Estimating Head Motion from Egocentric Vision

Satoshi Tsutsui, Sven Bambach, David Crandall
School of Informatics, Computing and Engineering
Indiana University
Bloomington, Indiana, USA

Chen Yu
Psychological and Brain Sciences
Indiana University
Bloomington, Indiana, USA

ABSTRACT

Lightweight wearable cameras record video from a "first-person" perspective, capturing the visual world of the wearer in everyday contexts. These videos are a rich source of information about people's behaviors and interactions. In this paper, we investigate using head-mounted cameras to estimate head (camera) motion, which could be used to infer non-verbal behaviors such as head turns and nodding in multimodal interactions. We propose Convolutional Neural Networks (CNNs) that combine raw images and optical flow fields to distinguish global ego-motion from moving objects in a scene. Our results suggest that CNNs do not directly learn useful visual features with end-to-end training from raw images alone; a better approach is to extract optical flow explicitly and then train CNNs to integrate flow and visual information.

ACM Reference Format:
Satoshi Tsutsui, Sven Bambach, David Crandall and Chen Yu. 2018. Estimating Head Motion from Egocentric Vision. In *2018 International Conference on Multimodal Interaction (ICMI '18), October 16–20, 2018, Boulder, CO, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3242969.3242982

1 INTRODUCTION

To interact effectively, an embodied agent needs to know its real-time position with respect to the rest of the physical world. Humans and robots have various sensors to do this, from fluid chambers in the human ear that sense balance, to accelerometers and gyroscopes in robots and other devices. But visual information is a particularly informative and fine-grained source of evidence, which is why visual Simultaneous Mapping and Localization (SLAM) is well-studied in robotics [16] and modern augmented reality systems (e.g., Google's ARCore [9]) often include visual odometry.

But SLAM and visual odometry assume that the surrounding scene is mostly static, and require complex, compute-intensive algorithms to infer pose based on reasoning about scene geometry. These assumptions are often unrealistic in highly dynamic environments. The recent availability of lightweight, wearable cameras allows for collecting video from a "first-person" perspective, capturing the visual world of the wearer in everyday interactive contexts. Such devices include Google Glass, GoPro Hero, Oculus, and Snapchat Spectacles, all of which are worn on the body and "look out" at the world, naturally gathering visual information from everyday interactions. Egocentric vision thus provides a unique perspective of the world that is inherently human-centric, and provides fine-grained, dense information about the camera wearer's actions [2, 14], gaze [13], trajectory [26], interactions [1], etc.

Figure 1: We estimate angular head motion in egocentric video, using both visual content and optical flow.

In this paper, we investigate how to exploit egocentric vision to infer multimodal behaviors from people wearing head-mounted cameras, and more specifically, to estimate head motion. From a practical perspective, inferring head motion is useful in a variety of pervasive computing applications. Non-verbal behaviors that induce head motion (e.g., head turns and nodding) are important in human-human and human-robot communication [8, 18, 24, 27]. Head motion is a proxy for eye movement and attention [23], since eye and head movements are highly coordinated [19] and eye gaze is usually centered in the egocentric view [3]. Finally, head motion can signal a person's internal states and be used to predict influential statements in group discussions [18], for example.

Of course, head motion can be directly measured with motion tracking sensors, but the alternative we consider here — inferring head motion using video from head-mounted cameras — has several potential advantages. First, it allows multimodal behaviors (vision, motion, etc.) to be acquired using only a camera, without the cost of multiple sensors. Second, using video avoids the need to synchronize the data streams from multiple sensors. Finally, it allows head motion to be retroactively inferred for existing videos.

