國立台灣科技大學
電 子 工 程 系

碩士學位論文

第一人稱視角下之動作辨識

Action Recognition in First-Person View Point

游騰凱

M10302149

指導教授： 陳郁堂 博士

中華民國一百零六年一月二十七日

中 文 摘 要

跌倒對年長者來說是一個嚴重的問題,大多時候長者的受傷來自於跌倒.有時候長者一個人在家跌倒,但是沒有人發現.家庭監控系統可以偵測到跌倒的動作並且迅速的讓其他人知道.因此如何準確的偵測跌倒是一個很重要的問題.現在科技日新月異.跟傳統的RGB攝影機相比,RGB-D的攝影機可以提供更多的資訊,例如深度及骨架.這可以更有效的去改善跌倒真測的準確度.所以我們需要找一個方法利用這些資訊去進行跌倒真測.

這篇論文中我們首先從RGB-D攝影機中截取了深度及骨架資訊.首先我們對每個影片都計算它們的深度及骨架特徵,分別叫"DMM-HOG"以及"Moving Pose Descriptor".然後我們用"pooling"的方法把我們的"Moving Pose Descriptor"轉換成一條富有意義向量去代表.再來我們採用"sparse code"去轉換我們的兩種特徵."sparse code"在先前的paper已經被證明對動作辨識很好的效果.最後我們用"logistic regression"成功的將我們的兩種特徵合併

在實驗中我們在兩個跌倒偵測的dataset進行測試,並且在其中都得到了最高的準確度.我們也在動作辨識的dataset"MSR Action3D"進行測試,結果也證明我們成功的將深度及骨架兩種不同資訊的優點合併.

Abstract

Fall is one of the most serious problems of the elderly.Most of the time elderly people injured from the fall.Sometimes the elderly fall at home, but nobody know. The home surveillance system can detect fall action and let other people know quickly.Therefore how to accurate detect fall action is an important issue.The technological advances now.Compared with traditional RGB camera,the RGB-D camera can provide more information such as depth and skeleton,it will be greatly improve accuracy of detect fall action.Therefore we want to find a way to do the fall detection by using depth and skeleton informations.

In this paper,we extract both depth and skeleton information from RGB-D video.First we compute the depth and skeleton descriptor for each video called 'DMM-HOG'and 'Moving Pose Descriptor'.Therefore we use the pooling operation to convert the 'Moving Pose Descriptor' into a vector with a meaningful representation.Then we apply sparse code to encode both descriptors.The sparse coding is proved to do action recognition very well by previous paper.Finally,we combine them successfully by logistic regression.In experiment we test on two fall detection dataset and get the highest accuracy with other methods.We also evaluate our approach on public action recognition dataset'MSR Action3D'.The result prove that we combine both advantage of depth and skeleton information successfully.

Acknowledgment

Table of contents

List of Tables

Chapter 1 Introduction

Action recognition is a well-known issue in computer vision, but most of the research focus on stable viewpoint. That means the camera will be placed at a certain place and the video will be kept stable. This kind of research topic has been discussed in the past decades. In recent years, many portable devices like Google Glass, HTC Re and GoPro are getting more popular. People can easily record their daily lives between themselves and the world. The numbers of this First-Person video are drastically increasing and most of them record people's daily lives. Thus, First-Person video or Egocentric video has become a new issue in action recognition.

First-Person video contains strong global motion produced by the person and we want to recognize actions in this situation. If we can precisely identify such videos, there are many applications such as human interaction, robot-human interaction or video based daily life-logging. Further more, once we know the action is performing, according to this, we can do some reactions or record and analyze someone's habit. Despite there are many related researches in recent years, they don't focus on feature selection and encoding temporal information.

Here, we apply Improved Trajectories by Wang [], the state-of-the-art method and handcraft feature (TRAJECTORY HOG HOF MBH) for many action recognition datasets. Not only for normal action recognition datasets, but also showed its robustness in egocentric video dataset [] because it calculates the homography between two consecutive frames. In our experiment, we discover that ego-motion in videos will cause lots of noise when extracting feature. We also explore the temporal relationship in videos. Here, we propose a method using Page Rank, a website ranking algorithm, which can analyze these extracted features and filter out the noise features. After this process, we encode these features with their spatio information from Fisher Vector and temporal information from another encoder named VideoDarwin.

