mathbf{S}_n$ to be sparse – preferring only non-zero elements along the diagonal of $\mathbf{S}_n$. Although Q' forces the receptive fields not to overlap, it also encourages them to be concentrated to a single cell. Q, on the other hand, allows large salient regions like "upperbody" while discouraging receptive fields from overlapping. We compare the performances of the two regularization terms Q and Q' in Section 4.6.3.

4.5 Temporal Attention

Recall that each frame I_n is represented by a set $\{\widehat{\mathbf{x}}_{n,1}, \ldots, \widehat{\mathbf{x}}_{n,K}\}$ of K enhanced spatial gated features, each generated by one of the K spatial attention models. We now consider how best to combine these features extracted from individual frames to produce a compact representation of the entire input video.

All parts of an object are seldom visible in every video frame—either because of self-occlusion or from an explicit foreground occluder (Fig. 4.1). Therefore, pooling features across time using a per-frame weight t_n is not sufficiently robust, since some frames could contain valuable partial information about an individual (*e.g.* face, presence of a bag or other accessory, etc.).

Instead of applying the same temporal attention weight t_n to all features extracted from frame I_n, we apply multiple temporal attention weights $\{t_{n,1}, \ldots, t_{n,K}\}$ to each frame—one for each spatial component. With this approach, our temporal attention model is able to assess the importance of a frame based on the merits of the different salient regions. Temporal attention models which only operate on whole frame features could easily lose fine-grained cues in frames with moderate occlusion.

Similarly, basic temporal aggregation techniques (compared to temporal attention models) like average pooling or max pooling generally weaken or over emphasize the contribution of discriminative features (regardless of whether the pooling is applied per-frame, or per-region). In our experiments, we compare our proposed per-region-per-frame temporal attention model to average and maximum pooling applied on a per-region basis, and indeed find that maximum performance is achieved with our temporal attention model.

Similar to spatial attention, we define the temporal attention $t_{n,k}$ for the spatial component k in frame n to be the softmax of a linear response function

$$e_{n,k} = (\mathbf{w}_{t,k})^{\mathsf{T}} \widehat{\mathbf{x}}_{n,k} + b_{t,k}, \tag{4.11}$$

where $\widehat{\mathbf{x}}_{n,k} \in \mathbb{R}^D$ is the enhanced feature of the k^{th} spatial component in the n^{th} frame, and $\mathbf{w}_{t,k} \in \mathbb{R}^D$ and $b_{t,k}$ are parameters to be learned.

$$t_{n,k} = \frac{e_{n,k}}{\sum_{j=1}^{N} e_{j,k}}. \tag{4.12}$$

The temporal attentions are then used to gate the enhanced spatial features on a per component basis by weighted averaging

$$\mathbf{x}_k = \sum_{n=1}^{N} t_{n,k} \widehat{\mathbf{x}}_{n,k}. \tag{4.13}$$

Combining (4.3), (4.4) and (4.13) summarizes how we apply attention on a spatial then temporal basis to extract and align portions of each raw feature $\mathbf{f}_{n,\ell}$ and then aggregate across time to produce a latent representation of each distinctive object region/part

$$\mathbf{x}_k = \sum_{n=1}^{N} t_{n,k} E\left(\sum_{\ell=1}^{L} s_{n,k,\ell} \mathbf{f}_{n,\ell}\right). \tag{4.14}$$

Finally, the entire input video is represented by a feature vector $\mathbf{x} \in \mathbb{R}^{K \times D}$ generated by concatenating the temporally gated features of each spatial component

$$\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_K]. \tag{4.15}$$

4.5.1 Re-Identification Loss

In this paper, we adopt the Online Instance Matching loss function (OIM) [28] to train the whole network. Typically, person search uses a multi-class softmax layer as the objective loss. Often, the number of mini-batch samples is much smaller than the number of identities in the training dataset, and network parameter updates can be biased. Instead, the OIM loss function uses a lookup table to store features of all identities appearing in the training set. In each forward iteration, a mini-batch sample is compared against all the identities when computing classification probabilities. This loss function has shown to be more effective than softmax when training person search networks.

4.6 Experiments

4.6.1 Datasets

We evaluate the proposed algorithm on three commonly used video-based person search datasets: PRID2011 [58], iLIDS-VID [59], and MARS [60]. PRID2011 consists of person videos from two camera views, containing 385 and 749 identities, respectively. Only the

first 200 people appear in both cameras. The length of each image sequence varies from 5 to 675 frames. iLIDS-VID consists of 600 image sequences of 300 subjects. For each person we have two videos with the sequence length ranging from 23 to 192 frames with an average duration of 73 frames. The MARS dataset is the largest video-based person search benchmark with 1,261 identities and around 20,000 video sequences generated by DPM detector [61] and GMMCP tracker [62]. Each identity is captured by at least 2 cameras and has 13.2 sequences on average. There are 3,248 distractor sequences in the dataset.

For PRID2011 and iLIDS-VID datasets, we follow the evaluation protocol from [59]. Datasets are randomly split into probe/gallery identities. This procedure is repeated 10 times for computing averaged accuracies. For the MARS dataset, we follow the original splits provided by [60] which use the predefined 631 identities for training and the remaining identities for testing.

4.6.2 Implementation details and evaluation metrics

We divide each input video sequence into $N = 6$ chunks of equal duration. We first pretrain the ResNet-50 model on image-based person search datasets, including CUHK01 [54], CUHK03 [19], 3DPeS [52], VIPeR [49], DukeMTMC-reID [57] and CUHK-SYSU [28]. and then fine-tune it on PRID2011, iLIDS-VID and MARS training sets. Once finished, we fix the CNN model and train the set of multiple spatial attention models with average temporal pooling and OIM loss function. Finally, the whole network, except the CNN model, is trained jointly. The input image is resized to 256×128. The network is updated using batched Stochastic Gradient Descent with an initial learning rate set to 0.1 and then dropped to 0.01. The aggregated feature vector after the last FC layer is embeded into 128-dimensions and L2-normalized to represent each video sequence. During the training stage, we utilize the Restricted Random Sampling to select training samples. For each video, we extract its L2-normalized feature and sent it to the OIM loss function to supervise the training process. During testing, we use the first image from each of N segments as a testing sample and its L2-normalized features are utilized to compute the similarity of the spatiotemporal gated features generated for the pair of videos being assessed.

Person search performance is reported using the rank-1 accuracy. On the MARS

Method	PRID2011	iLIDS-VID	MARS
Baseline	82.7	61.2	73.4 (58.1)
SpaAtn	84.2	64.9	74.5 (59.3)
SpaAtn+Q′	86.5	64.5	74.0 (58.2)
SpaAtn+Q	86.7	68.6	77.0 (60.9)
SpaAtn+Q+MaxPool	86.9	68.2	76.8 (60.5)
SpaAtn+Q+TemAtn	88.4	69.7	77.1 (61.2)
SpaAtn+Q+TemAtn+Ind	**93.2**	**80.2**	**82.3** (**65.8**)

Table 4.1: *Component analysis of the proposed method: rank-1 accuracies are reported. For MARS we provide mAP in brackets.* ***SpaAtn*** *is the multi-region spatial attention,* ***Q′*** *and* ***Q*** *are two regularization terms,* ***MaxPool*** *and* ***TemAtn*** *are max temporal pooling and the proposed temporal attention respectively.* ***Ind*** *represents fine-tuning the whole network to each dataset independently.*

dataset we also evaluate the mean average precision (mAP) [60]. Since mAP takes recall into consideration, it is more suitable for the MARS dataset which has multiple videos per identity.

4.6.3 Component Analysis of the Proposed Model

We investigate the effect of each component of our model by conducting several analytic experiments. In Tab. 4.1, we list the results of each component in the proposed network. **Baseline** corresponds to ResNet-50 trained with OIM loss on image-based person search datasets and then jointly fine-tuned on video datasets: PRID2011, iLIDS-VID, and MARS. **SpaAtn** consists of the subnetwork of ResNet-50 (from $res2x$ to $res5x$) and multiple spatial attention models. All spatial gated features generated by the same attention model are grouped together and averaged over all frames. For each video sequence, there will be K averaged feature vectors. We concatenate the K features and then send them to the last FC layer and OIM loss function to train the neural network. Compared with **Baseline**, **SpaAtn** improves the rank-1 accuracy by 1.5%, 3.7%, and 1.1% on PRID2011, iLIDS-VID and MARS, respectively. This shows that multiple spatial attention models are effective at finding persistent discriminative image regions which are useful for boosting the person search performance.

SpaAtn+Q' has the same network architecture as **SpaAtn** but with the text embedding diversity regularization term Q' [80]. **SpaAtn+Q** uses our proposed diversity regularization term Q based on Hellinger distance. From the results, we can

K	PRID2011	iLIDS-VID	MARS
1	86.2	64.7	76.0
2	83.4	64.6	75.7
4	86.9	64.6	**77.2**
6	**88.4**	**69.7**	77.1
8	88.0	66.9	76.7

Table 4.2: *The rank-1 accuracy using different number K of diverse spatial attention models.*

see that our proposed Hellinger regularization improves accuracy. We believe the improvement comes from being able to learn multiple attention models with sufficiently large (but minimally overlapping) receptive fields (see Fig.4.3 for sample receptive fields generated for the learned attention models using **SpaAtn+Q**). **SpaAtn+Q** and **SpaAtn+Q+MaxPool** are strategies for average temporal pooling and maximum temporal pooling, respectively. **SpaAtn+Q+TemAtn** applies multiple temporal attentions to each frame—one for each diverse spatial attention model. The assigned temporal attention weights reflect the pertinence of each spatially attended region (*e.g.* is the part fully visible and easy to detect?). We finally fine-tune the whole network, including the CNN model, to each video dataset independently. **SpaAtn+Q+TemAtn+Ind** is the final result of our proposed framework.

Different number of spatial attention models: We also carry out experiments to investigate the effect of varying the number K of spatial attention models (Tab. 4.2). When $K = 1$, the framework is limited to a single spatial attention model, which tends to cover the whole body. As K is increased, the network is able to discover a larger set of body parts, and since the receptive fields are regularized to have minimal overlap, the reception fields tend to shrink as K gets bigger. Interestingly, there is a general drop in perform when K is increased from 1 to 2. This implies treating a person as a single region instead of two distinct body parts is better. However, when a sufficiently large $K = 6$ number of spatial models is used, the network achieves maximum performance.

Example learned spatial attention models and corresponding receptive fields are shown in Fig. 4.3. The receptive fields generally correspond to specific body parts and have varying sizes dependent on the discovered concept. In constrast, the receptive fields generated by [75] tend to include background clutter and exhibit substantial overlap between different attention models. Our receptive fields, on the other hand,

Method	PRID2011	iLIDS-VID	MARS
STA [82]	64.1	44.3	-
DVDL [83]	40.6	25.9	-
TDL [69]	56.7	56.3	-
SI2DL [70]	76.7	48.7	-
mvRMLLC+Alignment [84]	66.8	69.1	-
AMOC+EpicFlow [76]	82.0	65.5	-
RNN [35]	70.0	58.0	-
IDE [85] + XQDA [6]	-	-	65.3 (47.6)
GEI+Kissme [60]	19.0	10.3	1.2 (0.4)
end AMOC+EpicFlow [76]	83.7	68.7	68.3 (52.9)
Mars [60]	77.3	53.0	68.3 (49.3)
SeeForest [36]	79.4	55.2	70.6 (50.7)
QAN [86]	90.3	68.0	-
PAM-LOMO+KISSME [87]	92.5	79.5	-
Ours	**93.2**	**80.2**	**82.3** (**65.8**)

Table 4.3: *Comparisons of our proposed approach to the state-of-the-art on PRID2011, iLIDS-VID, and MARS datasets. The rank-1 accuracies are reported and for MARS we provide mAP in brackets.*

have minimal overlap and focus primarily on the foreground regions.

4.6.4 Comparison with the State-of-the-art Methods

Table 4.3 reports the performance of our approach with other state-of-the-art techniques. On each dataset, our method attains the highest performance. We achieve maximum improvement on MARS dataset, where we improve the state-of-the-art by 11.7%. The previous best reported results are from PAM-LOMO+KISSME [87] (which learns signature representation to cater for high variance in a person's appearance) and from SeeForest [36] (which combines six spatial RNNs and temporal attention followed by a temporal RNN to encode the input video). In contrast, our network architecture is intuitive and straightforward to train. MARS is the most challenging data (it contains distractor sequences and has a substantially larger gallery set) and our methodology achieves a significant increase in mAP accuracy. This result suggests our spatiotemporal model is very effective for video-based person search in challenging scenarios.

4.7 Conclusions

A key challenge for successful video-based person search is developing a latent feature representation of each video as a basis for making comparisons. In this work, we propose

a new spatiotemporal attention mechanism to achieve better video representations. Instead of extracting a single feature vector per frame, we employ a diverse set of spatial attention models to consistently extract similar local patches across multiple images (Fig. 4.3). This approach automatically solves two common problems in video-based person search: aligning corresponding image patches across frames (because of changes in body pose, orientation relative to the camera, etc.) and determining whether a particular part of the body is occluded or not.

To avoid learning redundant spatial attention models, we employ a diversity regularization term based on Hellinger distance. This encourages the network to discover a set of spatial attention models that have minimal overlap between receptive fields generated for each image. Although diversity regularization is not a new topic, we are the first to learn a diverse set of spatial attention models for video sequences, and illustrate the importance of Hellinger distance for this task (our experiments illustrate how a diversity regularization term used in text embedding is less effective for images).

Finally, temporal attention is used to aggregate features across frames on a per-spatial attention model basis—*e.g.* all features from the facial region are combined. This allows the network to represent each discovered body part based on the most pertinent image regions within the video. We evaluated our proposed approach on three datasets and performed a series of experiments to analyze the effect of each component. Our method outperforms the state-of-the-art approaches by large margins which demonstrates its effectiveness in video-based person search.

