Object Recognition in Egocentric Images using Spatial Transformer Networks

Desh Raj 130101018
Sumeet Ranka 130101062
Akashdeep Goswami 130101088
Siddharth Kumar 130101072
Samyak Kumbhalwar 130101041

Abstract

Object detection and recognition in images and videos captured by body-mounted cameras are a significant underlying problem in many systems such as health monitoring, kitchen assistance, etc. In this paper, we propose a novel solution to the inherent distortion in these images in the form of spatial transformer networks (STN). We argue that the power of STNs arising from their ability to model any transformed grid, affine or otherwise, combined with the ability of convolutional neural networks to recognize local featues, makes them well-suited to attend to the important objects in an egocentric image, regardless of the clutter surrounding it. We extensively evaluate our proposed method on two popular datasets - the Georgia Tech Egocentric Activities (GTEA) data, and the Intel Egocentric Vision data. All the results clearly demonstrate that our proposed method achieves better performance than benchline models and is significantly close to complex state-of-the-art methods.

1. Introduction

In recent years, advances in camera technology and storage have made it possible to capture high-resolution images and videos using pocket-sized devices, and store large amounts of such data for processing afterwards. With such an advancement, computer vision has found increased applicability in domains such as elderly health monitoring, live assistance (e.g. kitchen assistance) systems, among others through the use of body-mounted or head-mounted cameras.

In conventional object recognition tasks, images are usually captured from a static camera in a brightly illuminated environment [6, 20]. However, in an egocentric setting, such an assumption is invalid because the camera is mounted on a mobile subject. Due to this instability and low illumination, objects in the image may appear out of focus. Further, due to the presence of multiple smaller objects in the subject's field of view, detecting local features may be extremely difficult.

For this reason, most of the existing methods which demonstrate high recognition rate with normal images fail miserably with egocentric images [22]. For instance, the recognition accuracy of SIFT matching [19] and latent HOG methods [4, 7] for handheld objects was found to be 33% and 64%, respectively.

In the last decade, deep learning methods such as convolutional neural networks (CNN) [15] have proven to be extremely efficient for vision problems since they exploit the locality of reference that is common with image data sets

[1, 10, 25]. CNNs usually have multiple layers of convolution with a max pooling layer, which provides some spatial invariance in the form of translational variability.

However, the spatial invariance provided by CNNs is limited in scope, since it takes into account only translational variations [3, 18]. If the image contains scaled, rotated, or spline-transformed objects, the convolution layers may require a large number of epochs to converge during backpropagation. This limitation has been explored by various researchers, and hence there have been various approaches for modelling transformations with neural networks. These include a generative model which can learn to generate transformed images of objects by composing parts [9, 28], constructing filter banks of transformed networks [12, 26], and neural networks with selective attention [1, 24].

In the seminal work by Jaderberg et al. [11], a novel method called the spatial transformer network (STN) has been proposed to alleviate the issue of transform invariance. This is a differentiable module which applies a spatial transformation to a feature map during a single forward pass, where the transformation is conditioned on the particular input, producing a single output feature map. The fundamental usefulness of STNs lies in the fact that they provide abstraction from the other layers in deep models. For instance, these networks can be plugged in between any two layers of a fully-connected network (FCN), or a CNN. They can also be initialized independently for different channels of the input. Further, multiple layers of STN may be used within any network. Apart from transforming images to increase classification accuracy, STNs may also be used for tasks that require an attention mechanism, and training them is achievable purely with backpropagation.

In addition to integrating them with FCNs and CNNs, these transformers have also been implemented with various recurrent networks [27]. The advantage of using an RNN-STN is that it can attend to individual elements in a sequence, while keeping the training simplicity that is salient in other STN architectures. Most of the models have been tested using the MNIST handwritten character data set, and some have also been employed on other popular data sets such as "street view house number" [21] and "fine-grained bird classification" [29].

In this work, we propose a novel application of STNs in the form of egocentric object recognition. Due to the inherent distortion present in egocentric images, they are well-suited for the transform-detect appraoch leveraged by STNs, and hence we argue that such a model may attend instinctively to the important objects in the image, without being distracted by background clutter. Our key contributions in this paper are as follows:

- We propose a new and natural domain of application for spatial transformer networks in the form of egocentric object recognition.
- We validate our arguments on 2 popular datasets, namely the Georgia Tech Egocentric Activities (GTEA) data, and the Intel Egocentric Vision (IEV) data.
- Our simple CNN-based model performs extremely well compared to existing appraoches that leverage complex feature engineering based on domain knowledge.