Chapter 2 Related work

2.1 Related Topics

In the past few years, more and more researchers are investing that how to describe an First-Person or egocentric video. Some of them focused on object recognition [] or object-based activity recognition, some of them coped with video summarization or "life-logging" [], social interaction recognition. Other issue like video annotation or segmentation in daily life have also been proposed. Almost all the topics from standard action video can be imported in First-Person video, but the environment is different. How to extract robust features becomes a very important issue.

2.2 Previous Method

M.S. Ryoo and Larry Matthies opened up two First-Person datasets, JPL-Interaction dataset [] and DogCentric Activity dataset [] which focused on First-Person view point action recognition. On the basis of action recognition, researchers can follow the previous process In JPL-Interaction dataset, they use global motion descriptor (optical flow) and local motion descriptor(3D XYT space-time features) then applied bag-of-words to represent motion information and multi-channel kernel SVM or hierarchical structure learning as classification. In DogCentric Activity dataset, they also used the same process but applied LBP (local binary pattern) and dense optical flow as global motion descriptors. After that, they proposed Pooled Motion Features for First-Person Videos [], an CNN-based feature named *pooled time series* (PoT). They claimed PoT particularly design for First-Person video, based on time series pooling of feature descriptors.

Wang [] estimate the camera motion and fix the video to reduce the ego-motion then extract HOG, HOF and MBH from each local trajectory descriptors. This method is the state-of-the-art results on action recognition challenging datasets. S

Narayan [] defined a foreground motion map to group these trajectories and introduced the NUS First-Person Interaction dataset in two perspectives, S Song [] proposed Multi-model Fisher Vector (MFV) which utilizes the Fisher Kernel framework to combine trajectory features and their new defined temporal feature. S Song also released two ego-centric activity recognition datasets.

Here, we keep the robustness of improved trajectory and propose a heuristic method from Page Rank algorithm. We apply Page Rank to score every trajectories and delete the noise. Different from above researches, we use VideoDarwin [] to encode the temporal information. The rest of this thesis is arranged as follows. In chapter 3, descriptors, encoding way and classification are discussed. In chapter 4, datasets, experiment setup and results are shown and we concluded in chapter 5.

Chapter 3 Approach

3.1 Feature representation of First-Person video

There are many classical handcraft image features, e.g., 3D-SIFT [], SURF [], HOG3D [], LBP [], and dense trajectories[]. Not only handcraft features, Deeping Learning feature like CNN also showed their robustness and generosity. Dense trajectories [] have been shown to outperform on action recognition datasets. In First-Person video, some camera motion will produced by the photographer. These camera motion could reduce our prediction accuracy. Thus, we use improved trajectories by Wang [] as our feature, which can help us to remove camera motion and extract robust features.

3.2 Camera motion estimation

In order to find the global background motion, we estimate two consecutive frames' homographic matrix by RANSAC. This homographic matrix can allow us to fix the images to remove camera motion. In order to build this homographic matrix, we need to find the relationship between two consecutive frames. Here we use SURF feature [] and optical flow from polynomial expansion to construct efficient candidate matches.

For SURF feature, we match these feature extract between two frames based on nearest neighbor rule. On the other hand, we use the good-feature-to-track criterion to find out salient features in video. After above method, we combine these two types of matches then find the homography matrix by RANSAC.

Random Sample Consensus(RANSAC), is to iteratively find a model that can separate inlier and outlier feature in a feature space. This model is homography matrix that can represent the relationship of camera motion, then we purify the image through this matrix to remove the camera motion cause from any shooting

environment.

3.3 Dense Trajectory features

Here, we use the state-of-the-art dense trajectory features as our feature descriptors []. We take the homography result sequences to extract these descriptors. Fisrt, we track the discriminative point using median filter in a dense optical field. We only track the feature points for t frames and sample the new one if the duration time above t, also delete statistic feature trajectories if they don't include motion characteristic or the trajectories have drastically change.

Several kinds of descriptor are extracted for every trajectories. We keep using the setting from [], *trajectory* record the location through the hole trajectory, then estimate *HOG*, *HOF*, *MBHX* and *MBHY*. *HOG*(histogram of gradient) give us appearance information. *HOF*(histogram of optical flow), *MBHX* and *MBHY*(derivative of x and y direction) tell us the motion information. The dimension of each descriptor are 30 for *trajectory*(t=15), 96 for *HOG*, 108 for *HOF* and 96 for *MBHX* and *MBHY*, total dimension include the frame index t is 427 for each trajectory. Other details can see [].