Chapter 5

Person Search with Natural Language Description

Searching persons in large-scale image databases with the query of natural language description has important applications in video surveillance. Existing methods mainly focused on searching persons with image-based or attribute-based queries, which have major limitations for a practical usage. In this paper, we study the problem of person search with natural language description. Given the textual description of a person, the algorithm of the person search is required to rank all the samples in the person database then retrieve the most relevant sample corresponding to the queried description. Since there is no person dataset or benchmark with textual description available, we collect a large-scale person description dataset with detailed natural language annotations and person samples from various sources, termed as CUHK Person Description Dataset (CUHK-PEDES). A wide range of possible models and baselines have been evaluated and compared on the person search benchmark. An Recurrent Neural Network with Gated Neural Attention mechanism (GNA-RNN) is proposed to establish the state-of-the art performance on person search.

The contribution of this paper is three-fold. 1) We propose to study the problem of searching persons with natural language. This problem setting is more practical for real-world scenarios. To support this research direction, a large-scale person description dataset with rich language annotations is collected and the user study on the natural language description of person is given. 2) We investigate a wide range of plausible solutions based on different vision and language frameworks, including image captioning [88,89], visual QA [90,91], and visual-semantic embedding [2], and establish baselines on the person search benchmark. 3) We further propose a novel Recurrent Neural Network with Gated Neural Attention (GNA-RNN) for person search, with the

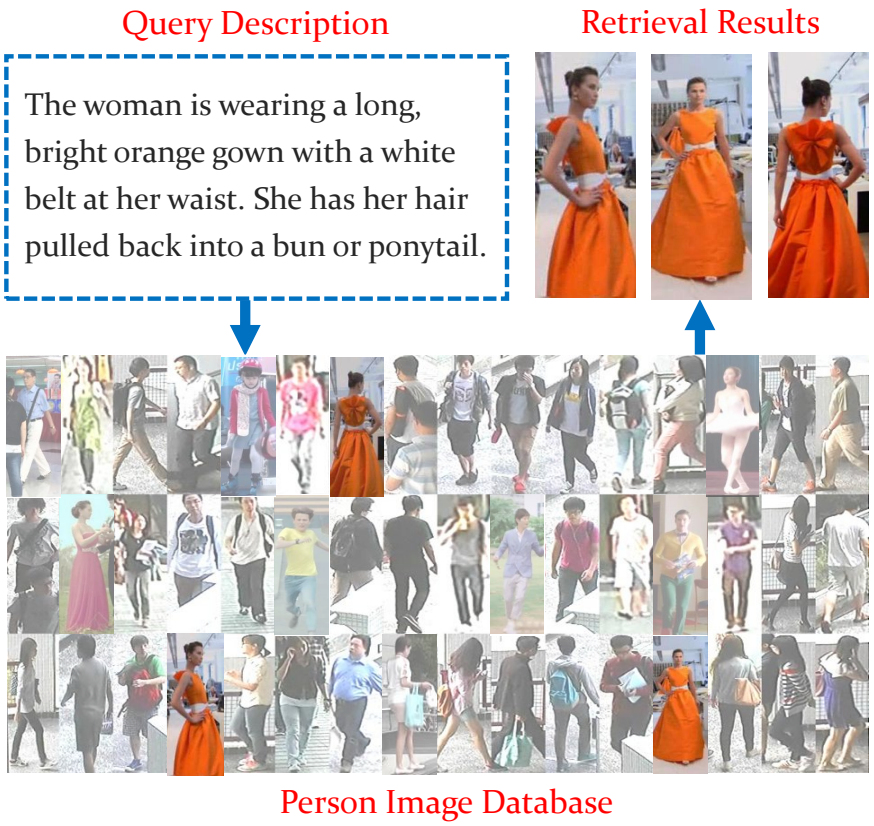


Figure 5.1: *Given the natural language description of a person, our person search system searches through a large-scale person database then retrieve the most relevant person samples.*

state-of-the-art performance on the person search benchmark. The dataset and code are made public to facilitate further research[1] .

5.1 Related Work

Searching person in a database with free-form natural language description is a challenging problem in computer vision. It has wide applications in video surveillance and activity analysis. Nowadays urban areas are usually equipped with thousands of surveillance cameras which generate gigabytes of video data every second. To search possible criminal suspects from such large-scale videos manually might take tens of days or even months to complete. Thus automatic person search is in urgent need. Based on modalities of the queries, existing person search methods can be mainly categorized into the ones with image-based queries and attribute-based queries. However, both modalities have major limitations and might not be suitable for practical usages.

[1] https://github.com/ShuangLI59/Person-Search-with-Natural-Language-Description

Image-based Person Search. Person search with image-based queries is known as person re-identification in computer vision [6, 30, 67]. Given a query image, the algorithms obtain affinities between the query and those in the image database. The most similar persons can be retrieved from the database according to the affinity values. It has many video surveillance applications, such as finding criminals [63], cross-camera person tracking [64], and person activity analysis [51].

Early person re-identification methods addressed the problem by manually designing discriminative features [32, 92, 93], learning feature transforms across camera views [21, 94, 95], and learning distance metrics [21–23, 67, 96]. Recent years, many researchers have proposed various deep learning based methods that jointly handle all these aspects. Li *et al.* [19] and Ahmed *et al.* [20] designed specific CNN models for person re-id. Both the networks utilize as input a pair of cropped pedestrian images and employ a binary verification loss function to train the parameters. Ding *et al.* [31] and Cheng *et al.* [42] exploited triplet samples for training CNNs to minimize the feature distance between the same person and maximize the distance between different people. Apart from using pairwise or triplet loss functions, Xiao *et al.* [40] proposed to learn features by classifying identities. Multiple datasets are combined together and a domain guided dropout technique is proposed to improve the feature learning. Several recent works addressed on solving person re-id on abnormal images, such as low-resolution images [97], or partially occluded images [98].

However, such a problem setting has major limitations in practice, as it requires at least one photo of the queried person being given. In many criminal cases, there might be only verbal description of the suspects' appearance available.

Attribute-based Person Search. Person search could also be done through attribute-based queries. A set of pre-defined semantic attributes are used to describe persons' appearances. Classifiers are then trained on each of the attributes. Given a query, similar persons in the database can be retrieved as the ones with similar attributes [99, 100]. However, the attributes have many practical limitations as well. On the one hand, attributes have limited capability of describing persons' appearance. For instance, the PETA dataset [101] defined 61 binary and 4 multi-class person attributes, while there are hundreds of words for describing a person's appearance. On the other hand, even with the exhausted set of attributes, labeling them for a large-scale person

image dataset is expensive.

Language-based Person Search. Facing the limitations of both modalities, we propose to use natural language description to search person. Figure 5.1 illustrates one example of the person search. It does not require a person photo to be given as in those image-based query methods. Natural language also can precisely describe the details of person appearance, and does not require labelers to go through the whole list of attributes.

As there are no existing datasets and methods designed for the person search with natural language, we briefly survey the language datasets for various vision tasks, along with the deep language models for vision that can be used as possible solutions for this problem.

Early language datasets for vision include Flickr8K [102] and Flickr30K [103]. Inspired by them, Chen *et al.* built a larger MS-COCO Caption [104] dataset. They selected 164,062 images from MS-COCO [105] and labeled each image with five sentences from independent labelers. Recently, Visual Genome [106] dataset was proposed by Krishna *et al.*, which incorporates dense annotations of objects, attributes, and relationships within each image. However, although there are persons in the datasets, they are not the main subjects for descriptions and cannot be used to train person search algorithms with language descriptions. For fine-grained visual descriptions, Reed *et al.* added language annotations to Caltech-UCSD birds [107] and Oxford-102 flowers [108] datasets to describe contents of images for text-image joint embedding.

Different from convolutional neural network which works well in image classification [7, 14] and object detection [109–111], recurrent neural network is more suitable in processing sequential data. A large number of deep models for vision tasks [72, 112–117] have been proposed in recent years. For image captioning, Mao *et al.* [118] learned feature embedding for each word in a sentence, and connected it with the image CNN features by a multi-modal layer to generate image captions. Vinyal *et al.* [89] extracted high-level image features from CNN and fed it into LSTM for estimating the output sequence. The NeuralTalk [88] looked for the latent alignment between segments of sentences and image regions in a joint embedding space for sentence generation.

Visual QA methods were proposed to answer questions about given images [91, 119–123]. Yang *et al.* [120] presented a stacked attention network that refined the

joint features by recursively attending question-related image regions, which leads to better QA accuracy. Noh *et al.* [119] learned a dynamic parameter layer with hashing techniques, which adaptively adjusts image features based on different questions for accurate answer classification.

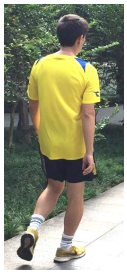
Visual-semantic embedding methods [2,88,124–126] learned to embed both language and images into a common space for image classification and retrieval. Reed *et al.* [2] trained an end-to-end CNN-RNN model which jointly embeds the images and fine-grained visual descriptions into the same feature space for zero-shot learning. Text-to-image retrieval can be conducted by calculating the distances in the embedding space. Frome *et al.* [124] associated semantic knowledge of text with visual objects by constructing a deep visual-semantic model that re-trained the neural language model and visual object recognition model jointly.

5.2 Dataset and Method Overview

Since there is no existing dataset focusing on describing person appearances with natural language, we first build a large-scale language dataset, with 40,206 images of 13,003 persons from existing person re-identification datasets. Each person image is described with two sentences by two independent workers on Amazon Mechanical Turk (AMT). On the visual side, the person images pooled from various image-based person search datasets are under different scenes, view points and camera specifications, which increases the image content diversity. On the language side, the dataset has 80,412 sentence descriptions, containing abundant vocabularies, phrases, and sentence patterns and structures. The labelers have no limitations on the languages for describing the persons. We perform a series of user studies on the dataset to show the rich expression of the language description. Examples from the dataset are shown in Figure 5.2.

We propose a novel Recurrent Neural Network with Gated Neural Attention (GNA-RNN) for person search. The GNA-RNN takes a description sentence and a person image as input and outputs the affinity between them. The sentence is input into a word-LSTM and processed word by word. At each word, the LSTM generates unit-level attentions for individual visual units, each of which determines whether certain person semantic attributes or visual patterns exist in the input image. The visual-unit attention mechanism weights the contributions of different units for different words. In

The woman is dressed up like Marilyn Monroe, with a white dress that is blowing upward in the wind, short curly blonde hair, and high heels.

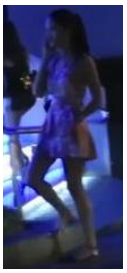


The man is wearing yellow sneakers, white socks with blue stripes on the top of them, black athletic shorts and a yellow with blue t-shirt. He has short black hair.

The man has dark hair and is wearing glasses. He has on a pink shirt, blue shorts, and white tennis shoes. He has on a blue backpack and is carrying a re-useable tote.

The woman has long light brown hair, is wearing a black business suit with white low-cut blouse with large, white cuffs, a gold ring, and is talking on a cellphone.

The girl is wearing a pink shirt with white shorts, she is wearing black converse, with her hair in a pony tail.



Young female with dark hair pulled back in a pony tail. Wearing a light colored thigh-length dress and sandals.

She is wearing black shoes, a short tight black skirt, and a bright blue blouse with long sleeves tucked into her waistband.

The man is wearing blue scrubs with a white lab coat on top. He is holding paperwork in his hand and has a name badge on the left side of his coat.

Figure 5.2: *Example sentence descriptions from our dataset that describe persons' appearances in detail.*

addition, we also learn word-level gates that estimate the importance of different words for adaptive word-level weighting. The final affinity is obtained by averaging over all units' responses at all words. Both the unit-level attention and word-level sigmoid gates contribute to the good performance of our proposed GNA-RNN.

5.3 Benchmark for person search with natural language description

Since there is no existing language dataset focusing on person appearance, we build a large-scale benchmark for person search with natural language, termed as CUHK Person Description Dataset (CUHK-PEDES). We collected 40,206 images of 13,003

persons from five existing image-based person search datasets, CUHK03 [19], Market-1501 [127], SSM [30], VIPER [49], and CUHK01 [53], as the subjects for language descriptions. Since persons in Market-1501 and CUHK03 have many similar samples, to balance the number of persons from different domains, we randomly selected four images for each person in the two datasets. All the image were labeled by crowd workers from Amazon Mechanical Turk (AMT), where each image was annotated with two sentence descriptions and a total of 80,412 sentences were collected. The dataset incorporates rich details about person appearances, actions, poses and interactions with other objects. The sentence descriptions are generally long (> 23 words in average), and has abundant vocabulary and little repetitive information. Examples of our proposed dataset are shown in Figure 5.2.

5.3.1 Dataset statistics

The dataset consists of rich and accurate annotations with open word descriptions. There were 1,993 unique workers involved in the labeling task, and all of them have greater-than 95% approving rates. We asked the workers to describe all important characteristics in the given images using sentences with at least 15 words. The large number of workers means the dataset has diverse language descriptions and methods trained with it are unlikely to overfit to descriptions of just a few workers.

Vocabulary, phrase sizes, and sentence length are important indicators on the capacity our language dataset. There are a total of 1,893,118 words and 9,408 unique words in our dataset. The longest sentence has 96 words and the average word length is 23.5 which is significantly longer than the 5.18 words of MS-COCO Caption [105] and the 10.45 words of Visual Genome [106]. Most sentences have 20 to 40 words in length. Figure 5.3 illustrates some person examples and high-frequency words.

5.3.2 User study

Based on the language annotations we collect, we conduct the user studies to investigate 1) the expressive power of language descriptions compared with that of attributes, 2) the expressive power in terms of the number of sentences and sentence length, and 3) the expressive power of different word types. The studies provide us insights for understanding the new problem and guidance on designing our neural networks.