The remainder of the paper is organized as follows. In Section 2, we discuss existing empirical and deep learning approaches for egocentric object recognition. Section 3 then provides an overview of the egocentric object recognition problem and the STN architecture as described in [11]. We proposed our method for this problem in Section 4 and describe our data sets and experiments in Section 5. We conlude by providing possible explanations for these results and hypothesize further improvements in Section 6.

2. Related research

Most of the existing methods for object recognition rely upon videos captured from the egocentric perspective and complex computational vision algorithms for feature extraction. Because of the use of videos rather than images, the task of background subtraction [16, 2] is simplified since a mobile object can be easily distinguished from a static background.

In [22], the authors perform background subtraction by computing dense optical flow and fitting it into multiple affine layers. A max-margin classifier is then used to combine motion with empirical knowledge of object location and background movement as well as temporal cues of support region and color appearance.

Color and depth information has also been exploited for hand and object tracking through the use of Kinect-style cameras [17]. For object recognition, the authors used RGB-D kernels in conjugation with linear SVM classifiers. Further, [5] used a robust, unsupervised bottom-up segmentation method which exploits the structure of the egocentric domain to partition each frame into hand, object, and background categories.

Surprisingly, deep learning methods have failed to keep up with empirical methods in this particular problem. Although fine-grained ibject detection and semantic segmentation have been solved efficiently using convolutional neural networks [13, 8], distorted object recognition is a task which has not yet seen much success [14].

In this paper, we propose a method which is independent of empirical, domain-specific knowledge, but which demonstrates a commendable performance on egocentric images. In this regard, our method is comparable with the simplest deep learning models in terms of complexity, and yet the recognition rate is at par with approaches that involve a significant amount of feature engineering.

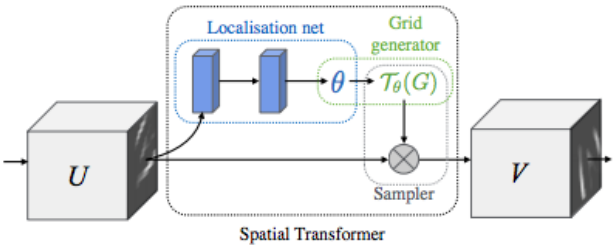

Figure 1. The architecture of a spatial transformer module. Source: [11]

3. Background

3.1. Egocentric object recognition

Object recognition is a process for identifying a specific object in a digital image or video. Object recognition algorithms rely on matching, learning, or pattern recognition algorithms using appearance-based or feature-based techniques. The term "egocentric" refers to any image or video captured from a first-person perspective, i.e., from a hand-held or body-mounted camera. Hence, egocentric object recognition is the task of identifying objects in such first-person images.

3.2. Spatial transformer networks (STN)

The spatial transformer mechanism consists of three modules as shown in Fig 1.

1. *Localisation network*: It takes the feature map as input and outputs the parameters of the spatial transformation that should be applied to it.

2. *Grid generator*: It uses the predicted transformation parameters to create a sampling grid, which is a set of points where the input feature map should be sampled to obtain the transformed image.

3. *Sampler*: It generates the image from the input feature map and the sampling grid.

The class of transformations τ_θ may contain different number of parameters, such as six in affine, eight in plane projective, piece-wise affine,

or thin-plate spline. Further, the sampling kernel used in the sampler may be selected from a large number of available choices, such as an integer or a bilinear sampling kernel.

Jaderberg et al. use bilinear sampling for the STN in a CNN model consisting of two max-pooling layers. The model is trained using back-propagation with stochastic gradient descent, with three weight layers in the classification network.

4. Proposed method

In this section, we describe our CNN-STN model for egocentric object recognition in detail. For this purpose, suppose our input feature map is $U \in \mathbb{R}^{H \times W \times C}$, where H and W are the height and width of the image, respectively, and C is the number of input channels. In the case of an RGB image, this value will be equal to 3. Fig. 2 shows an outline of our proposed method.

4.1. Localization network

This layer takes U as input and outputs θ, the parameters of the transformation τ_θ to be applied to the feature map. The size of θ varies according to the type of transformation performed. For instance, θ is equal to 6 in the case of an affine transformation.