3.4 Fisher Vector Encoding

Fisher Vector and Bag-of-Words are well known encoding method in computer vision. As the result from Improved Trajectories [] and Interaction Recognition in First-person videos [], showed us that Fisher Vector had the better performance than Bag-of-Words. Thus, we use Fisher Vector as our feature encoding. It encodes the first and second order statistic between the video descriptor and a Gaussian Mixture Model(GMM). We set the number of Gaussian parameter K=256 and randomly choose 256000 features to train GMM. Before training GMM, we use PCA to reduce the dimension of feature by half. Each kind of descriptors is represented as a $2DK$ dimensional Fisher Vector and normalized by power and L2 then concatenate

every normalized vector into a single vector. This concatenated vector encode the spatio information.

3.5 Modeling Video Evolution

Not only the spatio information, we also explore how to get the temporal information in each videos. VideoDarwin [] use the parameters of the ranking machine as a new video representation which help us to encode the temporal relationship through the entire video.

First, assume there are n frames of a video, $X = [\mathbf{X}_1, \mathbf{X}_2..., \mathbf{X}_n]$ and frame t is represented by vector $\mathbf{X}_t \in \mathbf{R}^D$. Here, we gather all the trajectory features in frame t and use the same GMM model to encode a fisher vector as the frame representation $\mathbf{X}_t$. Second, we apply a vector valued function F over the time variable $t, F : t \rightarrow \mathbf{v}_t$ where $\mathbf{v}_t \in \mathbf{R}^D$ and $F(X) = V$. In [1]'s experiment, *Time Vary Mean* was the best function and we also choose this function as our F. The mean vector at time t as $\mathbf{m}_t = \frac{1}{t} \times \sum_{\tau=1}^{t} \mathbf{x}_\tau$ and $\mathbf{v}_t = \frac{\mathbf{m}_t}{\|\mathbf{m}_t\|}$ captures only the direction of the unit mean appearance. The transformed vector $\mathbf{v}_t$ from $\mathbf{x}_t$ contains the temporal information from first frame up to frame t, denoted by $\mathbf{x}_{1:\mathbf{t}}$. This process give us a better dependency between the input X. Third, we want to model chronological order from V that $\mathbf{v}_t$ will be inherited by $\mathbf{v}_{t+1}$.

we estimate parameters $\mathbf{u} \in \mathbf{R}^D$ by SVR(Support Vector Regression), $\mathbf{u}$ can be viewed as a linear or non-linear model(different kernel) of V.

In our experiment, we follow from Improved Trajectories [],Egocentric Life-logging Videos [] and Action and Interaction Recognition [] , use the same parameter $C = 100$ and one-against-all linear SVM as our classify.