Figure 5.3: *High-frequency words and person images in our dataset.*

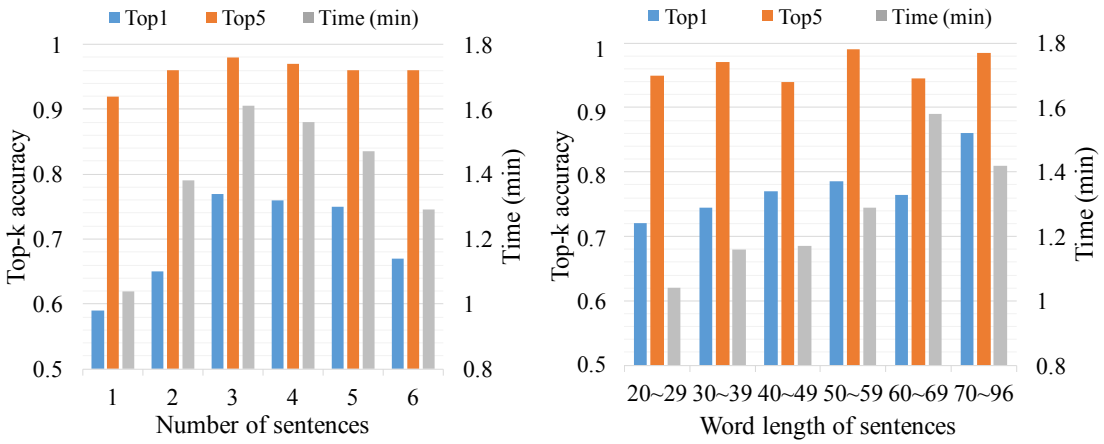


Figure 5.4: *Top-1 accuracy, top-5 accuracy, and average used time of manual person search using language descriptions with different number of sentences and different sentence lengths.*

Language vs. attributes. Given a descriptive sentence or annotated attributes of a query person image, we ask crowd workers from AMT to select its corresponding image from a pool of 20 images. The 20 images consist of the ground truth image, 9 images with similar appearances to the ground truth, and 10 randomly selected images from the whole dataset. The 9 similar images are chosen from the whole dataset

	orig. sent.	w/o nouns	w/o adjs	w/o verbs
top-1	0.59	0.38	0.44	0.57
top-5	0.92	0.81	0.85	0.92
time (min)	1.14	1.01	0.98	1.12

Table 5.1: *Top-1 accuracy, top-5 accuracy, and average used time of manual person search results using the original sentences, and sentences with nouns, or adjectives, or verbs masked out.*

by the LOMO+XQDA [6] method, which is a state-of-the-art method for person re-identification. The other 10 distractor images are randomly selected and have no overlap with the 9 similar images. The person attribute annotations are obtained from the PETA [101] dataset, which have 1,264 same images with our dataset. A total of 500 images are manually searched by the workers, and the average top-1 and top-5 accuracies of the searches are evaluated. The searches with language descriptions have 58.7% top-1 and 92.0% top-5 accuracies, while the searches with attributes have top-1 and top-5 accuracies of 33.3% and 74.7% respectively. In terms of the average used time for each search, using language descriptions takes 62.18s, while using attributes takes 81.84s. The results show that, from human's perspective, language descriptions are much precise and effective in describing persons than attributes. They partially endorse our choice of using language descriptions for person search.

Sentence number and length. We design manual experiments to investigate the expressive power of language descriptions in terms of the number of sentences for each image and sentence length. The images in our dataset are categorized into different groups based on the number of sentences associated with each image and based on different sentence lengths. Given the sentences for each image, we ask crowd workers from AMT to manually retrieve the corresponding images from pools of 20 images. The average top-1 and top-5 accuracies, and used time for different image groups are shown in Figure 5.4, which show that 3 sentences for describing a person achieved the highest retrieval accuracy. The longer the sentences are, the easier for users to retrieve the correct images.

Word types. We also investigate the importance of different word types, including nouns, verbs, and adjectives by using manual experiments with the same 20-image pools. For this study, nouns, or verbs, or adjectives in the sentences are masked out before provided to the workers. For instance, "the girl has pink hair" is converted

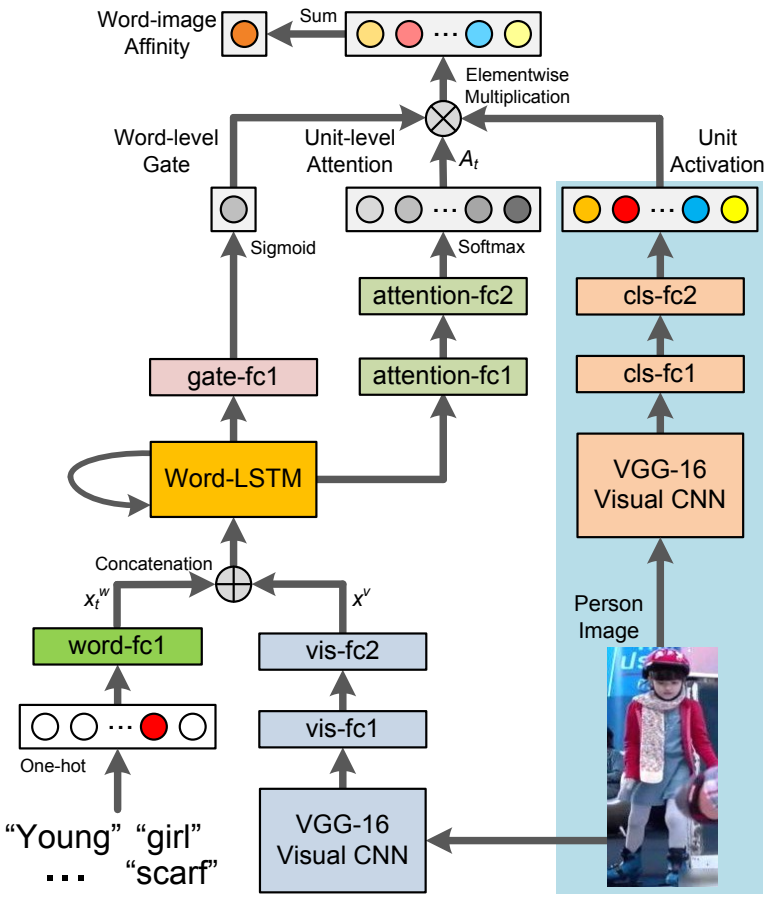


Figure 5.5: *The network structure of the proposed GNA-RNN. It consists of a visual sub-network (right blue branch) and a language sub-network (left branch). The visual sub-network generates a series of visual units, each of which encodes if certain appearance patterns exist in the person image. Given each input word, The language sub-network outputs world-level gates and unit-level attentions for weighting visual units.*

to "the **** has pink ****", where the nouns are masked out. Results in Table 5.1 demonstrate that the nouns provide most information followed by the adjectives, while the verbs carry least information. This investigation provides us important insights that nouns and adjectives should be paid much attention to when we design neural networks or collecting new language data.

5.4 GNA-RNN model for pedestrian search

The key to address the person search with language description is to effectively build word-image relations. Given each word, it is desirable if the neural network would search related regions to determine whether the word with its context fit the image. For a sentence, all such word-image relations can be investigated, and confidences of all

relations should be weighted and then aggregated to generate the final sentence-image affinity.

Based on this idea, we propose a novel deep neural network with Gated Neural Attention (GNA-RNN) to capture word-image relations and estimate the affinity between a sentence and a person image. The overall structure of the GNA-RNN is shown in Figure 5.5. The network model consists of a visual sub-network and a language sub-network. The visual sub-network generates a series of visual unit activations, each of which encodes if certain human attributes or appearance patterns (*e.g.*, white scarf) exist in the given person image. The language sub-network is a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) units, which takes words and images as input. At each word, it outputs unit-level attention and word-level gate to weight the visual units from the visual sub-network. The unit-level attention determines which visual units should be paid more attention to according to the input word. The word-level gate weight the importance of different words. All units' activations are weighted by both the unit-level attentions and word-level gates, and are then aggregated to generate the final affinity. By training such network in an end-to-end manner, the Gated Neural Attention mechanism is able to effectively capture the optimal word-image relations.

5.4.1 Visual units

The visual sub-network takes person images that are resized to 256×256 as inputs. It has the same bottom structure as VGG-16 network, and adds two 512-unit fully-connected layers at the "drop7" layer to generate 512 visual units, $\mathbf{v} = [v_1, ..., v_{512}]^T$. Our goal is to train the whole network jointly such that each visual unit determines whether certain human appearance pattern exist in the person image. The visual sub-network is first pre-trained on our dataset for person classification based on person IDs. During the joint training with language sub-network, only parameters of the two new fully-connected layers ("cls-fc1" and "cls-fc2" in Figure 5.5) are updated for more efficient training. Note that we do not manually constrain which units learn what concepts. The semantic meanings of the visual units automatically capture necessary semantic concepts via jointly training of the whole network.

5.4.2 Attention over visual units

To effectively capture the word-image relations, we propose a unit-level attention mechanism for visual units. At each word, the visual units having similar semantic meanings with the word should be assigned with more weights. Take Figure 5.5 as example, given the words "white scarf", the language sub-network would attend more the visual unit that corresponds to the concept of "white scarf". We train the language sub-network to to achieve this goal.

The language sub-network is a LSTM network [11], which is effective at capturing temporal relations of sequential data. Given an input sentence, the LSTM generates attentions for visual units word by word. The words are first encoded into length-K one-hot vectors, where K is the vocabulary size. Given a descriptive sentence, a learnable fully connected layer ("word-fc1" in Figure 5.5) converts the tth raw word to a word embedding feature x_w^t. Two 512-unit fully connected layers ("vis-fc1" and "vis-fc2" in Figure 5.5) following the "drop7" layer of VGG-16 are treated as visual features x_v for the LSTM. At each step, the LSTM takes $x_t = [x_t^w, x^v]^T$ as input, which is concatenation of tth word embedding x_t^w and image features x^v.

The LSTM consists of a memory cell c_t and three controlling gates, *i.e.* input gate i_t, forget gate f_t, and output gate o_t. The memory cell preserves the knowledge of previous step and current input while the gates control the update and flow direction of information. At each word, the LSTM updates the memory cell c_t and output a hidden state h_t in the following way,

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \\
c_t &= f_t \odot c_{t-1} + i_t \odot h(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \\
h_t &= o_t \odot h(c_t),
\end{aligned}
\tag{5.1}
$$

where $\odot$ represents the element-wise multiplication, W and b are parameters to learn.

For generating the unit-level attentions at each word, the output hidden state h_t is fed into a fully-connected layer with ReLU non-linearity function and a fully-connected

layer with softmax function to obtain the attention vector $A_t \in \mathbb{R}^{512}$, which has the same dimension as the visual units $\mathbf{v}$. The affinity between the sentence and the person image at the tth word can then be obtained by

$$a_t = \sum_{n=1}^{512} A_t(n) v_n, \qquad \text{s.t.} \sum_{n=1}^{512} A_t(n) = 1, \tag{5.2}$$

where $A_t(n)$ denotes the attention value for the nth visual unit. Since each visual unit determines the existence of certain person appearance patterns in the image, the visual units alone cannot generate sentence-image affinity. The attention values A_t generated by the language sub-network decides which visual units' responses should be summed up to compute the affinity value. If the language sub-network generates high attention value at certain visual unit, only if the visual unit also has high response, which denotes existence of certain visual concepts, will the elementwise multiplication generates high affinity value at this word. The final sentence-image affinity is summation of affinity values at all words, $a = \sum_{t=1}^{T} a_t$, where T is the number of words in the given sentence.

5.4.3 Word-level gates for visual units

The unit-level attention is able to associate the most related units to each word. However, the attention mechanism requires different units' attentions competing with each other. In our case with the softmax non-linearity function, we have $\sum_{n=1}^{512} A_t(n) = 1$, and found that such constraints are important for learning effective attentions.

However, according to our user study on different word types in Section 5.3.2, different words carry significantly different amount of information for obtaining language-image affinity. For instance, the word "white" should be more important than the word "this". At each word, the unit-level attentions always sum up to 1 and cannot reflect such differences. Therefore, we propose to learn world-level scalar gates at each word for learning to weight different words. The word-level scalar gate is obtained by mapping the hidden state h_t of the LSTM via a fully-connected layer with sigmoid non-linearity function $g_t = \sigma(W_g h_t + b_g)$, where σ denotes the sigmoid function, and W_g and b_g are the learnable parameters of the fully-connected layer.

Both the unit-level attention and world-level gate are used to weight the visual units

at each word to obtain the per-word language-image affinity $\hat{a}_t$,

$$\hat{a}_t = g_t \sum_{n=1}^{512} A_t(n) v_n, \tag{5.3}$$

and the final affinity is the aggregation of affinities at all words $\hat{a} = \sum_{t=1}^{T} \hat{a}_t$.

5.4.4 Training scheme

The proposed GNA-RNN is trained end-to-end with batched Stochastic Gradient Descent, except for the VGG-16 part of the visual sub-network, which is pre-trained for person classification and fixed afterwards. The training samples are randomly chosen from the dataset with corresponding sentence-image pairs as positive samples and non-corresponding pairs as negative samples. The ratio between positive and negative samples is 1:3. Given the training samples, the training minimizes the cross-entropy loss,

$$E = -\frac{1}{N} \sum_{i=1}^{N} \left[y^i \log \hat{a}^i + (1 - y^i) \log(1 - \hat{a}^i) \right] \tag{5.4}$$

where $\hat{a}^i$ denotes the predicted affinity for the ith sample, and y^i denotes its ground truth label, with 1 representing corresponding sentence-image pairs and 0 representing non-corresponding ones. We use 128 sentence-image pairs for each training batch. All fully connected layers except for the one for word-level gates have 512 units.