We estimate angular speed (magnitude of angular velocity) in particular. We primarily use optical flow, which is typically calculated by comparing adjacent video frames to estimate apparent motion on a pixel-by-pixel basis. The optical flow field for a head-mounted camera is created through a combination of the motion of the head itself and the motion of individual objects within the scene. If an object is stationary and at a known distance from the camera, then estimating camera motion is straightforward based on the optical flow of that object's pixels. But in real-world, dynamic scenes, some objects are stationary, some move in predictable patterns (cars driving down a street), and others move highly unpredictably (people's hands). Meanwhile, the distances to some objects can be easily estimated (e.g., one's own hands are 1-3 feet away) while others may be quite unpredictable. Finally, optical flow estimates are much more reliable for objects with distinctive appearances, compared to those with uniform textures or repeated patterns. The challenge of using optical flow to estimate head motion is thus to

ICMI '18, October 16–20, 2018, Boulder, CO, USA

ACM ISBN 978-1-4503-5692-3/18/10…$15.00
https://doi.org/10.1145/3242969.3242982

separate global pixel displacements created by head movements from local displacements created by other objects and activities in a scene.

After a brief review of related work, we first report that the statistics of optical flow calculated on head-mounted camera video are highly correlated with head motion (Section 3), using head-mounted motion sensors as ground truth. We then propose two approaches to use Convolutional Neural Networks to estimate head motion: one that operates on pre-calculated optical flow fields (Section 3.1), and one that jointly estimates optical flow and head motion (Section 3.2). Our experiments on a challenging dataset show that the latter approach performs better than several baselines, by implicitly learning to distinguish regions with optical flow caused by ego-motion from those caused by other motion in the scene.

2 RELATED WORK

Estimating people's head motion from third-person views has been studied [17] but is fundamentally different from egocentric video [5] in which the head does not actually appear in the frame. Visual odometry and Simultaneous Localization and Matching (SLAM) [16] are more related, and often used in robotics to construct maps of the environment. These approaches assume scenes are mostly static and require fine-grained geometric information and reasoning, which may not be realistic for head-mounted cameras capturing interactions with other people, for example. Several papers have used optical flow features for first-person vision tasks such as activity recognition [12, 15, 20], but do not explicitly estimate camera motion. Perhaps most related to ours is the work of Li *et al.* [14], who do estimate head motion as a feature for egocentric activity recognition, but their method uses fine-grained geometric information (by computing homographies between frames).

3 ESTIMATING HEAD MOTION

For a given pair of video frames $(I_t, I_{t+\Delta t})$, we wish to estimate the angular head (camera) change in degrees per unit time — a single scalar that is the Euclidean norm of the 3d Euler angle rates (speeds in yaw, pitch, roll dimensions). As noted above, optical flow is only indirect evidence of head motion, since it is created by the motion of both the camera and the objects in the scene. If we know which objects are stationary in the scene and their distance (depth) from the camera, then we can use their optical flow to estimate camera (head) rotational motion, but these are strong assumptions. A more realistic assumption is that *most* of the pixels in the frame are background and thus stationary and at uniform depth, so that the dominant optical flow displacement vector correlates with head motion. Figure 2(b) confirms this empirically, plotting the mode of the distribution of optical flow speed versus the actual angular head motion (measured by a wearable sensor) for one randomly-chosen video from our dataset.[1] A strong correlation is evident, with Pearson correlation coefficient 0.62.

But despite the relatively strong correlation overall, the most frequent optical flow vector is often not an accurate measure of head speed: the dominant displacement may be created by a large moving foreground object (e.g., a hand), for example. Instead of estimating head motion based only on a summary statistic of optical flow, we hypothesize that a machine learning algorithm could learn which flow patterns and which particular objects tend to be reliable indicators of head motion. In particular, we use Convolutional Neural Networks (CNNs), the *de facto* standard machine learning model in computer vision. We normalize the range of the ground truth speed to be between 0 and 1, and train a model that predicts (regresses) this scalar value. We consider two CNN architectures, each having a single-node output layer with a sigmoid activation function, but two different ways of considering optical flow information: (1) compute optical flow in preprocessing and input an encoding of optical flow to the CNN, or (2) input pairs of frames, relying on the network itself to learn to infer optical flow features.