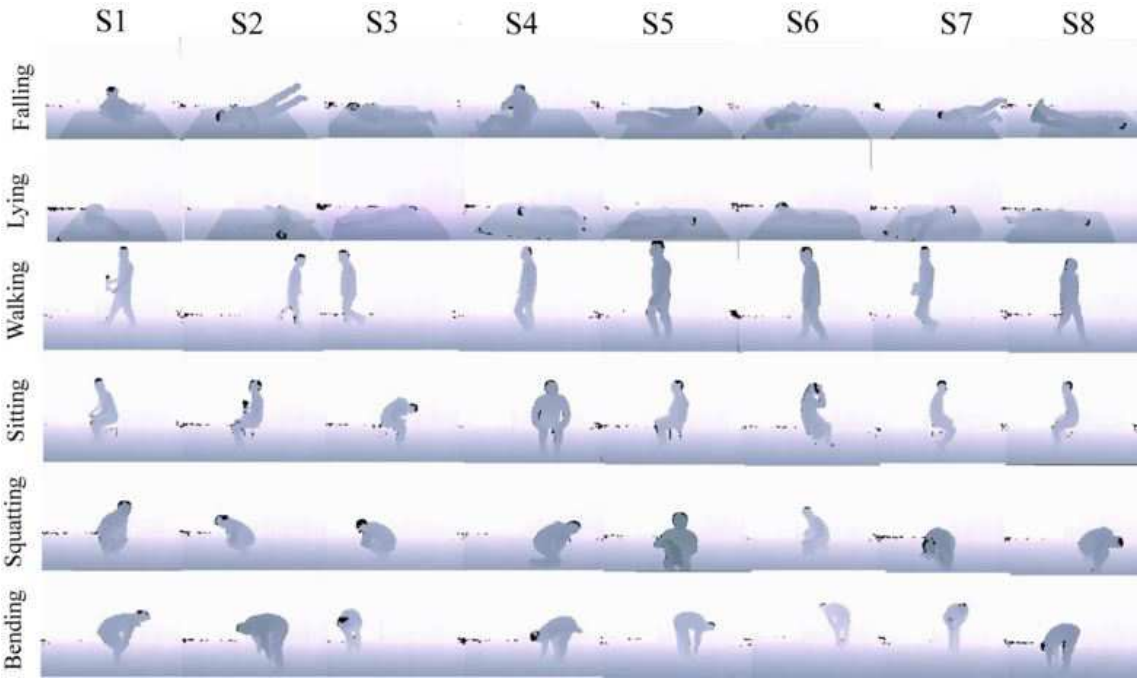

Figure 4.1: **The sample frames of SDUFall dataset**

Chapter 4 Experimental Results

4.1 Dataset

We conduct our experiments with Dog Centric Activity Dataset [**?**]. This dataset contains 10 different types of activities, including car, drink, feed, look left, look right, pet, ball play, shake, sniff and walk. Each activity may involves local or global motion. There were four different dogs which were equipped with a GoPro camera on their back.

4.2 Fall detection dataset

$$\begin{aligned} precision &= TP/(TP+FP) \\ sensitivity &= TP/(TP+FN) \\ specificity &= TN/(TN+FP) \end{aligned} \tag{4.1}$$

