Sparse Representations for Object and Ego-motion Estimation in Dynamic Scenes

Hirak J. Kashyap, Charless C. Fowlkes, Jeffrey L. Krichmar, *Senior Member, IEEE*

***Abstract*—Disentangling the sources of visual motion in a dynamic scene during self-movement or ego-motion is important for autonomous navigation and tracking. In the dynamic image segments of a video frame containing independently moving objects, optic flow relative to the next frame is the sum of the motion fields generated due to camera and object motion. The traditional ego-motion estimation methods assume the scene to be static and recent deep learning based methods do not separate pixel velocities into object and ego-motion components. We propose a learning-based approach to predict both ego-motion parameters and object-motion field from image sequences using a convolutional autoencoder, while being robust to variations due to unconstrained scene depth. This is achieved by i) training with continuous ego-motion constraints that allow to solve for ego-motion parameters independently of depth and ii) learning a sparsely activated overcomplete ego-motion field basis set, which eliminates the irrelevant components in both static and dynamic segments for the task of ego-motion estimation. In order to learn the ego-motion field basis set, we propose a new differentiable sparsity penalty function that approximates the number of nonzero activations in the bottleneck layer of the autoencoder and enforces sparsity more effectively than L1 and L2 norm based penalties. Unlike the existing direct ego-motion estimation methods, the predicted global ego-motion field can be used to extract object-motion field directly by comparing against optic flow. Compared to the state-of-the-art baselines, the proposed model performs favorably on pixel-wise object-motion and ego-motion estimation tasks, when evaluated on real and synthetic datasets of dynamic scenes.**

***Index Terms*—Ego-motion, object-motion, sparse representation, convolutional autoencoder, overcomplete basis.**

I. Introduction

OBJECT and ego-motion estimation in videos of dynamic scenes are fundamental to autonomous navigation and tracking, and have found considerable attention in the recent years due to the surge in technological developments for self-driving vehicles [1]–[9]. The task of 6DoF ego-motion prediction is to estimate the six parameters that describe the three-dimensional translation and rotation of the camera between two successive frames. Whereas, object-motion can be estimated either at instance level where each object is assumed rigid [10] or pixel-wise without any rigidity assumption, that is, parts of objects can move differently [6], [11]. Pixel-wise object-motion estimation is more useful since many objects in the real world, such as people, are not rigid [12].

In order to compute object velocity, the camera or observer's ego-motion needs to be compensated [13]. Likewise, the presence of large moving objects can affect the perception of ego-motion [14]. Both ego and object-motion result in the movement of pixels between two successive frames, which is known as optic flow and encapsulates multiple sources of variation. Scene depth, ego-motion, and velocity of independently moving objects determine pixel movements in videos. These motion sources of optic flow are ambiguous, particularly in the monocular case, and so the decomposition is not unique [4].

Several different approaches for ego-motion estimation have been proposed. Feature based methods compute ego-motion based on motion of rigid background features between successive frames [15]–[20]. Another well studied approach is to jointly estimate structure from motion (SfM) by minimizing warping error across the entire image [21]–[24]. While the traditional SfM methods are effective in many cases, they rely on accurate feature correspondences, which are difficult to find in low texture regions, thin or complex structures, and occlusion regions. To overcome some of the issues with SfM approaches, Zhou et al. [2] proposed a deep learning based SfM method using inverse warping loss, which was then further improved in [3], [5], [25]. These deep learning methods rely on finding the rigid background segments for ego-motion estimation [2], [6], [26]. However, these methods do not separate pixel velocities into ego and object-motion components. All of these prior methods that solve for both object and ego-motion use depth as additional input [11], [27], [28]. Joint estimation of object and ego-motion from monocular RGB frames can be ambiguous [4]. However, the estimation of ego and object-motion components from their composite optic flow could be improved by using the geometric constraints of motion-field to regularize a deep neural network based predictor [19], [29].

We introduce a novel approach for predicting 6DoF ego-motion and image velocity generated by moving objects in videos, considering motion-field decomposition in terms of ego and object-motion sources in the dynamic image segments. Our approach first predicts the ego-motion field covering both rigid background and dynamic segments, from which object-motion and 6DoF ego-motion parameters can be derived in closed form. Compared to the existing approaches, our method does not assume a static scene [15], [16], [31] and does not require dynamic segment mask [2], [6], [26] or depth [11], [27], [28] for ego-motion prediction from monocular RGB frames. This is achieved by using continuous ego-motion constraints to train a neural network based predictor, which allow the network to remove variations due to depth and

Manuscript received XX, 2019; revised XX, 2019. Corresponding author: Hirak J. Kashyap (email: kashyaph@uci.edu).