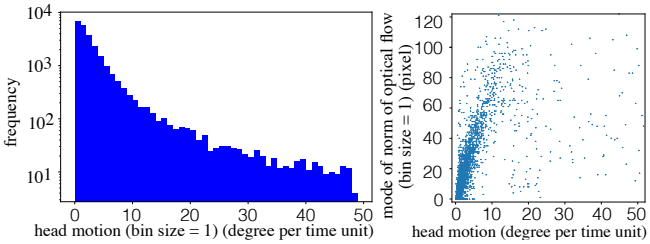


Figure 2: Statistics of head-camera motion: (a) Distribution of ground truth head rotation speeds, and (b) actual head motion speed versus mode of optical flow vector lengths.

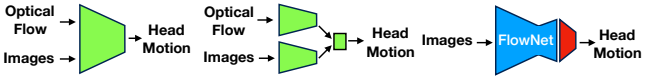


Figure 3: Overview of network architectures: (a) VGG with early fusion, (b) VGG with late fusion, and (c) FlowNet⁺.

3.1 Approach 1: CNN on preprocessed flow

Our first approach computes dense optical flow between adjacent frames and then passes them as input to a neural network. Our *base CNN* is modified from VGG Configuration A [22]: we kept the convolutional layers (the first 12 layers), but added a fully connected layer with 128 hidden nodes and ReLU activations and an output layer that produces a single scalar value with sigmoid activation. We use batch normalization [10] in the convolution layers.

We tried presenting the flow and visual information to the network in various ways. For pair of frames $(I_t, I_{t+\Delta t})$, **Flow** simply feeds the dense optical flow F_t into the base CNN, encoded as a 2-channel "image" in which the two channels correspond to the x and y components of the displacement vectors. **Flow+Visual** concatenates optical flow and the first input image of the pair, yielding a 5-channel input, while **Flow+Double Visual** combines optical flow and both input images, creating a single 8-channel input. Finally, we tried a late-fusion approach: **Flow+Visual (Late Fusion)** feeds optical flow and the image into separate streams of the base CNN, extracts the output of the last convolution layer of each, and passes their concatenation to the fully connected network (Figure 3(b)).

[1] In particular, for each pair of frames I_t and $I_{t+\Delta T}$ where $\Delta T = 0.17$sec, we calculated optical flow [25], computed the magnitude of each pixel's displacement vector, rounded the magnitudes to integers, and selected the mode (most frequent discretized speed).

3.2 Approach 2: CNN on image pairs

In theory, precomputing optical flow is redundant because a pair of images already contains the flow information, so our second approach investigates CNNs that operates on just the image pair. A simple way is to train a standard CNN to estimate head motion from the visual information in a single six-channel image containing the three RGB planes of each of the two frames, assuming that the CNN can extract its own representation of visual change without explicitly being trained to compute optical flow. We do this with the base CNN in Approach 1, and call it **Double Visual**.

Alternatively, we can use a CNN designed to extract optical flow. We use FlowNetS [6], a CNN to estimate the flow from a concatenated 6-channel input image, and add several additional layers to estimate head motion: three 3×3 convolution layers (having 32, 64, and 128 filters) interspersed with two 2×2 max-pooling layers, and two fully-connected layers similar to those in Section 3.1 (one of which produces a 128-d vector with ReLU, and another that outputs a scalar using a sigmoid). We call this modified architecture *FlowNet*$^+$ and illustrate it in Figure 3(c).

We investigated several ways of training this new model. **Tune whole FlowNet$^+$** simply trains the network from scratch on our training dataset. **Tune last FlowNet$^+$** initializes FlowNet weights with those pretrained for optical flow estimation, and in training updates only the weights for the layers we added. **Fine-tune FlowNet$^+$** begins with the FlowNet weights, but updates all weights (in all convolutional and fully-connected layers) during training on our dataset for head speed detection. We expect this to allow the whole network to be optimized for best performance on our task and dataset, learning to integrate visual and flow information most effectively. Finally, **Tune FlowNet$^+$ only** begins with the weights of **Tune last FlowNet$^+$**, but only fine-tunes the FlowNet weights (not those of the additional layers we added).