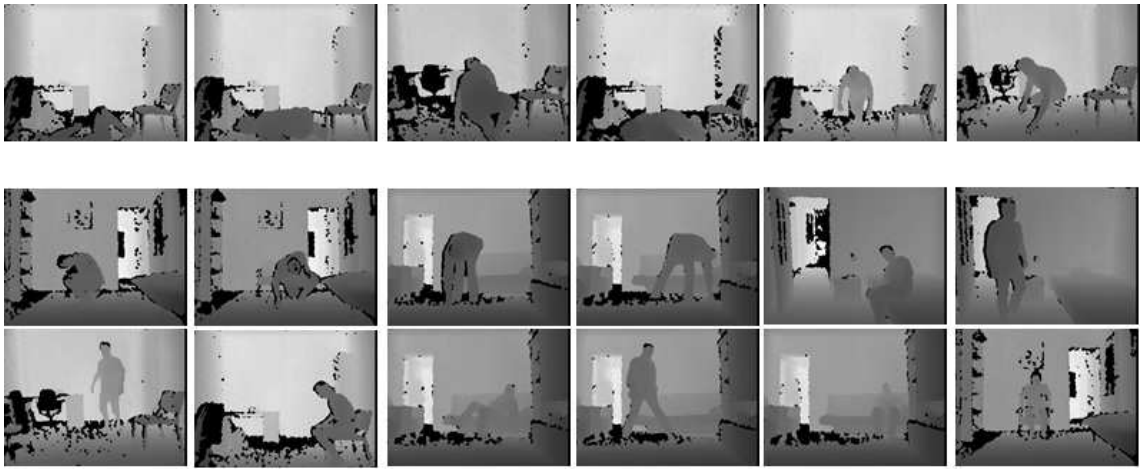

Figure 4.2: **The sample frames of UR fall detection dataset**

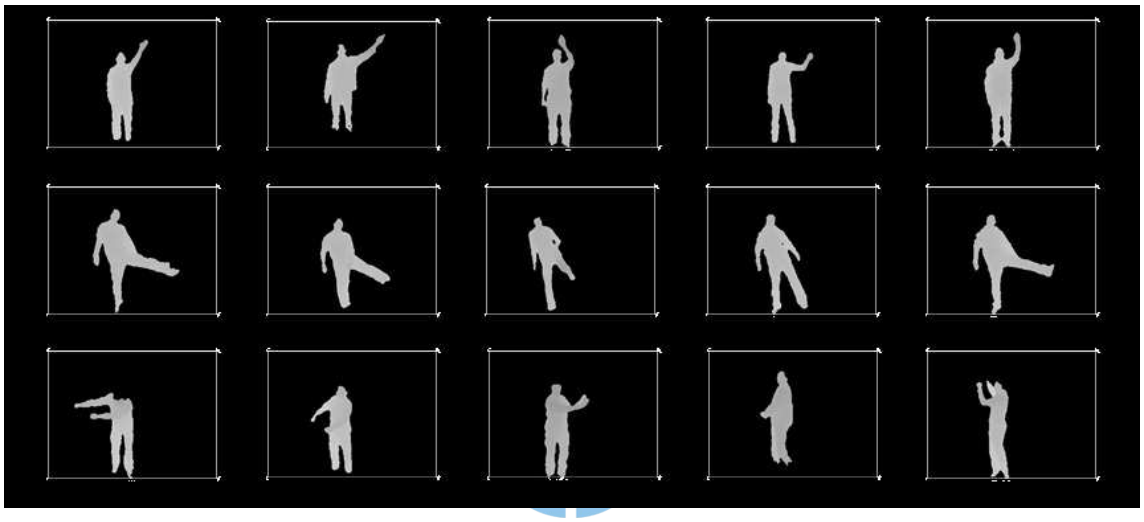

Figure 4.3: **The sample frames of MSR Action3D dataset.The top line is high wave action,middle is side kick action and bottom is golf swing**

4.3 Action Recognition dataset

Table 4.8 shows the result we compare to [**?**].The [**?**] using DMM as their descriptor and apply SVM to classify.

4.4 Fail Case

Table 4.1: Define TP,FP,TN and FN

	Positive(True label)	Negative(True label)
Positive(prediction)	TP(True Positive)	FP(False Positive)
Negative(prediction)	TN(True Negative)	FN(False Negative)

Table 4.2: change DMM gap and weight

Method	Accuracy(%)	Fall vs nonfall accuracy (%)	Sensitivity(%)	Specificity(%)
Before change	95.19	98.70	94.44	99.56
After change	96.67	99.26	95.56	100

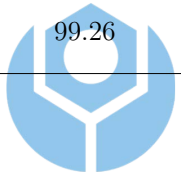

Table 4.3: Comparison with [?] methods on SDUFall Dataset

Method	Accuracy(%)	Sensitivity(%)	Specificity(%)
SVM	63.12	57.40	75.78
ELM	84.36	86.16	76.51
PSO-ELM	85.34	89.96	77.07
VPSO-ELM	86.83	91.15	77.14
Ours	98.70	94.44	99.56

Table 4.4: Fall vs non-Fall comparison with [?] methods on SDUFall Dataset

Method	Accuracy(%)
FV-ELM	89.84
FV-SVM	88.83
FV-K-NN	86.33
FV-NN	88.01
Ours	98.70

Table 4.5: overall class comparison with [?] methods on SDUFall Dataset

Method	Accuracy(%)
BOW-SVM	32.23
BOW-ELM	51.32
BOW-VPSO-ELM	54.17
FV-SVM	64.67
Ours	95.19