Hirak J. Kashyap and Charless C. Fowlkes are affiliated with Department of Computer Science, University of California, Irvine, CA, 92697.

Jeffrey L. Krichmar is affiliated with Department of Cognitive Sciences and Department of Computer Science, University of California, Irvine, CA, 92697.

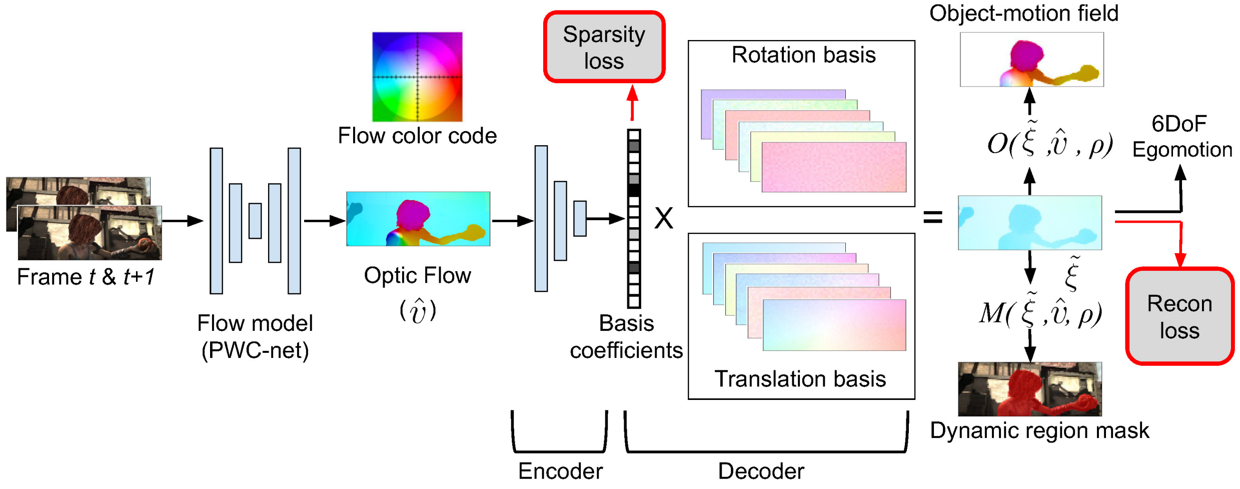


Fig. 1. The proposed sparse autoencoder framework for prediction of object-motion field and ego-motion. Optic flow is obtained using PWC-net [30], which is used to predict unit depth ego-motion field ($\tilde{\xi}$) and subsequently, the 6DoF ego-motion parameters. The output of the encoder forms the basis coefficients, whereas the rotation and translation basis sets are learned during training. Object-motion field and dynamic region masks are calculated using $\tilde{\xi}$, optic flow ($\hat{v}$), and inverse depth (ρ) through operations O and M, respectively. Red arrows denote loss calculation during training.

moving objects in the input frames [19], [29].

Figure 1 depicts the workflow of the proposed solution. To achieve robust ego-motion field prediction in the presence of variations due to depth and moving objects, an overcomplete sparse basis set of rotational and translational ego-motion is learned using a convolutional autoencoder with a nonzero basis activation penalty at the bottleneck layer. The proposed asymmetric autoencoder has a single layer linear decoder that learns the translational and rotational ego-motion basis sets as connection weights, whereas a fully convolutional encoder provides the basis coefficients that are sparsely activated. In order to penalize the number of non-zero neuron activations at the bottleneck layer during training, we propose a continuous and differentiable sparsity penalty term that approximates L0 norm for rectified signal, such as ReLU activation output. Compared to the L1 norm and L2 norm penalties, the proposed sparsity penalty is advantageous since it penalizes similar to the uniform L0 norm operator and does not result in a large number of low magnitude activations.

We propose a new motion-field reconstruction loss comprising continuous ego-motion constraints for end-to-end training of the asymmetric convolutional autoencoder. Compared to the existing baselines methods [2]–[5], [11], [15], [15], [17], [19], [27], [28], SparseMFE achieves state-of-the-art ego-motion and object-motion prediction performances on standard benchmark KITTI and MPI Sintel datasets [1], [32]. Our proposed method for learning a sparse overcomplete basis set from optic flow is effective, as evidenced by an ablation study of the bottleneck layer neurons, which shows that SparseMFE achieves state-of-the-art ego-motion performance on KITTI using only 3% basis coefficients.

In the remainder of the paper, we describe the SparseMFE method in detail, compare our method with existing methods on benchmark datasets, and then discuss the advantages of our proposed method.