5.5 Experiments

There is no existing method specifically designed for the problem. We investigate a wide range of possible solutions based on state-of-the-art language models for vision tasks, and compare those solutions with our proposed method. We also conduct component analysis of our proposed deep neural networks to show that our proposed Gated Neural Attention mechanism is able to capture complex word-image relations. Extensive experiments and comparisons with state-of-the-art methods demonstrate the effectiveness of our GNA-RNN for this problem.

5.5.1 Dataset and evaluation metrics

The dataset is splitted into three subsets for training, validation, and test without having overlaps with same person IDs. The training set consists of 11,003 persons, 34,054 images and 68,108 sentence descriptions. The validation set and test set contain 3,078 and 3,074 images, respectively, and both of them have 1,000 persons. All experiments are performed based on this train-test split.

We adopt the top-k accuracy to evaluate the performance of person retrieval. Given a query sentence, all test images are ranked according to their affinities with the query. A successful search is achieved if any image of the corresponding person is among the top-k images. Top-1 and top-10 accuracies are reported for all our experiments.

5.5.2 Compared methods and baselines

We compare a wide range of possible solutions with deep neural networks, including methods for image captioning, visual QA, and visual-semantic embedding. Generally, each type of methods utilize different supervisions for training. Image captioning, visual QA, and visual-semantic embedding methods are trained with word classification losses, answer classification losses, and distance-based losses, respectively. We also propose several baselines to investigate the influences of detailed network structure design. To make fair comparisons, the image features for all compared methods are from our VGG-16 network pre-trained model.

Image captioning. Vinyals *et al.* [89] and Karpathy *et al.* [88] proposed to generate natural sentences describing an image using deep recurrent frameworks. We use the code provided by Karpathy *et al.* to train the image captioning model. We follow the testing strategy in [128] to use image captioning method for text-to-image retrieval. During the test phase, given a person image, instead of recursively using the predicted word as inputs of the next time step to predict the image caption, the LSTM takes the given sentence word by word as inputs. It calculates the per-word cross entropy losses between the given word and the predicted word from LSTM. Corresponding sentence-image pairs would have low average losses, while non-corresponding ones would have higher average losses.

Visual QA. Agrawal *et al.* [112] proposed the deeper LSTM Q + norm I method to answer questions about the given image. We replace the element-wise multiplication

	NeuralTalk	CNN-RNN	EmbBoW	QAWord	QAWord-img	QABoW	GNA-RNN
top-1	13.66	8.07	8.38	11.62	10.21	8.00	**19.05**
top-10	41.72	32.47	30.76	42.42	44.53	30.56	**53.64**

Table 5.2: *Quantitative results of the proposed GNA-RNN and compared methods on the proposed dataset.*

	GNA-RNN	w/o pre-train	w/o gates	w/o attention
top-1	**19.05**	8.93	13.86	4.85
top-10	**53.64**	32.32	44.27	27.16

Table 5.3: *Quantitative results of GNA-RNN on the proposed dataset without VGG-16 pre-training, without world-level gates or without unit-level attentions.*

between the question and image features, with concatenation of question and image features, and replace the multi-class classifier with a binary classifier. Since the proposed GNA-RNN has only one layer for the LSTM, we change the LSTM in deeper LSTM Q + norm I to one layer as well for fair comparison. The norm I in [112] is also changed to contain two additional fully-connected layers to obtain image features instead of the original one layer following our model's structure. We call the modified model QAWord. Where to concatenate features of question and image modalities might also influence the classification performance. The QAWord model concatenates image features with sentence features output by the LSTM. We investigate concatenating the word embedding features and image features before inputting them into the LSTM. Such a modified network is called QAWord-img. We also replace the language model in QAWord with the simple language model in [90], which encodes sentences using the traditional Bag-of-Word (BoW) method, and call it QABoW.

Visual-semantic embedding. These methods try to map image and sentence features into a joint embedding space. Distances between image and sentence features in the joint space could then be interpreted as the affinities between them. Distances between corresponding sentence-image pairs should be small, and should be high between non-corresponding paris. Reed *et al.* [2] presented a CNN-RNN for zero-shot text-to-image retrieval. We utilize their code and compare it with our proposed framework. We also investigate replacing the language model in CNN-RNN with the simple BoW language model [90] for sentence encoding and denote it as EmbBoW.

# units	128	256	512	1024	2048
top-1	16.15	16.75	19.05	18.62	18.25
top-10	48.58	49.25	53.64	52.39	51.59

Table 5.4*: Top-1 and top-10 accuracies of GNA-RNN with different number of visual units.*

5.5.3 Quantitative and qualitative results

Quantitative evaluation. Table 5.2 shows the results of our proposed framework and the compared methods. We use a single sentence as query to do the person search. Our approach achieves the best performance in terms of both top-1 and top-10 accuracies and outperforms other methods by a large margin. It demonstrates that our proposed network can better capture complex word-image relations than the compared ones.

For all the baselines, the image captioning method NeuralTalk outperforms the other baselines. It calculates the average loss at each word as the sentence-image affinity, and obtains better results than visual QA and visual embedding approaches, which encode the entire sentence into a feature vector. Such results show that the LSTM might have difficulty encoding complex person descriptive sentences into a single feature vector. Word-by-word processing and comparison might be more suitable for the person search problem. We also observe that QAWord-img and QAWord has similar performance. This demonstrates that, the modality fusion between image and word before or after LSTM has little impact on the person search performance. Both ways capture word-image relations to some extent. For the visual-semantic embedding method, the CNN-RNN does not perform well in terms of top-k accuracies with the provided code. The distance-based losses might not be suitable for learning good models for the person search problem. EmbBoW and QABoW use the traditional Bag-of-Word method to encode sentences and have worse performances than their counterparts with RNN language models, which show that the RNN framework is more suitable in processing natural language data.

Component analysis. We pre-train the visual VGG model for image-based person search task first, and then fine-tune whole network for text-to-person search. Without the pre-training, top-1 and top-10 accuracies drop apparently as shown in Table 5.3. This means the initial training affects the final performance a lot. To investigate the effectiveness of the proposed unit-level attentions and word-level gates, we design two

baselines for comparison. For the first baseline (denoted as "w/o gates"), we remove the word-level gates and only keep the unit-level attentions. In this case, different words are equally weighted in estimating the sentence-image affinity. For the second baseline (denoted as "w/o attention"), we try to keep the world-level gates, and replace the unit-level attentions with average pooling over units. We list top-1 and top-10 accuracies of the two baselines in Table 5.3. Both the unit-level attention and word-level gates are important for achieving good performance by our GNA-RNN.

Investigation on the impact of the number of visual units. Results of different number of visual units are listed in Table 5.4. Models with more visual units might over-fit the dataset. 512 units achieves the best result.

Qualitative evaluation. We conduct qualitative evaluation for our proposed GNA-RNN. Figure 5.6 and Figure 5.7 shows 6 person search results with natural language descriptions by our proposed GNA-RNN.

The four cases in Figure 5.6 show successful cases where corresponding images are within the top-6 retrieval results. For the successful cases, we can observe that each top image has multiple regions that fit parts of the descriptions. Some non-corresponding images also show correlations to the query sentences.

In terms of failure cases, there are two types of them. The first type of failure searches do retrieve images that are similar to the language descriptions, however, the exact corresponding images are not within the top retrieval results. For instance, the second case in Figure 5.7 does include persons (top-2, top-3, and top-4) similar to the descriptions, who all wear white tops and red shorts/skirts. Other persons have some characteristics that partially fits the descriptions. The top-1 person has a "hand bag". The top-4 person wears "white top", and the top-6 person carries a "red bag". The second type of failure cases show that the GNA-RNN fails to understand the whole sentence but only captures separate words or phrases. Take the first case in Figure 5.7 as an example, the phrase "brown hair" is not encoded correctly. Instead, only the word "brown" is captured, which leads to the "brown" suit for the top-1 and top-6 persons, and "brown" land in the top-2 image. We also found some rare words/concepts or detailed descriptions are difficult to learn and to locate, such as "ring", "bracelet", "cell phones", *etc.*, which might be learned if more data is provided in the future.

Visual unit visualization. We also inspect the learned visual units to see whether

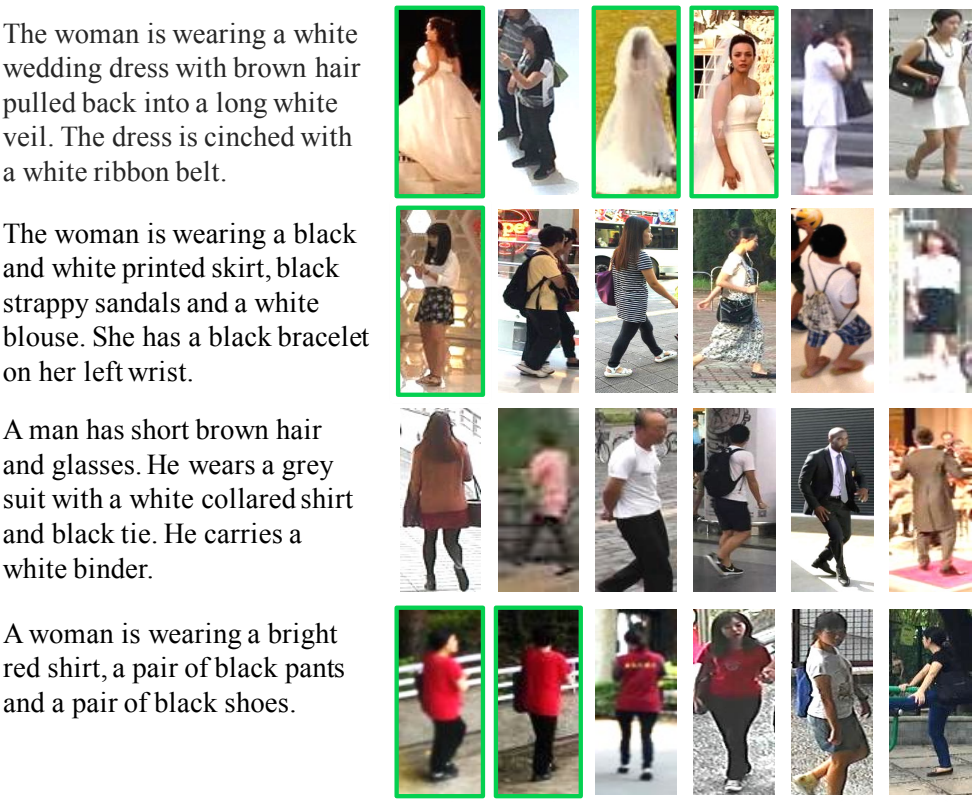


Figure 5.6: *Examples of top-6 person search results with natural language description by our proposed GNA-RNN. Corresponding images are marked by green rectangles. Successful searches where corresponding persons are in the top-6 results.*

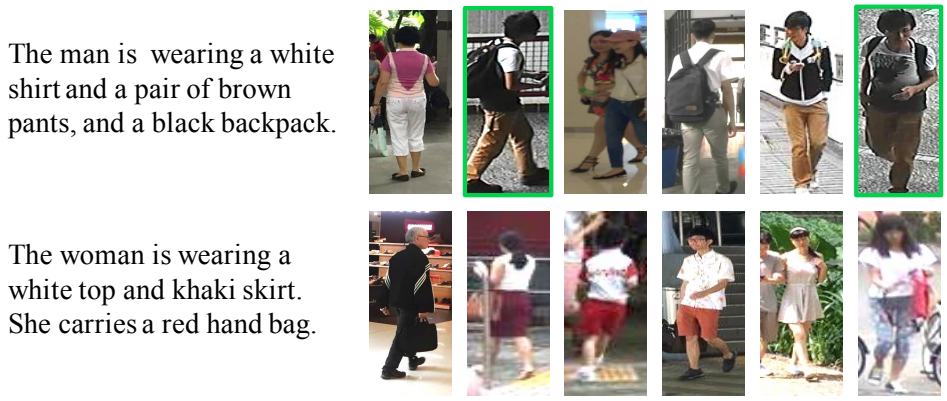


Figure 5.7: *Examples of top-6 person search results with natural language description by our proposed GNA-RNN. Corresponding images are marked by green rectangles. Failure cases where corresponding persons are not in the top-6 results.*

they implicitly capture common visual patterns in person images. We choose some frequent adjectives and nouns. For each frequent word, we collect its unit-level attention vectors for a large number of training images. Such unit-level attention vectors are averaged to identify its most attended visual units. For each of such units, we retrieve

Figure 5.8: *Images with the highest activations on 4 different visual units. The 4 units are identified as the one with the maximum average attention values in our GNA-RNN with the same word ("backpack", "sleeveless", "pink", "yellow") and a large number of images. Each unit determines the existence of some common visual patterns.*

the training images that have the highest responses on the units. Some examples of the visual units obtained in this way are shown in Figure 5.8. Each of them captures some common image patterns.

5.6 Conclusions

In this chapter, we studied the problem of person search with natural languages. We collected a large-scale person dataset with 80,412 sentence descriptions of 13,003 persons. Various baselines are evaluated and compared on the benchmark. A GNA-RNN model was proposed to learn affinities between sentences and person images with the proposed gated neural attention mechanism, which established the state-of-the art performance on person search.

Chapter 6

From Natural Language based Person Search to Textual-Visual Matching

Textual-visual matching aims at measuring similarities between sentence descriptions and images. Natural language based person search is a branch or textual-visual matching. A typical feature of the natural language based person search is that it contains identity-level annotations, *i.e.* each person may have multiple images and sentence descriptions. However, most existing methods tackle this problem without effectively utilizing identity-level annotations.