4 EXPERIMENTS

We tested our approach for head motion estimation using a dataset that was collected for a psychological study of child-parent interactions [4]. Parents and children sat across from each other at a table in a lab, and played freely with colorful toys. The children and parents wore head-mounted cameras and head position sensors. The lab was draped in white, which in the original study was designed to avoid distracting the children, but creates a scenario that is challenging for optical flow due to a lack of distinctive visual landmarks. Moreover, data from children is likely to be more challenging than that from adults because children's behavior is generally less predictable. We have three child-parent pairs with four distinct videos for each, and use three for train and one for test. For all experiments, we set $\Delta t = 0.17$ seconds (5 frames) and have 24,950 image pairs $(I_t, I_{t+\Delta t})$ for training and 8,797 for testing.

4.1 Evaluation Metrics

A natural evaluation metric is sum-squared error between the predicted and ground truth rotation speeds; another is to treat the problem as a classification task by binning the ground truth into discrete categories and reporting classification accuracy. However, the observed distribution of head speeds is highly non-uniform, as shown in Figure 2(a), which means that a trivial estimator that always predicts zero head motion could achieve high accuracy under these metrics. As a stricter metric, we partition pairs of frames into ten subsets according to ground truth speed in 5° increments up to 50° (e.g., all frames having ground truth motion in $[0°, 5°)$, $[5°, 10°)$, etc.), compute the mean absolute error within each subset, and average these 10 errors. We call this the *weighted mean error*.

4.2 Implementation and baselines

We resized video frames to 128×128 pixels before passing them to our networks. During training, we minimized L2 loss (squared error) using the Adam optimizer with its default hyper-parameters [11], except that we divided the learning rate by a factor of 100 during fine-tuning. We used a batch size of 32, and used undersampling with a bin size of 5 degrees to address the class imbalance problem.

We also implemented two baseline techniques. **Linear Regression** fits a line to the scatter plot in Figure 2(b), and then uses the mode of the optical flow magnitude distribution to regress head motion. **Camera Geometry** estimates point correspondences between I_t and $I_{t+\Delta t}$, then estimates the camera angle change based on geometric reasoning. We follow the process in previous work [14], and use ORB [16] for point detection and matching and RANSAC [7] for computing homographies. Note that this method requires knowing the intrinsic camera parameters (e.g., focal length).

4.3 Results and Discussion

Table 1 presents results of our experiments using mean and standard deviation of absolute error within each 5° ground truth bin, as well as overall weighted mean error. We see that both linear regression and camera geometry have much larger weighted mean errors (21.3 and 28.1, respectively) compared to the CNN based approaches (e.g., 6.1 for Fine-tune FlowNet$^+$). Interestingly, the camera geometry baseline has very high standard deviations on absolute error; this is likely caused by image pairs for which few distinctive feature points are visible and a homography cannot be calculated.

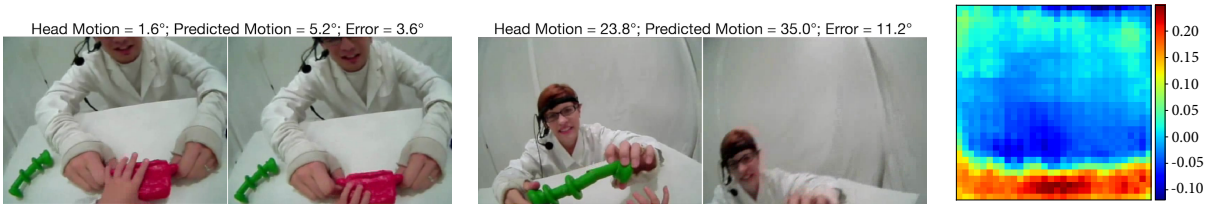
For Approach 1 based on VGG, the model with only optical flow performs better than the models with visual information (Flow+ Visual, Flow+ Double Visual), so adding visual information seems to confuse the CNN. The simple concatenation (Flow+Visual) is more effective than late fusion, potentially because the task needs to integrate lower level features (including optical flow) but the later CNN layers only see high level features [28]. Double Visual has the highest weighted mean error. This is evidence that the base CNN is not learning information equivalent to optical flow, which suggests the importance of extracting optical flow explicitly.