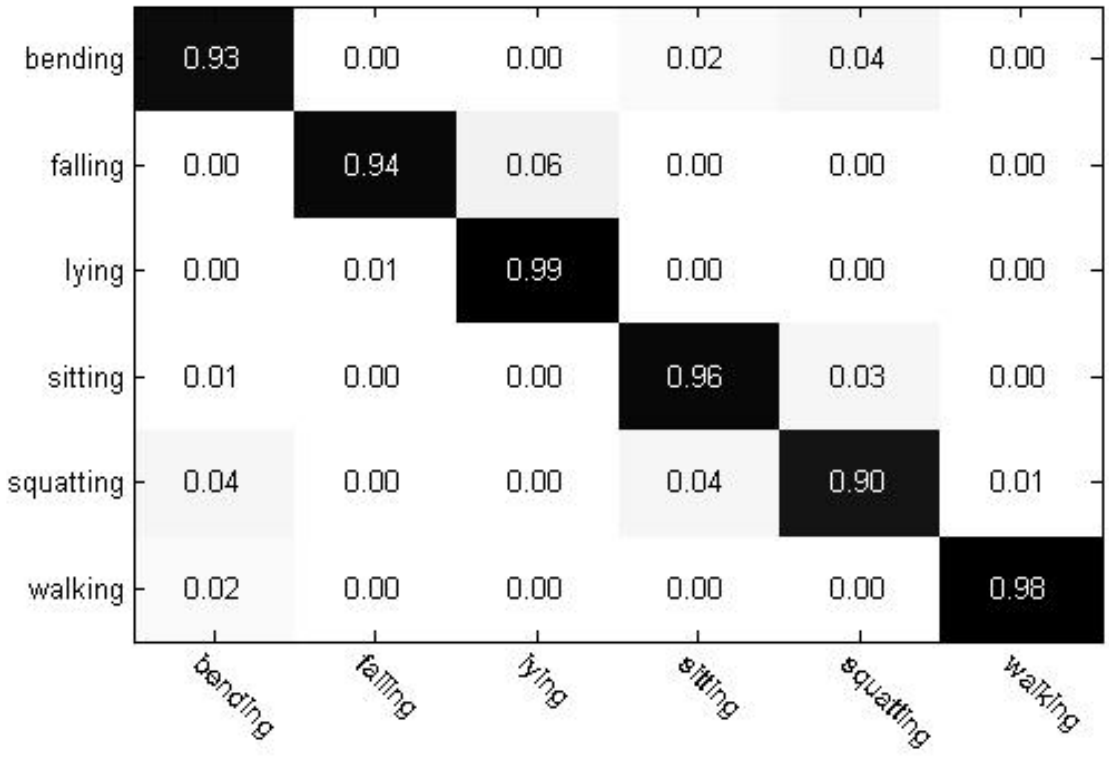

Figure 4.4: **Confusion matrix on SDUFall dataset**

Table 4.6: Comparison with [?] methods on URFallDetection Dataset

Method	Accuracy(%)	Precision(%)	Sensitivity(%)	Specificity(%)
SVM-depth only	90.00	83.30	100.00	80.00
SVM-depth +acc	98.33	96.77	100.00	96.67
Threshold UFT	95.00	90.91	100.00	90.00
Threshold LFT	86.67	82.35	93.33	80.00
Ours	100.00	100.00	100.00	100.00

Table 4.7: Action subset in [?]

AS1(Action Set 1)	AS2(Action Set 2)	AS3(Action Set 3)
Horizontal Wave	High Wave	High Throw
Hammer	Hand Catch	Forward Kick
Forward Punch	Draw X	Side Kick
High Throw	Draw Tick	Jogging
Hand Clap	Draw Circle	Tennis Swing
Bend	Hands Wave	Tennis Serve
Tennis Serve	Forward Kick	Golf Swing
Pick up Throw	Side Boxing	Pick up Throw

Table 4.8: Comparison with [?] methods on MSR Action3D Dataset

Method	3D Silhouettes(%)	EigenJoints(%)	DMM+SVM(%)	Ours(%)
AS1one	89.5	94.7	97.3	**97.8**
AS2one	89.0	**95.4**	92.2	95.0
AS3one	96.3	97.3	**98.0**	97.4
AS1two	93.4	97.3	98.7	**98.7**
AS2two	92.9	98.7	94.7	**98.7**
AS3two	96.3	97.3	98.7	**98.7**

Table 4.9: Comparison with [?] methods on MSR Action3D Dataset

Method	JP(%)	RJP(%)	JA(%)	BPL(%)	LG(%)	Ours(%)
AS1	91.65	92.15	85.80	83.87	95.29	94.45
AS2	75.36	79.24	65.47	75.23	83.87	80.93
AS3	94.64	93.31	94.22	91.54	98.22	97.20
Average	87.22	88.23	81.83	83.54	92.46	90.86

Table 4.10: Comparison with [?] methods and state of the art on MSR Action3D Dataset

Method	Average(%)
Histograms of 3D joints	78.97
EigenJoints	82.30
Joint angle similarities	83.53
Spatial and temporal part-set	90.22
Covariance descriptors	90.53
Random forest	90.90
Lie group	92.46
Ours	90.86

References

[1] B. Fernando, E. Gavves, M. José Oramas, A. Ghodrati, and T. Tuytelaars, "Modeling video evolution for action recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June, pp. 5378–5387, 2015.