In this paper, we propose an identity-aware two-stage framework for the textual-visual matching problem. We take the natural language based person search as an example to introduce the proposed method. Our stage-1 CNN-LSTM network learns to embed cross-modal features with a novel Cross-Modal Cross-Entropy (CMCE) loss. The stage-1 network is able to efficiently screen easy incorrect matchings and also provide initial training point for the stage-2 training. The stage-2 CNN-LSTM network refines the matching results with a latent co-attention mechanism. The spatial attention relates each word with corresponding image regions while the latent semantic attention aligns different sentence structures to make the matching results more robust to sentence structure variations. Extensive experiments on three datasets with identity-level annotations show that our framework outperforms state-of-the-art approaches by large margins.

The contribution of this paper is three-fold. 1) We propose a novel identity-aware two-stage deep learning framework for solving the problem of textual-visual matching. The stage-1 network can efficiently screen easy incorrect matchings and also acts as the initial point for training stage-2 network. The stage-2 network refines matching results with binary classification. Identity-level annotations ignored by most existing

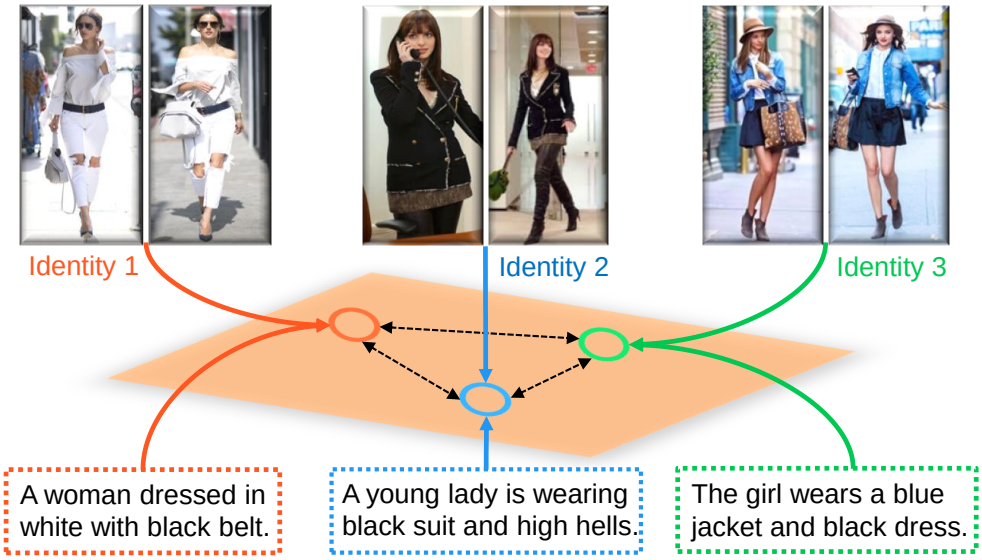


Figure 6.1: *Learning deep features for textual-visual matching with identity-level annotations. Utilizing identity-level annotations could jointly minimize intra-identity discrepancy and maximize inter-identity discrepancy, and thus results in more discriminative feature representations.*

methods are utilized to learn better feature representations. 2) To take advantage of the identity-level annotations, our stage-1 network employs a specialized CMCE loss with feature buffers. Such a loss enables effective feature embedding and fast evaluation. 3) A novel latent co-attention mechanism is proposed for our stage-2 network. It has a spatial attention module that focuses on relevant image regions for each input word, and a latent semantic attention module that automatically aligns different words' feature representations to minimize the impact of sentence structure variations.

6.1 Related Work

Identifying correspondences and measuring similarities between natural language descriptions and images is an important task in computer vision and has many applications, including text-image embedding [2, 129–131], zero-shot learning [124, 132, 133], and visual QA [123, 134–137]. We call such a general problem *textual-visual matching*, which has drawn increasing attention in recent years. The task is challenging because the complex relations between language descriptions and image appearance are highly non-linear and there exist large variations or subtle variations in image appearance for similar language descriptions.

There have been large scale image-language datasets and deep learning techniques [102,

103, 105, 106, 134] proposed for textual-visual matching, which considerably advanced research progress along this direction. However, identity-level annotations provided in benchmark datasets are ignored by most existing methods when performing matching across textual and visual domains.

Visual matching with identity-level annotations. Visual matching tasks with identity-level annotations, such as person re-identification [6, 20, 28, 28, 47, 85, 138] and face recognition [41, 43], are well-developed research areas. Visual matching algorithms either classify all the identities simultaneously [6, 39, 40] or learn pair-wise or triplet distance loss function [20, 41–43] for feature embedding. However, both of them have major limitations. The first type of loss function faces challenges when the number of classes is too large. The limited number of classes (identities) in each mini-batch leads to unstable training behavior. For the second type of loss function, the hard negative training samples might be difficult to sample as the number of training sample increases, and the computation time of constructing pairs or triplets increases quadratically or cubically with the number of test samples.

Textual-visual matching. Measuring similarities between images and languages aims at understanding the relations between images and language descriptions. It gains a lot of attention in recent years because of its wide applications in image captioning [88, 89, 118, 139], visual QA [134–137], and text-image embedding [2, 124, 130, 131, 140]. Karpathy *et al.* [88] combined the convolutional neural network for image regions and bidirectional recurrent neural networks for descriptions to generate image captions. The word-image pairwise affinities are calculated for sentence-image ranking. Nam *et al.* [136] jointly learned image and language attention models to capture the shared concepts between the two domains and evaluated the affinity by computing the inner product of two fixed embedding vectors. [130] tackled the matching problem with deep canonical correlation analysis by constructing the trace norm objective between image and language features. In [140], Klein *et al.* presented two mixture models, Laplacian mixture model and Hybird Gaussian-Laplacian mixture model to learn Fisher vector representations of sentences. The text-to-image matching is conducted by associating the generated Fisher vector and VGG image features.

Identity-aware textual-visual matching. Although identity-level annotations are widely used in visual matching tasks, they are seldom exploited for textual-visual

matching. Using such annotations can assist cross-domain feature embedding by minimizing the intra-identity distances and capturing the relations between textual concepts and visual regions, which makes textual-visual matching methods more robust to variations within each domain.

Reed *et al.* [2] collected fine-grained language descriptions for two visual datasets, Caltech-UCSD birds (CUB) and Oxford-102 Flowers, and first used the identity-level annotations for text-image feature learning. In [1], Li *et al.* proposed a large scale person search dataset with language descriptions and performed description-person image matching using an CNN-LSTM network with neural attention mechanism. However, these approaches face the same problems with existing visual matching methods. To solve these problems and efficiently learn textual and visual feature representations, we propose a novel two-stage framework for identity-aware textual-visual matching. Our approach outperforms both above state-of-the-art methods by large margins on the three datasets.

6.2 Method Overview

Textual-visual matching aims at conducting accurate verification for images and language descriptions. However, identity-level annotations provided by many existing textual-visual matching datasets are not effectively exploited for cross-domain feature learning. In this chapter, we propose a two-stage framework for identity-aware textual-visual matching, which consists of two deep neural networks. The stage-1 network learns identity-aware feature representations of images and language descriptions by introducing a Cross-Modal Cross-Entropy (CMCE) loss to effectively utilize identity-level annotations for feature learning (see Figure 6.1). After training, it provides initial matching results and also serves as the initial point for training stage-2 network. The stage-2 deep neural network employs a latent co-attention mechanism that jointly learns the spatial attention and latent semantic attention to match salient image regions and latent semantic concepts for textual-visual affinity estimation.

Our stage-1 network consists of a CNN and a LSTM for learning textual and visual feature representations. The objective is to minimize the feature distances between descriptions and images belonging to the same identities. The stage-1 network utilizes a specialized CMCE loss with dynamic buffers, which implicitly minimizes intra-identity

feature distances and maximize inter-identity feature distances over the entire dataset instead of just small mini-batches. In contrast, for the pairwise or triplet loss functions, the probability of sampling hard negative samples during training decreases quadratically or cubically as the number of training sample increases.

The trained stage-1 network is able to efficiently screen easy incorrect matchings for both training and testing. However, one limitation of the CMCE loss in stage-1 is that the generated textual and visual features are not tightly coupled. A further refinement on stage-1 results is essential for obtaining accurate matching results. Our stage-2 network is a tightly coupled CNN-LSTM network with latent co-attention. It takes a pair of language description and image as input and outputs the textual-visual matching confidence, which is trained with the binary cross-entropy loss.

Conventional RNNs for language encoding have difficulty in remembering the complete sequential information when the input descriptions are too long. It tends to miss important words appearing in the beginning of the sentence. The RNN is also variant to different sentence structures. Sentences describing the same image but with different sentence structures could be represented by features with large differences. For instance, "the girl who has blond hair is wearing a white dress and heels" and "The girl wears heels and a white dress. She has blond hair." Both sentences describe the same person but the first one might focus more on "white dress and heels", and the second one might assign "blond hair" with higher weights. Inspired by the word alignment (attention) technique in neural machine translation [141], a latent co-attention mechanism is proposed for the stage-2 CNN-LSTM network. The visual spatial attention module associates word to its related image regions. The latent semantic attention module aligns different sentence structures with an alignment decoder LSTM. At each step of the LSTM, it learns how to weight different words' features to be more invariant to sentence structure variations.

6.3 Stage-1 CNN-LSTM with CMCE Loss

The structure of stage-1 network is illustrated in Figure 6.2, which is a loosely coupled CNN-LSTM . Given an input textual description or image, both the visual CNN and language LSTM are trained to map the input image and description into a joint feature embedding space, such that the features representations belonging to the same identity

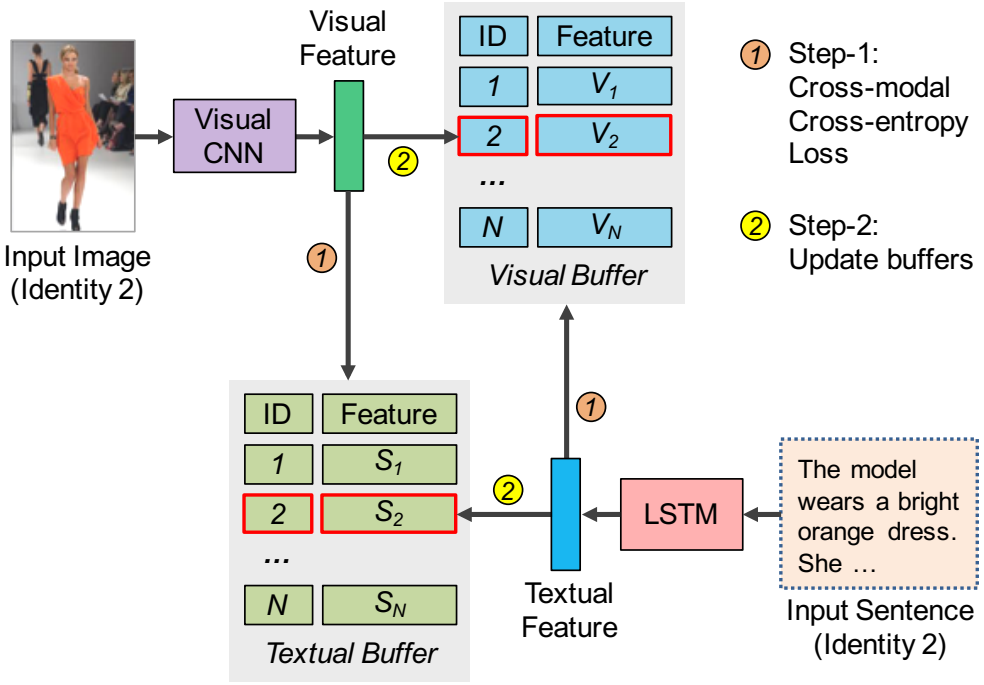


Figure 6.2: *Illustration of the stage-1 network. In each iteration, the images and text descriptions in a mini-batch are first fed into the CNN and LSTM respectively to generate their feature representations. The CMCE loss is then computed by comparing sampled features in one modality to all other features in the feature buffer of the other modality (Step-1). The CNN and LSTM parameters are updated by backpropagation. Finally, the visual and textual feature buffers are updated with the sampled features (Step-2).*

should have small feature distances, while those of different identities should have large distances. To achieve the goal, the CNN-LSTM network is trained with a CMCE loss.

6.3.1 Cross-Modal Cross-Entropy Loss

For the conventional pairwise classification loss [134, 137] or triplet max-margin loss [2, 131], if there are N identities in the training set, the number of possible training samples would be $O(N^2)$. It is generally difficult to sample hard negative samples for learning effective feature representations. On the other hand, during evaluation phase, the time complexity of feature calculation of pairwise or triplet loss would increase quadratically with N, which would take lots of computation time. To solve this problem, we propose a novel CMCE loss that efficiently compares a mini-batch of n identity features from one modality to those of all N identities in another modality in each iteration. Intuitively, the sampled n identity features are required to have high affinities with their corresponding identities in the other modality and low affinities

with all other $N - n$ ones in the entire identity set. The cross-modal affinity is calculated as the inner products of features from the two modalities. By using the proposed loss function, hard negative samples are all covered in each training epoch and the evaluation time complexity of sampling all test samples is only $O(N)$.