For the FlowNet-based approaches, simply training the whole network from scratch performs worse; this is consistent with our hypothesis that we need to teach the network to extract the optical flow explicitly. Training the last layers of FlowNet$^+$ yielded a similar performance to the Flow only model for the base CNN, which makes sense since this is equivalent to using Flow only. Jointly fine-tuning FlowNet and the last layers (Fine-tune FlowNet$^+$) gave the best performance, suggesting that the network is finally able to use both optical flow and visual information effectively.

These experiments suggest that optical flow is critical evidence for head motion prediction, and that it is important to incorporate the flow explicitly (either fed as preprocessed features or using

Table 1: Head motion estimation error compared to ground truth, in terms of mean error ± standard deviation.

	Ground truth range (number of samples) (Units: Degrees per five frames ≈ 0.17s)										weighted mean error
	0-5 (7261)	5-10 (790)	10-15 (260)	15-20 (131)	20-25 (97)	25-30 (61)	30-35 (43)	35-40 (38)	40-45 (37)	45-50 (19)	
Linear Regression	2.0±1.0	2.9±1.4	8.1±1.6	13.4±1.5	18.7±1.4	23.8±1.5	28.3±1.4	33.6±1.7	38.6±1.5	43.6±1.3	21.3
Camera Geometry [14]	1.8±9.4	4.2±16.7	9.3±25.6	11.3±21.7	14.1±28.0	28.0±39.0	43.6±50.3	42.5±44.1	64.5±50.8	61.4±51.8	28.1
Flow	1.5±2.7	5.0±5.0	6.4±6.2	7.1±6.3	9.0±6.0	8.8±5.3	8.1±6.0	8.7±6.5	9.6±7.8	10.9±4.9	7.5
Flow+Visual	2.7±3.6	5.5±5.5	6.1±5.3	6.9±5.4	6.8±5.1	7.0±4.4	8.8±5.6	8.1±5.7	10.8±7.4	16.4±6.7	7.9
Flow+Visual (Late Fusion)	4.8±3.7	8.1±5.4	7.9±5.8	8.2±5.3	8.1±4.9	6.6±4.6	6.4±5.0	7.4±5.3	10.9±7.1	13.8±5.6	8.2
Flow+Double Visual	2.6±3.6	6.7±5.2	6.8±5.8	6.7±6.3	7.3±5.4	6.7±4.7	6.7±5.5	8.4±5.8	11.4±6.4	16.3±6.6	8.0
Double Visual	12.6±7.7	12.4±8.0	11.2±6.9	10.1±6.4	6.7±4.4	6.2±3.6	4.5±3.9	7.3±5.9	11.4±5.8	16.6±4.6	9.9
Tune whole FlowNet$^+$	23.3±1.2	18.4±1.4	13.1±1.4	7.9±1.4	2.6±1.4	2.5±1.5	7.0±1.4	12.3±1.6	17.5±1.3	22.3±1.3	12.7
Tune last FlowNet$^+$	3.8±4.9	5.6±6.4	7.2±6.6	7.7±6.6	7.5±6.0	7.5±5.1	6.7±4.6	7.8±4.9	7.0±5.6	11.5±7.8	7.2
Fine-tune FlowNet$^+$	2.4±3.8	3.7±5.0	5.0±5.7	4.8±4.6	5.9±4.5	5.4±4.4	6.8±5.1	7.8±5.0	8.2±5.2	11.2±7.2	**6.1**
Tune FlowNet$^+$ only	2.1±2.4	3.1±3.5	4.7±5.2	5.5±4.5	6.2±5.0	6.3±4.6	6.5±5.1	9.2±5.8	7.8±6.1	14.1±9.7	6.5