Although a localization network can internally take any form such as a fully-connected or a convolutional network, we use the latter in our model to have shared weights so that the model learns and converges faster. It consists of a final regression layer to produce the transformation parameter θ.

4.2. Grid generator

This module takes the transformation parameter θ and a regular grid G as input and outputs a transformed grid, i.e. $\tau_\theta(G)$. The transformation is performed such that the new grid naturally attends to the object to be recognized, as shown in Fig. 3.

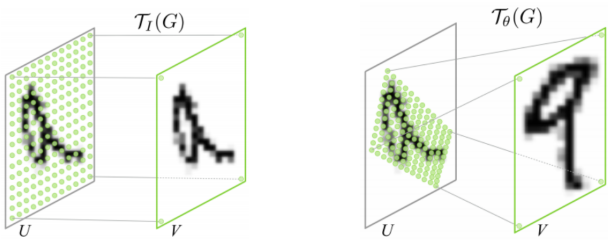

Figure 3. Two examples of applying the parameterised sampling grid to an image U producing the output V. Source: [11]

4.3. Sampler

It takes a set of sampling points $\tau_\theta(G)$, along with the input feature map U, and produces the sampled output feature map V.

Each (x_i^s,y_i^s) coordinate in $\tau_\theta(G)$ defines the spatial location in the input where a sampling kernel is applied to get the value at a particular pixel in the output V . This can be written as

$$V_i^c = \sum_n^H \sum_m^W U_{nm}^c k(x_i^s - m; \Phi_x) k(y_i^s - n; \Phi_y) \quad \forall i \in [1 \dots H'W'] \forall c \in [1 \dots C] \tag{1}$$

where Φ_x and Φ_y are the parameters of a generic sampling kernel $k()$ which defines the image interpolation (e.g. bilinear), U_{nm}^c is the value at location (n,m) in channel c of the input, and V_i^c is the output value for pixel i at location (x_i^t,y_i^t) in channel c. For sampling, any kernel such as an integer kerner or a bilinear sampling karnel may be used. In our model, we use bilinear interpolation for this purpose.

4.4. Convolutional layer

Once we have obtained the transformed image from the STN module, we perform a simple object classification task using a convolutional network. The convolutional layer identifies local features in an image such as edges, basic polygon shapes and boundaries. If we use m kernels of size $k \times k$, the nonlinear output generated by one of these m kernels is given as

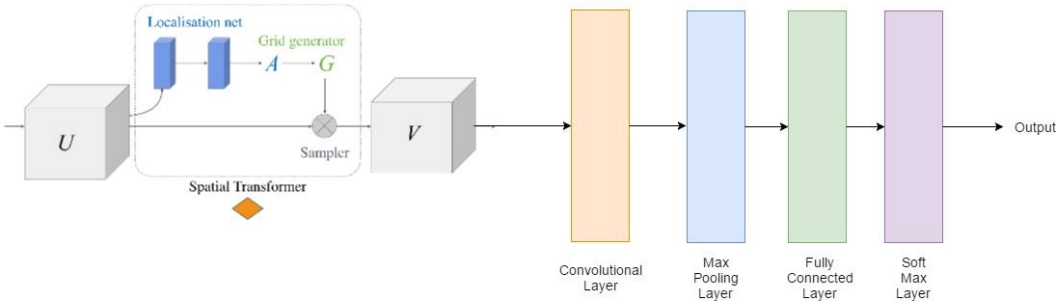

Figure 2. Our proposed CNN-STN architecture for egocentric object recognition.

$$y_{ij}^{l} = \sigma(\sum_{a}^{m}\sum_{b}^{m}\omega_{ab}y_{(i+a)(j+b)}^{l-1}) \quad (2)$$

4.5. Max pooling

After the convolutional layer, we apply a max-pooling layer to obtain global features from the image. In addition to extracting global features, this layer also reduces the dimensions of the image, thus ensuring that we have lesser number of weights to train.

4.6. Fully-connected layer

The fully-connected layer contains as many nodes as the number of object categories, and it outputs a score corresponding to each of these classes.

4.7. Softmax layer

We apply a softmax layer to obtain a probability distribution from the category scores provided by the fully connected layer using the equation

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e_{z_k}} \forall j = 1, \ldots, k. \quad (3)$$

where $\mathbf{z}$ is a K-dimensional vector representing the final probability distribution.