In each training iteration, a mini-batch of images belonging to n different identities are transformed to visual features, each of which is denoted by $v \in \mathbb{R}^D$. D is the feature embedding dimension for both modalities. Textual features of all N identities are pre-stored in a textual feature buffer $S \in \mathbb{R}^{D \times N}$, where S_i denotes the textual feature of the ith identity. The affinities between a visual feature representation v and all textual features S could then be calculated as $S^T v$. The probability of the input image v matching the ith identity in the textual feature buffer can be calculated with the following cross-modal softmax function,

$$p_i^S(v) = \frac{\exp\left(S_i^T v / \sigma_v\right)}{\sum_{j=1}^{N} \exp\left(S_j^T v / \sigma_v\right)}, \tag{6.1}$$

where σ_v is a temperature hyper-parameter to control how peaky the probability distribution is. Similarly, in each iteration, a mini-batch of sentence descriptions belonging to n identities are also sampled. Let $s \in \mathbb{R}^D$ denote one text sample's feature in the mini-batch. All visual features are pre-stored in a visual feature buffer $V \in \mathbb{R}^{D \times N}$. The probability of s matching the kth identity in the visual feature buffer is defined as

$$p_k^V(s) = \frac{\exp\left(V_k^T s / \sigma_s\right)}{\sum_{j=1}^{N} \exp\left(V_j^T s / \sigma_s\right)}, \tag{6.2}$$

where σ_s is another temperature hyper-parameter. In each iteration, our goal is to maximize the above textual and visual matching probabilities for corresponding identity pairs. The learning objective can then be define as minimizing the following CMCE loss,

$$L = -\sum_{v} \log p_{t_v}^S(v) - \sum_{s} \log p_{t_s}^V(s), \tag{6.3}$$

where t_v and t_s are the target identities of visual feature v and textual feature s respectively. Its gradients are calculated as

$$\frac{\partial L}{\partial v} = \frac{1}{\sigma_v}\left[(p^S_{t_v} - 1)S_{t_v} + \sum_{\substack{j=1 \\ j\neq t_v}}^{N} S_j p^S_j\right], \tag{6.4}$$

$$\frac{\partial L}{\partial s} = \frac{1}{\sigma_s}\left[(p^V_{t_s} - 1)V_{t_s} + \sum_{\substack{j=1 \\ j\neq t_s}}^{N} V_j p^V_j\right]. \tag{6.5}$$

The textual and visual feature buffers enable efficient calculation of textual-visual affinities between sampled identity features in one modality and all features in the other modality. This is the key to our cross-modal entropy loss. Before the first iteration, image and textual features are obtained by the CNN and LSTM. Each identity's textual and visual features are stored in its corresponding row in the textual and visual feature buffers. If an identity has multiple descriptions or images, its stored features in the buffers are the average of the multiple samples. In each iteration, after the forward propagation, the loss function is first calculated. The parameters of both visual CNN and language LSTM are updated via backpropgation. For the sampled identity images and descriptions, their corresponding rows in the textual and visual feature buffers are updated by the newly generated features. If a corresponding identity t has multiple entity images or descriptions, the buffer rows are updated as the running weighted average with the following formulations, $S_{t_v} = 0.5S_{t_v} + 0.5s$ and $V_{t_s} = 0.5V_{t_s} + 0.5v$, where s and v are the newly generated textual and visual features, t_s and t_v denote their corresponding identities.

Although our CMCE loss has similar formation with softmax loss function, they have major differences. First, the CMCE propagates gradients across textual and visual domains. It can efficiently embed features of the same identity from different domains to be similar and enlarge the distances between non-corresponding identities. Second, the feature buffers store all identities' feature representations of different modalities, making the comparison between mini-batch samples with all identities much efficient.

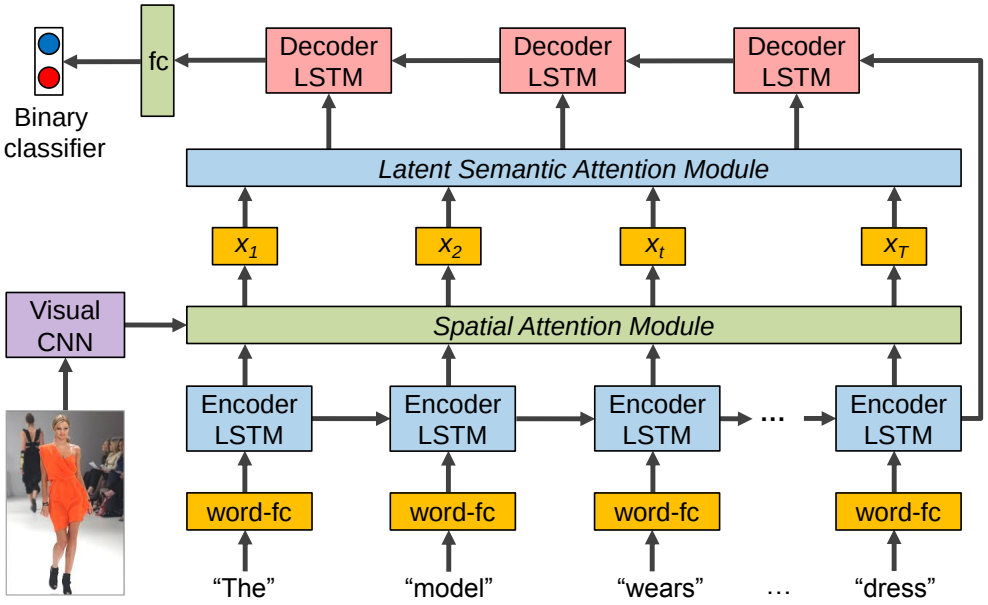


Figure 6.3: *Illustration of the stage-2 network with latent co-attention mechanism. The spatial attention associates the relevant visual regions to each input word while the latent semantic attention automatically aligns image-word features by the spatial attention modules to enhance the robustness to sentence structure variations.*

6.4 Stage-2 CNN-LSTM with Latent Co-attention

After training, the stage-1 network is able to obtain initial matching results efficiently because the textual and visual features can be calculated independently for each modality. However, the visual and text feature embeddings might not be optimal because stage-1 compresses the whole sentence into a single vector. The correspondences between individual words and image regions are not established to capture word-level similarities. Stage-1 is also sensitive to sentence structure variations. A further refinement on the stage-1 matching results is desirable for obtaining accurate matching results. For stage 2, we propose a tightly coupled CNN-LSTM network with latent co-attention mechanism, which takes a pair of text description and image as inputs and outputs their matching confidence. Stage-2 framework associates individual words and image regions with spatial attention to better capture world-level similarities, and automatically realigns sentence structures via latent semantic attention. The trained stage-1 network serves as the initial point for the stage-2 network. In addition, it screens easy negatives, so only the hard negative matching samples from stage-1 results are utilized for training stage-2. With stage-1, stage-2 can focus on handling more

challenging samples that have most impact on the final results.

The network structure for stage-2 network is shown in Figure 6.3. The visual feature for the input image is obtained by a visual CNN. Word features are generated by the encoder LSTM. At each word, a joint word-image feature is obtained via the spatial attention module, which relates the word feature to its corresponding image regions. A decoder LSTM then automatically aligns encoded features for the words to enhance robustness against sentence structure variations. The output features of the decoder LSTM is utilized to obtain the final matching confidence. The idea of spatial and latent semantic co-attention was for the first time proposed and the network is accordingly designed. Unlike LSTM decoders for NLP [89, 141], whose each step corresponds to a specific output word, each step of our semantic decoder captures a latent semantic concept and the number of steps is predefined as the number of concepts.

6.4.1 Encoder word-LSTM with spatial attention

For the visual CNN and encoder LSTM, our goal is to generate a joint word-visual feature representation at each input word. The naive solution would be simply concatenating the visual feature with word feature at each word. However, different words or phrases may relate more to specific visual regions instead of the overall image. Inspired by [89], we adopt a spatial attention mechanism to weight more on relevant visual regions for each word.

Given an input sentence description, we first encode each word to an one-hot vector and then transform them to a feature vector through a fully-connected layer and an encoder word-LSTM. We denote the word features by $H = \{h_1, \cdots, h_T\}$, $H \in \mathbb{R}^{D_H \times T}$, where h_t denotes the hidden state of the encoder LSTM at time step t and D_H is the hidden state dimension. Let $I = \{i_1, \cdots, i_L\}$, $I \in \mathbb{R}^{D_I \times L}$ represent the visual features from all L regions in the input image, where D_I is the image feature dimension, and i_l is the feature vector at the spatial region l. At time step t, the spatial attention a_t over each image region k can be calculated as

$$e_{t,k} = W_P \left\{ \tanh \left[W_I i_k + (W_H h_t + b_H) \right] \right\} + b_P, \tag{6.6}$$

$$a_{t,k} = \frac{\exp(e_{t,k})}{\exp\left(\sum_{j=1}^{L} e_{t,j}\right)}, \text{ for } k = 1, \cdots, L, \tag{6.7}$$

where $W_I \in \mathbb{R}^{K \times D_I}$ and $W_H \in \mathbb{R}^{K \times D_H}$ are the parameter matrices that transform visual and semantic features to the same K-dimensional space, and $W_P \in \mathbb{R}^{1 \times K}$ converts the coupled textual and visual features to affinity scores. Given a word at time t, the attentions $a_{t,k}$ over all L image regions are normalized by a softmax function and should sum up to 1. Intuitively, $a_{t,k}$ represents the probability that the tth word relates to the kth image region. The obtained spatial attentions are then used to gate the visual features by weighted averaging,

$$\tilde{i}_t = \sum_{k=1}^{L} a_{t,k} i_k. \tag{6.8}$$

In this way, the gated visual features focus more on relevant regions to the tth word. To incorporate both textual and visual information at each word, we then concatenate the gated visual features $\tilde{i}_t$ and hidden states h_t of LSTM as the output of the spatial attention module, $x_t = \left[\tilde{i}_t, h_t\right]$.

6.4.2 Decoder LSTM with latent semantic attention

Although the LSTM has a memory state and a forget gate to capture long-term information, it still faces challenges on processing very long sentences to compress all information of the input sentence into a fixed-length vector. It might not be robust enough against sentence structure variations. Inspired by the word alignment (attention) technique in neural machine translation [141], we propose to use a decoder LSTM with latent semantic attention to automatically align sentence structures and estimate the final matching confidence. Note that unlike the conventional decoder LSTM in machine translation, where each step corresponds to an actual word, each step of our decoder LSTM has no physical meaning but only latent semantic meaning. Given the final features encoded by the encoder LSTM, the M-step decoder LSTM processes the encoded feature step by step while searches through the entire input sentence to align the image-word features, $x_t, t = \{1, \cdots, T\}$. At the mth time step of the decoding process, the latent semantic attention a'_m for the tth input word is calculated as

$$e'_{m,t} = f(c_{m-1}, x_t), \tag{6.9}$$

$$a'_{m,t} = \frac{\exp(e'_{m,t})}{\sum_{j=1}^{T} \exp(e'_{m,j})}, \tag{6.10}$$

where f is an importance function that weights the importance of the jth word for the mth decoding step. It is modeled a two-layer Convolutional Neural Network. c_{m-1} is the hidden state by decoder LSTM for step $m-1$. At each decoding step m, the semantic attention "soft" aligns the word-image features by a weighted summation,

$$\tilde{x}_m = \sum_{j=1}^{T} a'_{m,j} x_j. \tag{6.11}$$

The aligned image-word features $\tilde{x}_m$ are then transformed by two fully-connected layers and fed into the M-step decoding LSTM to obtain the final matching confidence. By automatically aligning image-word features with latent semantic attention, at each decoding step, the decoder LSTM is able to focus more on relevant information by re-weighting the source image-word features to enhance the network's robustness to sentence structure variations. For training the stage-2 network, we also utilize identity-level annotations when constructing text-image training pairs. If an image and a sentence have the same identity, they are treated as a positive pair. Easier training samples are filtered out by the stage-1 network. The decoder LSTM is trained with the binary cross-entropy loss,

$$E = -\frac{1}{N'} \sum_{i=1}^{N'} [y_i \log C_i + (1 - y_i) \log(1 - C_i)] \tag{6.12}$$

where N' is the number of samples for training the stage-2 network, C_i is the predicted matching confidence for the ith text-image pair, and y_i denotes its target label, with 1 representing the text and image pair belonging to the same identity and 0 representing different identities.

6.5 Experiments

6.5.1 Datasets and evaluation metrics

Our proposed algorithm takes advantage of identity-level annotations from the data for achieving robust matching results. Three datasets with identity-level annotations, CUHK-PEDES [1], Caltech-UCSD birds (CUB) [2], and Oxford-102 Flowers [2], are chosen for evaluation.

CUHK-PEDES dataset. The CUHK-PEDES dataset [1] contains 40,206 images

of 13,003 person identities. Each image is described by two sentences. There are 11,003 persons, 34,054 images and 68,108 sentence descriptions in the training set. The validation set and test set consist of 3,078 and 3,074 images, respectively, and both of them contain 1,000 persons. The top-1 and top-10 accuracies are chosen to evaluate the performance of person search with natural language description following [1], which are the percentages of successful matchings between the query text and the top-1 and top-10 scored images.

CUB dataset and Flower dataset. The CUB and Flower datasets contain 11,788 bird images and 8,189 flower images respectively, where each image is labeled by ten textual descriptions. There are 200 different categories in CUB and the dataset is splited into 100 training, 50 validation, and 50 test categories. Flower has 102 flower classes and three subsets, including 62 classes for training, 20 for validation, and 20 for test. We have the same experimental setup as [2] for fair comparison. There is no overlap between training and testing classes. Similar to [2], identity classes are used only during training, and testing is on new identities. We report the AP@50 for text-to-image retrieval and the top-1 accuracy for image-to-text retrieval. Given a query textual class, the algorithm first computes the percent of top-50 retrieved images whose identity matches that of the textual query class. The average matching percentage of all 50 test classes is denoted as AP@50.