(a) (b) (c)

Figure 4: (a) and (b): Results on two sample image pairs. (c): Visualization of the differences in optical flow produced by networks tuned and not tuned on head motion estimation. Red shows generally unreliable regions for estimating ego-motion.

transfer learning from the optical flow extractor), because neural networks do not internally learn the flow equivalent features if we just use end-to-end training from visual cues. Similar observations can be found in the computer vision literature for third-person videos, where optical flow is extracted in a preprocessing step and found to be a critical feature [21].

Figures 4(a) and 4(b) show results generated by FlowNet$^+$ on two sample image pairs. Pair (a) is a case of an accurate estimation, with error of less than 4°, while (b) is a case in which the network over-estimated actual head motion by over 10°.

Which visual features are these networks cuing on? To help answer this, we compared the optical flow output of Tune FlowNet$^+$ Only with that of Tune last FlowNet$^+$. The only difference between these two techniques is that the former has been allowed to modify its optical flow calculation weights to produce optical flows that are more reliable for head motion estimation. This means that the difference between the optical flows produced by the two networks should reveal image regions that the network has chosen to remove because they are thought to be unreliable. Figure 4(c) shows a heatmap of the mean differences of optical flow magnitudes across all 8,797 test images at each spatial position within the frame. We see that the network often ignores optical flow near the bottom of the image, for example, perhaps because the hands are often located there and are particularly poor estimators of head motion.

5 CONCLUSION AND DISCUSSION

We have focused on estimating head motion in egocentric video. We observed a high correlation between optical flow and head motion, so we investigated several CNN-based methods for estimating head motion based on combinations of optical flow and visual information. We achieved the best performance by fine-tuning FlowNet$^+$, a network that is pre-trained for optical flow estimation. We found that CNNs trained only from visual cues do not work as well as when optical flow is explicitly provided, which is more evidence that optical flow is critical to this task. We also demonstrated that partially fine-tuning FlowNet$^+$ could help reveal which part(s) of the video tend to yield reliable evidence for ego-motion estimation.

A limitation of our work is that we only estimate head motion speed instead of velocity; while speed alone is sufficient for many applications, others may require direction as well. In future work, our network could be extended to estimate velocity by replacing the last layer with a triplet (yaw, pitch, roll) regression. Other future work is to apply our head motion estimation techniques for a particular task such as communication behavior analysis. Moreover, we are interested in learning object-level knowledge for ego- and non-ego motion, e.g. discovering that hands often create non-ego motion while tables are less likely to do so.

Acknowledgments. This work was supported by the National Science Foundation (CAREER IIS-1253549), the National Institutes of Health (R01 HD074601), and the IU Office of the Vice Provost for Research, the College of Arts and Sciences, and the School of Informatics, Computing, and Engineering through the Emerging Areas of Research Project "Learning: Brains, Machines, and Children." We thank Melissa Elston, Melissa Hall, Steven Elmlinger, Seth Foster, and Charlene Ty for assisting with data collection, Shujon Naha and Zehua Zhang for helpful discussions, and Yi Li for refining Figure 2.

REFERENCES

[1] Maedeh Aghaei. 2017. Social signal extraction from egocentric photo-streams. In *International Conference on Multimodal Interaction (ICMI)*.

[2] Sven Bambach, David J Crandall, and Chen Yu. 2015. Viewpoint integration for hand-based recognition of social interactions from a first-person view. In *International Conference on Multimodal Interaction (ICMI)*.