5. Experimental results

5.1. Description of datasets

We perform our experiments on two different datasets as described below.

5.1.1 Georgia Tech Egocentric Activities (GTEA) dataset [5]

This dataset contains 7 types of daily activities, each performed by 4 different subjects. The camera is mounted on a cap worn by the subject. For our experiments, we removed the frames containing no objects, and also scaled the images down from 720x405 to 20% proportions. We were ultimately left with 3047 images and 7 classes. The final count of instances corresponding to each object class is shown in Table 1.

Table 1. Description of GTEA dataset

Object class	No. of instances
Cheese	246
Chocolate	357
Coffee	393
Honey	324
Hotdog	168
Peanut	699
Tea	860

5.1.2 Intel Egocentric Vision (IEV) dataset [23]

This is a dataset for the recognition of handled objects using a wearable camera, collected by Matthai Philipose and Xiaofeng Ren at Intel Research Seattle. It includes ten video sequences from two human subjects manipulating 42 everyday objects. It contains 100,000+ frames, of which approximately 30% are background

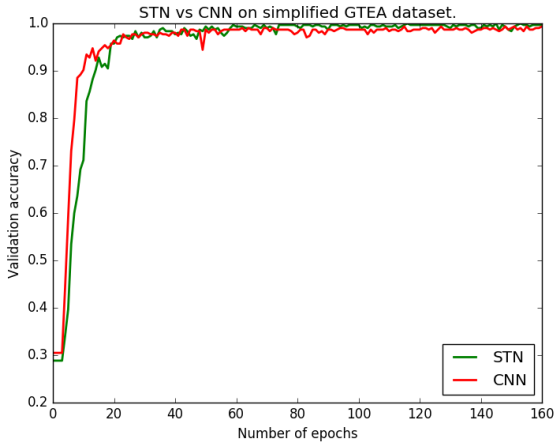

Figure 4. Validation accuracy at different time instances for the GTEA dataset classification using a CNN-STN model vs a simple CNN model.

Figure 5. GTEA dataset classification using a CNN-STN model (Output at epochs 15, 70 and 145).

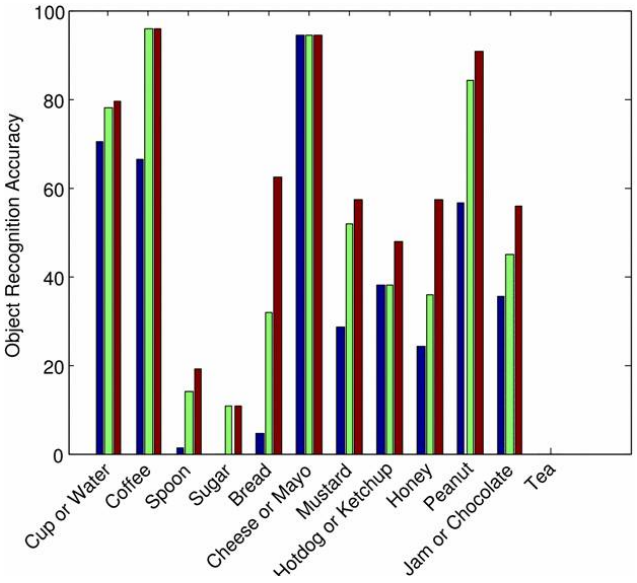

Figure 6. GTEA dataset object recognition results using [5]. Blue bars show how well the highest score detection in each frame matches the ground-truth object label. Green and red bars, depict these results for any of the 2 and 3 highest score detections.

frames, i.e., do not contain any object. We again removed these background frames for our evaluation purpose, and scale the images to 20% proportions to reduce the number of parameters.

5.2. Results on GTEA dataset

Since no formal train-test split was available, we randomly selected 80% of the images for training and the rest of the images were used for testing. The validation accuracy at different epochs for the STN model compared to a simple CNN model is shown in Fig. 4.

From the figure, we can observe that although initially our model learns slowly due to the presence of a larger number of parameters, it ultimately performs almost 2-3% better than a simple CNN model in the object classification task.

To understand the transformations made by the STN module, we observe its output at various time instances, such as epochs 15, 70, and 145, and the corresponding outputs are shown in Fig. 5. It is evident from these images that the STN module accurately focuses on the handheld object and avoids getting distracted by the background clutter. Due to this transformation, the CNN module which comes after the STN is able to recognize the object easily.