II. Background

A. Related work

1) Ego-motion estimation: Ego-motion algorithms are categorized as direct methods [21], [22] and feature-based methods [15], [17]–[20]. Direct methods minimize photometric image reconstruction error by estimating per pixel depth and camera motion, however they are slow and need good initialization. On the other hand, feature based methods use feature correspondences between two images to calculate camera motion. The feature based methods can be divided into two sub-categories, the first category of approaches uses a sparse discrete set of feature points and are called discrete approaches [15], [17], [20]. These methods are fast, but are sensitive to independently moving objects. The second category uses optic flow induced by camera motion between the two frames to predict camera motion, also known as continuous approaches [19], [26], [33], [34]. This approach can take advantage of global flow pattern consistency to eliminate outliers, although it requires correct scene structure estimate [35].

Deep neural networks have been used to formulate direct ego-motion estimation as a prediction problem to achieve state-of-the-art results. Zhou et al. proposed deep neural networks that learned to predict depth and camera motion by training with a self supervised inverse warping loss between the source and the target frames [2]. This self supervised deep learning approach has since been adopted by other methods to further improve ego-motion prediction accuracy [3], [5], [25], [36]. Tung et al. formulated the same problem in an adversarial framework where the generator synthesizes camera motion and scene structure that minimize the warping error to a target frame [37]. These methods do not separate the pixel velocities in the dynamic segments into ego and object-motion components.

2) Object-motion estimation: Compared to monocular ego-motion estimation, fewer methods have been proposed for object-motion estimation from monocular videos. 3D motion field or scene flow was first defined in [38] to describe motion of moving objects in the scene. Many approaches use depth as an additional input. Using RGBD input, scene flow was modelled as piecewise rigid flow superimposed with non-rigid residual from camera motion in [27]. In another RGBD method, dynamic region segmentation was used to solve static regions as visual odometry and the dynamic regions as moving rigid patches [28]. All of these methods assume a rigidity prior and fail with increasingly non-rigid dynamic scenes. To mitigate this, 2D scene flow or pixel-wise object-motion was estimated as non-rigid residual optic flow in the dynamic segments through supervised training of a deep neural network [11].

For RGB input, Vijayanarasimhan et al. proposed neural networks to jointly optimize for depth, ego-motion, and fixed number of objects using inverse warping loss [6]. Due to the inherent ambiguity in the mixture of motion sources in optic flow, an expectation-maximization framework was proposed to train deep neural networks to jointly optimize for depth, ego-motion, and object-motion [4]. These methods were only evaluated qualitatively on datasets with limited object movements.

3) Sparse autoencoder: For high dimensional and noisy data, such as optic flow, a sparse overcomplete representation is an effective method for robust representation of underlying structures [39], [40]. It has been widely used in non-Gaussian noise removal applications from images [41], [42]. A similar representation was proposed to be used in primary visual cortex in the brain to encode variations in natural scenes [43].

Multiple schemes of learning sparse representations have been proposed, such as sparse coding [44], sparse autoencoder [45], sparse winner-take-all circuits [46] and sparse RBMs [47]. Of these, autoencoders are of particular interest since they can be trained comparatively easily via either end-to-end error backpropagation or layerwise training in case of stacked denoising autoencoders [48]. For both types of training, autoencoders learn separable information of the input in deep layers, which are shown to be highly useful for downstream tasks, such as image classification [49], [50] and salient object detection [51].

To learn a representation of underlying motion sources in optic flow, an autoencoder with sparsity regularization is well suited due to its scalability to high dimensional data and feature learning capabilities in presence of noise [46], [52], [53]. In our method, we use a sparse autoencoder to represent ego-motion from noisy optic flow input by removing other components, such as depth and object-motion.

Taken together, the existing monocular ego and object-motion methods, except for [11], cannot estimate both 6DoF ego-motion and unconstrained pixel-wise object-motion in complex dynamic scenes. The method by Lv et al. [11] requires RGBD input for ego-motion prediction and dynamic segment labels for supervision. Therefore, in the following sections we introduce our SparseMFE method that does not require supervision of moving objects for training and estimates ego-motion from RGB input in presence of variations due to depth and independently moving objects.

B. Motion field and flow parsing

Here we analyze the geometry of instantaneous static scene motion under perspective projection. Although these equations were derived previously for ego-motion [19], [26], [29], we illustrate their use in deriving a simplified expression of instantaneous velocities of independently moving objects.