[3] Sven Bambach, John Franchak, David Crandall, and Chen Yu. 2014. Detecting hands in children's egocentric views to understand embodied attention during social interaction. In *Annual Meeting of the Cognitive Science Society (CogSci)*.

[4] Sven Bambach, Linda B Smith, David J Crandall, and Chen Yu. 2016. Objects in the center: How the infant's body constrains infant scenes. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL)*.

[5] Alejandro Betancourt, Pietro Morerio, Carlo S Regazzoni, and Matthias Rauterberg. 2015. The evolution of first person vision methods: A survey. *IEEE Transactions on Circuits and Systems for Video Technology* 25, 5 (2015), 744–760.

[6] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. 2015. Flownet: Learning optical flow with convolutional networks. In *International Conference on Computer Vision (ICCV)*.

[7] Martin A Fischler and Robert C Bolles. 1987. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In *Readings in computer vision*. Elsevier, 726–740.

[8] Jeffrey M Girard. 2014. Perceptions of interpersonal behavior are influenced by gender, facial expression intensity, and head pose. In *International Conference on Multimodal Interaction (ICMI)*.

[9] Google. [n. d.]. ARCore Overview. https://developers.google.com/ar/discover/. Accessed: 2018-05-01.

[10] Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*.

[11] Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

[12] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. 2011. Fast unsupervised ego-action learning for first-person sports videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[13] Yin Li, Alireza Fathi, and James M Rehg. 2013. Learning to predict gaze in egocentric video. In *International Conference on Computer Vision (ICCV)*.

[14] Yin Li, Zhefan Ye, and James M Rehg. 2015. Delving into egocentric actions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[15] Minghuang Ma, Haoqi Fan, and Kris M Kitani. 2016. Going deeper into first-person activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[16] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics* 31, 5 (2015), 1147–1163.

[17] Erik Murphy-Chutorian and Mohan Manubhai Trivedi. 2009. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 4 (2009), 607–626.

[18] Fumio Nihei, Yukiko I Nakano, Yuki Hayashi, Hung-Hsuan Hung, and Shogo Okada. 2014. Predicting influential statements in group discussions using speech and head motion information. In *International Conference on Multimodal Interaction (ICMI)*.

[19] Jeff Pelz, Mary Hayhoe, and Russ Loeber. 2001. The coordination of eye, head, and hand movements in a natural task. *Experimental Brain Research* 139, 3 (2001), 266–277.

[20] Yair Poleg, Ariel Ephrat, Shmuel Peleg, and Chetan Arora. 2016. Compact CNN for Indexing Egocentric Videos. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE.

[21] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems (NIPS)*.

[22] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*.

[23] Oleg Špakov, Poika Isokoski, Jari Kangas, Jussi Rantala, Deepak Akkil, and Roope Raisamo. 2016. Comparison of three implementations of HeadTurn: a multimodal interaction technique with gaze and head turns. In *International Conference on Multimodal Interaction (ICMI)*.

[24] Ramanathan Subramanian, Yan Yan, Jacopo Staiano, Oswald Lanz, and Nicu Sebe. 2013. On the relationship between head pose, social attention and personality prediction for unstructured and dynamic group interactions. In *International conference on Multimodal Interaction (ICMI)*.

[25] Deqing Sun, Stefan Roth, and Michael Black. 2010. Secrets of optical flow estimation and their principles. In *Conference Computer Vision and Pattern Recognition (CVPR)*.

[26] Takuma Yagi, Karttikeya Mangalam, Ryo Yonetani, and Yoichi Sato. 2018. Future Person Localization in First-Person Videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[27] Akiko Yamazaki, Keiichi Yamazaki, Takaya Ohyama, Yoshinori Kobayashi, and Yoshinori Kuno. 2012. A techno-sociological solution for designing a museum guide robot: regarding choosing an appropriate visitor. In *Human-Robot Interaction (HRI), 2012 7th ACM/IEEE International Conference on*. IEEE, 309–316.

[28] Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*.