6.5.2 Implementation details

For fair comparison with existing baseline methods on different datasets, we choose VGG-16 [8] for the CUHK-PEDES dataset and GoogleNet [13] for the CUB and Flower datasets as the visual CNN. For stage-1 network, the visual features are obtained by $L2$-normalizing the output features at "drop7" and "avgpool" layers of VGG-16 and GoogleNet. We take the last hidden state of the LSTM to encode the whole sentence and embed the textual vector into the 512-dimensional feature space with the visual image. The textual features is also $L2$-normalized. The temperature parameters σ_v and σ_s in Eqs. (6.1) and (6.2) are empirically set to 0.04. The LSTM is trained with the Adam optimizer with a learning rate of 0.0001 while the CNN is trained with the batched Stochastic Gradient Descent. For the stage-2 CNN-LSTM network, instead of embedding the visual images into 1-dimensional vectors, we take the output

	Text-Image Retrieval	
Method	Top-1 (%)	Top-10 (%)
deeper LSTM Q+norm I [134]	17.19	57.82
iBOWIMG [90]	8.00	30.56
NeuralTalk [89]	13.66	41.72
Word CNN-RNN [2]	10.48	36.66
GNA-RNN [1]	19.05	53.64
GMM+HGLMM [140]	15.03	42.27
Stage-1	21.55	54.78
Stage-2	**25.94**	**60.48**

Table 6.1: *Text-to-image retrieval results by different compared methods on the CUHK-PEDES dataset [1].*

	Text-Image Retrieval	
Method	Top-1 (%)	Top-10 (%)
Triplet	14.76	51.29
Stage-1	21.55	54.78
Stage-2 w/o SMA+SPA+stage-1	17.19	57.82
Stage-2 w/o SMA+SPA	22.11	58.05
Stage-2 w/o SMA	23.58	58.68
Stage-2 w/o ID	23.47	54.77
Stage-2	**25.94**	**60.48**

Table 6.2: *Ablation studies on different components of the proposed two-stage framework. "w/o ID": not using identity-level annotations. "w/o SMA": not using semantic attention. "w/o SPA": not using spatial attention. "w/o stage-1": not using stage-1 network for training initialization and easy result screening.*

of the "pool5" layer of VGG-16 or the "inception (5b)" layer of GoogleNet as the image representations for learning spatial attention. During the training phase, we first train the language model and fix the CNN model, and then fine-tune the whole network jointly to effectively couple the image and text features. The training and testing samples are screened by the matching results of stage-1. For each visual or textual sample, we take its 20 most similar samples from the other modality by stage-1 network and construct textual-visual pair samples for stage-2 training and testing. Each text-image pair is assigned with a label, where 1 represents the corresponding pair and 0 represents the non-corresponding one. The step length M of the decoding LSTM is set to 5.

6.5.3 Results on CUHK-PEDES dataset

We compare our proposed two-stage framework with six methods on the CUHK-PEDES dataset. The top-1 and top-10 accuracies of text-to-image retrieval are recorded in

Table 6.1. Note that only text-to-image retrieval results are evaluated for the dataset because image-to-text retrieval is not a practical problem setting for the dataset. Our method outperforms state-of-the-art methods by large margins, which demonstrates the effectiveness of the proposed two-stage framework in matching textual and visual entities with identity-level annotations.

Our stage-1 model outperforms all the compared methods. The gain on top-1 accuracy by our proposed method is 2.50% compared with the state-of-the-art GNA-RNN [1], which has more complex network structure than ours. This shows the advantages of the CMCE loss. Furthermore, the introduction of feature buffers make the comparison computation more efficient even with a large number of identities. GMM+HGLMM [140] uses the Fisher Vector as a sentence representation by pooling the word2vec embedding of each word in the sentence. The Word CNN-RNN [2] aims to minimize the distances between corresponding textual-visual pairs and maximize the distances between non-corresponding ones within each mini-batch. However, such a method is restricted by the mini-batch size and cannot be applied to dataset with a large number of identities. Our CMCE loss results in a top-1 accuracy of 21.55%, which outperforms the Word CNN-RNN's 10.48%. The stage-1 CNN-LSTM with CMCE loss performs well on both accuracy and time complexity with its loosely coupled network structure.

The stage-2 CNN-LSTM with latent co-attention further improves the performances by 4.39% and 5.70% in terms of top-1 and top-10 accuracies. The co-attention mechanism aligns visual regions with latent semantic concepts effectively to minimize the influence of sentences structure variations. Compared with methods that randomly sample pairs, such as deeper LSTM Q+norm I [134], iBOWIMG [90], NeuralTalk [89] and GNA-RNN [1], our network focuses more on distinguishing the hard samples after filtering out most easy non-correponding samples by the stage-1 network.

6.5.4 Ablation studies

In this section, we investigate the effect of each component in the stage-1 and stage-2 networks by performing a series of ablation studies on the CUHK-PEDES dataset. We first investigate the importance of proposed CMCE loss. We train our stage-1 model with the proposed loss replaced by triplet loss [2], named "Triplet". As shown in Table

6.2, its top-1 drops by 6.79% on the CUHK-PEDES set compared with our stage-1 with the new loss function. In addition, triplet loss [2] needs 3 times more training time. Then we investigate the importance of the identity-level annotations to the textual-visual matching performance by ignoring the annotations. In this case, each image or sentence is treated as an independent identity. The top-1 and top-10 accuracies of "Stage-2 w/o ID" have 2.47% and 5.71% drops compared with the results of "Stage-2", which demonstrates that the identity-level annotations can help textual-visual matching by minimizing the intra-identity feature variations.

To demonstrate the effectiveness of our latent semantic attention, we remove it from the original stage-2 network, denoted as "Stage-2 w/o SMA". The top-1 accuracy drops by 2.36%, which shows the latent semantic attention can help align the visual and semantic concepts and mitigate the LSTM's sensitivity to different sentence structures. The spatial attention tries to relate words or phrases to different visual regions instead of the whole image. Based on the framework of "Stage-2 w/o SMA", we further remove the spatial attention module from the stage-2 network, denoted as "Stage-2 w/o SMA+SPA", which can be viewed as a simple concatenation of the visual and textual features from the CNN and LSTM, followed by two fully-connected layers for binary classification. Both the top-1 and top-10 accuracies decrease compared with "Stage-2 w/o SMA".

The stage-1 network is able to provide samples for the training and evaluation of stage-2 network, and also serves as the initial point for its training. To investigate the influence of stage-1 network, we design one additional baselines, denoted as "Stage-2 w/o SMA+SPA+Stage-1". This baseline is trained without using the stage-1 network. It shows an apparent performance drop compared with the "Stage-2 w/o SMA+SPA" baseline, which demonstrates the necessity of the stage-1 network in our proposed framework. Since stage-1 network chooses only 20 most closest images of each query text for stage 2 during the evaluation phase, the effect of some components might not be apparent in terms of the top-10 accuracy.

	Image-Text		Text-Image	
	Top-1 Acc (%)		AP@50 (%)	
Methods	DA-SJE	DS-SJE	DA-SJE	DS-SJE
BoW [143]	43.4	44.1	24.6	39.6
Word2Vec [142]	38.7	38.6	7.5	33.5
Attributes [144]	50.9	50.4	20.4	50.0
Word CNN [2]	50.5	51.0	3.4	43.3
Word CNN-RNN [2]	54.3	56.8	4.8	48.7
GMM+HGLMM [140]	36.5		35.6	
Triplet	52.5		52.4	
Stage-1	61.5		55.5	
Stage-2	–		**57.6**	

Table 6.3: *Image-to-text and text-to-image retrieval results by different compared methods on the CUB dataset [2].*

6.5.5 Results on the CUB and Flower datasets

Tables 6.3 and 6.4 show the experimental results of image-to-text and text-to-image retrieval on the CUB and Flower datasets. We compare with state-of-the-art methods on the two datasets. The CNN-RNN [2] learns a CNN-RNN textual encoder for sentence feature embedding and transforms both visual and textual features into the same embedding space. Different text features are also combined with the CNN-RNN methods. The Word2Vec [142] averages the pre-trained word vector of each word in the sentence description to represent textual features. BoW [143] is the output of an one-hot vector passing through a single layer linear projection. Attributes [144] maps attributes to the embedding space by learning a encoder function. Different types of textual representations are combined with the CNN-RNN framework for testing. Our method outperforms the state-of-the-art CNN-RNN by more than 3% in terms of top-1 image-to-text retrieval accuracy and about 10% in terms of text-to-image retrieval AP@50 on both datasets, which shows the effectiveness of the proposed method. For the "Triplet" baseline, the top-1 and AP@50 drop by 9.0% and 3.1% on CUB dataset, and drop by 4.1% and 3.1% on Flower dataset which demonstrate the proposed loss function performs better than the traditional triplet loss. Since the top-1 accuracy provided by [2] is computed by fusing sentences of the same class into one vector and our stage-2 network is therefore not suitable for the image-to-text retrieval task, we only report the stage-1 results on image-to-text retrieval which has already outperformed other baselines.

	Image-Text		Text-Image	
	Top-1 Acc (%)		AP@50 (%)	
Methods	DA-SJE	DS-SJE	DA-SJE	DS-SJE
BoW [143]	56.7	57.7	28.2	57.3
Word2Vec [142]	54.6	54.2	16.3	52.1
Word CNN [2]	60.2	60.7	8.7	56.3
Word CNN-RNN [2]	60.9	65.6	7.6	59.6
GMM+HGLMM [140]	54.8		52.8	
Triplet	64.3		64.9	
Stage-1	68.4		68.0	
Stage-2	–		**70.1**	

Table 6.4: *Image-to-text and text-to-image retrieval results by different compared methods on the Flower dataset [2].*

6.5.6 Qualitative results

We also conduct qualitative evaluations of the proposed methods. Figure 6.4 shows example text-to-image retrieval results. Most sentences can correctly match images corresponding to their descriptions. In the first case, almost all the persons wear a sweater with "black gray and white stripes". Different images of the same identity (the first, second, and fifth person images) appear in the top-ranked results, which shows the proposed two-stage CNN-LSTM can correctly match identities across different domains and minimizes the intra-identity distances. Some mis-matching results are even challenging for human to distinguish with subtle differences in visual appearance. In the second case, the first and second person both wear "white short sleeved shirt", but only the first one is the true matching result because of the "black purse" carried on her shoulder.

6.6 Conclusion

In this chapter, we proposed a novel two-stage framework for identity-aware visual-semantic matching. The framework consists of two deep neural networks. The stage-1 CNN-LSTM network learns to embed the input image and description to the same feature space and minimizes the intra-identity distance simultaneously with the CMCE loss. It serves as initial point for stage-2 training and also provides training and evaluation samples for stage-2 by screening most incorrect matchings. The stage-2 network is a CNN-LSTM with latent co-attention mechanism which jointly learns the spatial attention and latent semantic attention by an alignment decoder LSTM. It automatically

The man is wearing a sweater with black gray and white stripes on it. He is wearing tan pants and gray shoes. He is carrying a bag on his back.

This woman is wearing a white short sleeved shirt, a white skirt and gray flat shoes. She is also carrying a black purse on her shoulder.

This bird is nearly all brown with a hooked bill.

A brown bird with a white crown and a small yellow pointed beak.

This is a white flower with wide petals and a pink and yellow pistil.

This flower has thick and sharply tipped petals of bright yellow which angle directly upwards.

Figure 6.4: *Example text-to-image retrieval results by the proposed framework. Corresponding images are marked by green rectangles. (Left to right) For each text description, the matching results are sorted according to the similarity scores in a descending order. (Row 1) results from the CUHK-PEDES dataset [1]. (Row 2) results from the CUB dataset [2]. (Row 3) results from the Flower dataset [2].*

aligns different words and image regions to minimize the impact of sentence structure variations. We evaluate the proposed method on three datasets and perform a series

of ablation studies to verify the effect of each component. Our method outperforms state-of-the-art approaches by a large margin and demonstrates the effectiveness of the proposed framework for identity-aware visual-textual matching.

Chapter 7

Conclusions

In this thesis we propose three deep learning based approaches to make person search applicable to real-world applications. For the video-based person search, we employ a diversity regularized spatial attention model to detect similar local patches across multiple video frames. This method determines whether a particular part of the body is occluded or not and align corresponding image patches across frames. On the other hand, we collect a large-scale person description dataset. A recurrent neural network with gated neural attention is proposed to learn the affinities between the query sentence and person images. We extend the natural language based person search to the general visual-semantic matching problem. A cross-model cross-entropy loss function and a co-attention model are designed to solve the visual-semantic matching. The two-stage framework achieves accuracy and efficiency simultaneously.

Despite our efforts toward person search, the methods developed can be applied to other vision tasks as well. The multiple spatial attention and diversity regularization can be used for fine-grained classification; the cross-model cross-entropy loss is suitable for cross-domain feature learning; the method for textual-visual matching is also compatible with language and vision-related tasks such as image captioning and visual question and answering. Person search is far from being solved, more cues and techniques such as social relationship and whole scene images are also worth to be investigated.

Bibliography

[1] S. Li, T. Xiao, H. Li, B. Zhou, D. Yue, and X. Wang, "Person search with natural language description," in *CVPR*, 2017.

[2] S. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *CVPR*, 2016.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, 2004.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[5] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *CVPR*, 2010.

[6] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *CVPR*, 2015.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2014.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, 1997.

[12] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.

[17] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[18] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[19] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *CVPR*, 2014.

[20] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *CVPR*, 2015.

[21] B. Prosser, W.-S. Zheng, S. Gong, T. Xiang, and Q. Mary, "Person re-identification by support vector ranking." in *BMVC*, 2010.

[22] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *ECCV*, 2008.

[23] S. Liao and S. Z. Li, "Efficient psd constrained asymmetric metric learning for person re-identification," in *ICCV*, 2015.

[24] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *CVPR*, 2012.

[25] S. Bak and P. Carr, "One-shot metric learning for person re-identification," in *CVPR*, 2017.

[26] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *CVPR*, 2013.

[27] B. Ma, Y. Su, and F. Jurie, "Local descriptors encoded by fisher vectors for person re-identification," in *ECCV*, 2012.

[28] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "Joint detection and identification feature learning for person search," in *CVPR*, 2017.

[29] I. Kviatkovsky, A. Adam, and E. Rivlin, "Color invariants for person reidentification," *TPAMI*, vol. 35, no. 7, pp. 1622–1634, 2013.