Fig. 6 shows the results obtained for object recognition on the GTEA dataset using [5]. From these results, it is obvious that this appraoch was only able to recognize around 2-3 of the 7 objects to a satisfactory accuracy, and the average accuracy for the best of these models is less than 80%. Hence our CNN-STN model outperforms this benchline model by a large margin.

5.3. Results on IEV dataset

Although the model performed well on the GTEA dataset, its performance on the IEV data was miserable. Even after 100 iterations over the entire training set, the model could only recognize the objects with around 20% accuracy. We attribute this failure to the following reasons:

1. While the GTEA is a small dataset (only 3000 samples) with 7 simple classes, IEV consists of 70000 images categorized into 42 types. Our simple CNN-STN with 1 STN module, 1 convolutional layer, and 1 pooling layer may not have sufficient number of parameters to model such a large dataset efficiently.

2. Results may also be affected due to the downsampling of the images owing to resource constraints. We believe that an STN can better transform the image if provided in its original form.

3. Some improvement in performance may also be obtained by tuning the hyperparameters using a method such as grid-search.

6. Conclusion and future work

In this endeavor, we proposed an STN-based CNN model for object recognition in egocentric images. From the results obtained on the GTEA dataset, we validated our claim that an STN can efficiently model the distortions present in an egocentric image because of the natural mobility and low illumination in such images. Although the results with the larger IEV dataset were unsatisfactory, we argued that these could be improved by considering a more complex STN model such as that used for Street View House Number recognition in [11].

From existing appraoches for egocentric object recognition, it may be observed that domain-specific knowledge improves results drastically in most models [22]. For our proposed model, a simple background subtraction module appended before the STN may be hypothesized to improve recognition rate, since the STN would then only have to perform affine transformations without the need to first attend to one of many objects present in the first-person perspective. Such an argument presents the need to integrate domain-specific knowledge and some vision methods (e.g. segmentation) into the CNN-STN model that has been proposed in this paper.

References

[1] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[2] S. Brutzer, B. Höferlin, and G. Heidemann. Evaluation of background subtraction techniques for video surveillance. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1937–1944. IEEE, 2011.

[3] T. S. Cohen and M. Welling. Transformation properties of learned visual representations. *arXiv preprint arXiv:1412.7659*, 2014.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[5] A. Fathi, X. Ren, and J. M. Rehg. Learning to recognize objects in egocentric activities. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference On*, pages 3281–3288. IEEE, 2011.

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[7] P. Geismann and G. Schneider. A two-staged approach to vision-based pedestrian recognition using haar and hog features. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 554–559. IEEE, 2008.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[9] G. E. Hinton, A. Krizhevsky, and S. D. Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer, 2011.

[10] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

[11] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

[12] A. Kanazawa, A. Sharma, and D. Jacobs. Locally scale-invariant convolutional neural networks. *arXiv preprint arXiv:1412.5104*, 2014.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] S.-Y. Kung, S.-H. Lin, L.-J. Lin, and M. Fang. Neural network for locating and recognizing a deformable object, Dec. 15 1998. US Patent 5,850,470.

[15] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[16] D.-S. Lee. Effective gaussian mixture learning for video background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):827–832, 2005.

[17] J. Lei, X. Ren, and D. Fox. Fine-grained kitchen activity recognition using rgb-d. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 208–211. ACM, 2012.

[18] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015.

[19] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[20] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *2011 International Conference on Computer Vision*, pages 89–96. IEEE, 2011.

[21] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[22] X. Ren and C. Gu. Figure-ground segmentation improves handled object recognition in egocentric video. In *CVPR*, volume 2, page 6, 2010.

[23] X. Ren and M. Philipose. Egocentric recognition of handled objects: Benchmark and analysis. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8. IEEE, 2009.

[24] P. Sermanet, A. Frome, and E. Real. Attention for fine-grained categorization. *arXiv preprint arXiv:1412.7054*, 2014.

[25] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[26] K. Sohn and H. Lee. Learning invariant representations with local transformations. *arXiv preprint arXiv:1206.6418*, 2012.

[27] S. K. Sønderby, C. K. Sønderby, L. Maaløe, and O. Winther. Recurrent spatial transformer networks. *arXiv preprint arXiv:1509.05329*, 2015.

[28] T. Tieleman. *Optimizing neural networks that generate images*. PhD thesis, Citeseer, 2014.

[29] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.