[30] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "End-to-end deep learning for person search," *arXiv preprint arXiv:1604.01850*, 2016.

[31] S. Ding, L. Lin, G. Wang, and H. Chao, "Deep feature learning with relative distance comparison for person re-identification," *Pattern Recognition*, 2015.

[32] O. Hamdoun, F. Moutarde, B. Stanciulescu, and B. Steux, "Person re-identification in multi-camera system by signature based on interest point descriptors collected on short video sequences," in *ICDSC*, 2008.

[33] B. Prosser, S. Gong, and T. Xiang, "Multi-camera matching using bi-directional cumulative brightness transfer functions." in *BMVC*, 2008.

[34] W. Li, Y. Wu, M. Mukunoki, Y. Kuang, and M. Minoh, "Locality based discriminative measure for multiple-shot human re-identification," *Neurocomputing*, 2015.

[35] N. McLaughlin, J. Martinez del Rincon, and P. Miller, "Recurrent convolutional network for video-based person re-identification," in *CVPR*, 2016.

[36] Z. Zhou, Y. Huang, W. Wang, L. Wang, and T. Tan, "See the forest for the trees: Joint spatial and temporal recurrent neural networks for video-based person re-identification," in *CVPR*, 2017.

[37] S. Li, S. Bak, P. Carr, and X. Wang, "Diversity regularized spatiotemporal attention for video-based person re-identification," *arXiv preprint arXiv:1803.09882*, 2018.

[38] D. Chung, K. Tahboub, and E. J. Delp, "A two stream siamese convolutional neural network for person re-identification," in *CVPR*, 2017.

[39] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *ICCV*, 2009.

[40] T. Xiao, H. Li, W. Ouyang, and X. Wang, "Learning deep feature representations with domain guided dropout for person re-identification," in *CVPR*, 2016.

[41] I. Masi, S. Rawls, G. Medioni, and P. Natarajan, "Pose-aware face recognition in the wild," in *CVPR*, 2016.

[42] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *CVPR*, 2016.

[43] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015.

[44] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *ICML*, 2007.

[45] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *NIPS*, 2005.

[46] B. McFee and G. R. Lanckriet, "Metric learning to rank," in *ICML*, 2010.

[47] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," in *CVPR*, 2016.

[48] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, "Compact bilinear pooling," in *CVPR*, 2016.

[49] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *PETS*, 2007.

[50] W.-S. Zheng, S. Gong, and T. Xiang, "Associating groups of people," in *BMVC*, 2009.

[51] C. C. Loy, T. Xiang, and S. Gong, "Multi-camera activity correlation analysis," in *CVPR*, 2009.

[52] D. Baltieri, R. Vezzani, and R. Cucchiara, "3dpes: 3d people dataset for surveillance and forensics," in *ACM workshop on Human gesture and behavior understanding*, 2011.

[53] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning." in *ACCV*, 2012.

[54] W. Li and X. Wang, "Locally aligned feature transforms across views," in *CVPR*, 2013.

[55] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang, "Joint detection and identification feature learning for person search," in *CVPR*, 2017.

[56] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015.

[57] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," *arXiv preprint arXiv:1701.07717*, 2017.

[58] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof, "Person re-identification by descriptive and discriminative classification," in *Image Analysis*, 2011.

[59] T. Wang, S. Gong, X. Zhu, and S. Wang, "Person re-identification by video ranking," in *ECCV*, 2014.

[60] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian, "Mars: A video benchmark for large-scale person re-identification," in *ECCV*, 2016.

[61] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *TPAMI*, 2010.

[62] A. Dehghan, S. Modiri Assari, and M. Shah, "Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *CVPR*, 2015.

[63] X. Wang, "Intelligent multi-camera video surveillance: A review," *Pattern recognition letters*, 2013.

[64] S.-I. Yu, Y. Yang, and A. Hauptmann, "Harry potter's marauder's map: Localizing and tracking multiple persons-of-interest by nonnegative discretization," in *CVPR*, 2013.

[65] Y. Shen and Z. Miao, "Multihuman tracking based on a spatial–temporal appearance match," *IEEE transactions on Circuits and systems for video technology*, vol. 24, no. 3, pp. 361–373, 2014.

[66] N. Gheissari, T. B. Sebastian, and R. Hartley, "Person reidentification using spatiotemporal appearance," in *CVPR*, 2006.

[67] W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *CVPR*, 2011.

[68] D. Li, X. Chen, Z. Zhang, and K. Huang, "Learning deep context-aware features over body and latent parts for person re-identification," in *CVPR*, 2017.

[69] J. You, A. Wu, X. Li, and W.-S. Zheng, "Top-push video-based person re-identification," in *CVPR*, 2016.

[70] X. Zhu, X.-Y. Jing, F. Wu, and H. Feng, "Video-based person re-identification by simultaneously learning intra-video and inter-video distance metrics." in *IJCAI*, 2016.

[71] X. Ma, X. Zhu, S. Gong, X. Xie, J. Hu, K.-M. Lam, and Y. Zhong, "Person re-identification by unsupervised video matching," *Pattern Recognition*, vol. 65, pp. 197–210, 2017.

[72] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015.

[73] S. Li, T. Xiao, H. Li, B. Zhou, D. Yue, and X. Wang, "Person search with natural language description," in *CVPR*, 2017.

[74] S. Li, T. Xiao, H. Li, W. Yang, and X. Wang, "Identity-aware textual-visual matching with latent co-attention," in *ICCV*, 2017.

[75] X. Liu, H. Zhao, M. Tian, L. Sheng, J. Shao, S. Yi, J. Yan, and X. Wang, "Hydraplus-net: Attentive deep features for pedestrian analysis," *arXiv preprint arXiv:1709.09930*, 2017.

[76] H. Liu, Z. Jie, K. Jayashree, M. Qi, J. Jiang, S. Yan, and J. Feng, "Video-based person re-identification with accumulative motion context," *arXiv preprint arXiv:1701.00193*, 2017.

[77] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: towards good practices for deep action recognition," in *ECCV*, 2016.

[78] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear cnns for fine-grained visual recognition," in *TPAMI*, 2017.

[79] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, "Semantic segmentation with second-order pooling," in *ECCV*, 2012.

[80] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, "A structured self-attentive sentence embedding," *arXiv preprint arXiv:1703.03130*, 2017.

[81] R. Beran, "Minimum hellinger distance estimates for parametric models," *The Annals of Statistics*, pp. 445–463, 1977.

[82] K. Liu, B. Ma, W. Zhang, and R. Huang, "A spatio-temporal appearance representation for video-based pedestrian re-identification," in *ICCV*, 2015.

[83] S. Karanam, Y. Li, and R. J. Radke, "Person re-identification with discriminatively trained viewpoint invariant dictionaries," in *ICCV*, 2015.

[84] J. Chen, Y. Wang, and Y. Y. Tang, "Person re-identification by exploiting spatio-temporal cues and multi-view metric learning," *IEEE Signal Processing Letters*, vol. 23, no. 7, pp. 998–1002, 2016.

[85] L. Zheng, H. Zhang, S. Sun, M. Chandraker, and Q. Tian, "Person re-identification in the wild," *arXiv preprint arXiv:1604.02531*, 2016.

[86] Y. Liu, J. Yan, and W. Ouyang, "Quality aware network for set to set recognition," in *CVPR*, 2017.

[87] F. M. Khan and F. Bremond, "Multi-shot person re-identification using part appearance mixture," in *WACV*, 2017.

[88] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *CVPR*, 2015.

[89] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *CVPR*, 2015.

[90] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus, "Simple baseline for visual question answering," *arXiv preprint arXiv:1512.02167*, 2015.

[91] M. Ren, R. Kiros, and R. Zemel, "Exploring models and data for image question answering," in *NIPS*, 2015.

[92] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu, "Shape and appearance context modeling," in *ICCV*, 2007.

[93] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *CVPR*, 2013.

[94] F. Porikli, "Inter-camera color calibration by correlation model function," in *ICIP*, 2003.

[95] Y. Shen, W. Lin, J. Yan, M. Xu, J. Wu, and J. Wang, "Person re-identification with correspondence structure learning," in *ICCV*, 2015.

[96] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Learning to rank in person re-identification with metric ensembles," in *CVPR*, 2015.

[97] X. Li, W.-S. Zheng, X. Wang, T. Xiang, and S. Gong, "Multi-scale learning for low-resolution person re-identification," in *ICCV*, 2015.

[98] W.-S. Zheng, X. Li, T. Xiang, S. Liao, J. Lai, and S. Gong, "Partial person re-identification," in *ICCV*, 2015.

[99] D. A. Vaquero, R. S. Feris, D. Tran, L. Brown, A. Hampapur, and M. Turk, "Attribute-based people search in surveillance environments," in *WACV*, 2009.

[100] C. Su, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Deep attributes driven multi-camera person re-identification," *arXiv preprint arXiv:1605.03259*, 2016.

[101] Y. Deng, P. Luo, C. C. Loy, and X. Tang, "Pedestrian attribute recognition at far distance," in *ACM MM*, 2014.

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

[103] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," *Transactions of the Association for Computational Linguistics*, vol. 2, 2014.

[104] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick, "Microsoft coco captions: Data collection and evaluation server," *arXiv preprint arXiv:1504.00325*, 2015.

[105] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[106] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *arXiv preprint arXiv:1602.07332*, 2016.

[107] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-ucsd birds 200," 2010.

[108] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proceedings of Indian Conference on Computer Vision, Graphics & Image Processing*, 2008.

[109] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," in *CVPR*, 2016.

[110] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang *et al.*, "T-cnn: Tubelets with convolutional neural networks for object detection from videos," *arXiv preprint arXiv:1604.02532*, 2016.

[111] K. Kang, H. Li, T. Xiao, W. Ouyang, J. Yan, X. Liu, and X. Wang, "Object detection in videos with tubelet proposal networks," in *CVPR*, 2017.

[112] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *ICCV*, 2015.

[113] R. Hu, M. Rohrbach, and T. Darrell, "Segmentation from natural language expressions," *arXiv preprint arXiv:1603.06180*, 2016.

[114] J. Johnson, A. Karpathy, and L. Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," *arXiv preprint arXiv:1511.07571*, 2015.

[115] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu, "Are you talking to a machine? dataset and methods for multilingual image question," in *NIPS*, 2015.

[116] X. Chen and C. L. Zitnick, "Learning a recurrent visual representation for image caption generation," *arXiv preprint arXiv:1411.5654*, 2014.

[117] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt *et al.*, "From captions to visual concepts and back," in *CVPR*, 2015.

[118] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," *arXiv preprint arXiv:1412.6632*, 2014.

[119] H. Noh, P. H. Seo, and B. Han, "Image question answering using convolutional neural network with dynamic parameter prediction," *arXiv preprint arXiv:1511.05756*, 2015.

[120] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," *arXiv preprint arXiv:1511.02274*, 2015.

[121] K. Saito, A. Shin, Y. Ushiku, and T. Harada, "Dualnet: Domain-invariant network for visual question answering," *arXiv preprint arXiv:1606.06108*, 2016.

[122] M. Malinowski, M. Rohrbach, and M. Fritz, "Ask your neurons: A neural-based approach to answering questions about images," in *ICCV*, 2015.

[123] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach, "Multimodal compact bilinear pooling for visual question answering and visual grounding," *arXiv preprint arXiv:1606.01847*, 2016.

[124] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov *et al.*, "Devise: A deep visual-semantic embedding model," in *NIPS*, 2013.

[125] W. Liu, T. Mei, Y. Zhang, C. Che, and J. Luo, "Multi-task deep visual-semantic embedding for video thumbnail selection," in *CVPR*, 2015.

[126] M. Ren, R. Kiros, and R. Zemel, "Image question answering: A visual semantic embedding model and a new dataset," *CoRR, abs/1505.02074*, vol. 7, 2015.

[127] L. Zheng, L. Shen, L. Tian, S. Wang, J. Bu, and Q. Tian, "Person re-identification meets image search," *arXiv preprint arXiv:1502.02171*, 2015.

[128] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell, "Natural language object retrieval," *CVPR*, 2016.

[129] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille, "Explain images with multimodal recurrent neural networks," *arXiv preprint arXiv:1410.1090*, 2014.

[130] F. Yan and K. Mikolajczyk, "Deep correlation for matching images and text," in *CVPR*, 2015.

[131] L. Wang, Y. Li, and S. Lazebnik, "Learning deep structure-preserving image-text embeddings," in *CVPR*, 2016.

[132] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *NIPS*, 2009.

[133] M. Rohrbach, M. Stark, and B. Schiele, "Evaluating knowledge transfer and zero-shot learning in a large-scale setting," in *CVPR*, 2011.

[134] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *ICCV*, 2015.

[135] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in *CVPR*, 2016.

[136] H. Nam, J.-W. Ha, and J. Kim, "Dual attention networks for multimodal reasoning and matching," *arXiv preprint arXiv:1611.00471*, 2016.

[137] J. Lu, J. Yang, D. Batra, and D. Parikh, "Hierarchical question-image co-attention for visual question answering," in *NIPS*, 2016.

[138] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang, "Spindle net: Person re-identification with human body region guided feature decomposition and fusion," in *CVPR*, 2017.

[139] X. Chen and C. Lawrence Zitnick, "Mind's eye: A recurrent visual representation for image caption generation," in *CVPR*, 2015.

[140] B. Klein, G. Lev, G. Sadeh, and L. Wolf, "Associating neural word embeddings with deep image representations using fisher vectors," in *CVPR*, 2015.

[141] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[142] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013.

[143] Z. S. Harris, "Distributional structure," *Word*, vol. 10, no. 2-3, pp. 146–162, 1954.

[144] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, "Evaluation of output embeddings for fine-grained image classification," in *CVPR*, 2015.