Let us denote the instantaneous camera translation velocity as $t = (t_x, t_y, t_z)^T \in R^3$ and the instantaneous camera rotation velocity as $\omega = (\omega_x, \omega_y, \omega_z)^T \in R^3$. Given scene depth $Z(p_i)$ and its inverse $\rho(p_i) = \frac{1}{Z(p_i)} \in R$ at an image location $p_i = (x_i, y_i)^T \in R^2$ of a calibrated camera image, the image velocity $v(p_i) = (v_i, u_i)^T \in R^2$ due to camera motion is given by,

$$v(p_i) = \rho(p_i)A(p_i)t + B(p_i)\omega \tag{1}$$

where,

$$A(p_i) = \begin{bmatrix} f & 0 & -x_i \\ 0 & f & -y_i \end{bmatrix}$$

$$B(p_i) = \begin{bmatrix} -x_i y_i & f + x_i^2 & -y_i \\ -f - y_i^2 & x_i y_i & x_i \end{bmatrix}$$

If p_i is normalized by the focal length f, then it is possible to replace f with 1 in the expressions for $A(p_i)$ and $B(p_i)$.

If the image size is N pixels, then the full expression of instantaneous velocity at all the points due to camera motion, referred to as ego-motion field (EMF), can be expressed in a compressed form as,

$$v = \rho A t + B\omega \tag{2}$$

where, A, B, and ρ entails the expressions $A(p_i)$, $B(p_i)$, and $\rho(p_i)$ respectively for all the N points in the image as follows.

$$v = \begin{bmatrix} v(p_1) \\ v(p_2) \\ \vdots \\ v(p_N) \end{bmatrix} \in R^{2N \times 1}, \quad \rho A t = \begin{bmatrix} \rho_1 A(p_1) t \\ \rho_2 A(p_2) t \\ \vdots \\ \rho_N A(p_N) t \end{bmatrix} \in R^{2N \times 1}$$

$$B\omega = \begin{bmatrix} B(p_1)\omega \\ B(p_2)\omega \\ \vdots \\ B(p_N)\omega \end{bmatrix} \in R^{2N \times 1}$$

Note that the rotational component of EMF is independent of depth.

The monocular continuous ego-motion computation uses this formulation to estimate the unknown parameters t and ω given the point velocities v generated by camera motion [19], [29]. However, instantaneous image velocities obtained from the standard optic flow methods on real data are usually different from the EMF [26]. The presence of moving objects further deviates the optic flow away from the EMF. Let us call the input optic flow as $\hat{v}$, which is different from v.

Therefore, monocular continuous methods on real data solve the following minimization objective to find t, ω, and ρ.

$$t^*, \omega^*, \rho^* = \underset{t,\omega,\rho}{\operatorname{argmin}} \|\rho A t + B\omega - \hat{v}\|^2 \tag{3}$$

Following [19], [54], without loss of generality, the objective function can be first minimized for ρ as,

$$t^*, \omega^*, \rho^* = \underset{t,\omega}{\operatorname{argmin}}\, \underset{\rho}{\operatorname{argmin}} \|\rho A t + B\omega - \hat{v}\|^2 \tag{4}$$

Therefore, the minimization for t^* and ω^* can be performed as,

$$t^*, \omega^* = \underset{t,\omega}{\operatorname{argmin}} \left\| A^{\perp} t^T (B\omega - \hat{v}) \right\|^2 \tag{5}$$

where $A^{\perp}t$ is orthogonal complement of At. This resulting expression does not depend on ρ and can be optimized directly to find optimal t^* and ω^*.

In dynamic scenes, the independently moving objects generate additional image velocities. Therefore, the resulting optic flow can be expressed as the sum of the flow components due to ego-motion ($\hat{v}_e$) and object-motion ($\hat{v}_o$). Following this, Equation 5 can be generalized as,

$$t^*, \omega^* = \underset{t,\omega}{\operatorname{argmin}} \left\| A^{\perp} t^T (B\omega - \hat{v}_e - \hat{v}_o) \right\|^2 \tag{6}$$

Since $\hat{v}_o$ is independent of t and ω, it can considered as non-gaussian additive noise and Equation 6 provides a robust formulation of Equation 5. After solving for t^* and ω^*, image velocity due to object-motion across the entire image can be recovered as,

$$\tilde{v}_o = \hat{v} - \rho A t^* + B\omega^* \tag{7}$$

We will refer to $\tilde{v}_o$ as the predicted object-motion field (OMF). Equation 7 is equivalent to flow parsing, which is a mechanism proposed to be used by the human visual cortex to extract object velocity during self movement [55].

Note that the expression is dependent on ρ. Although human observers are able to extract depth in the dynamic segments using stereo input and prior information about objects, the structure-from-motion methods cannot reliably estimate depth in the dynamic segments without prior information about objects [3], [5], [25], [36]. Since, the separation into EMF and OMF in the dynamic segments cannot be automated without prior information about objects, the datasets of generic real world scenes do not provide ground truth OMF [1].

III. Representation of Ego-Motion Using a Sparse Basis Set

We propose to represent ego-motion as depth normalized translation EMF and rotational EMF, which can be converted to 6DoF ego-motion parameters in closed form. In this setup, the minimization in Equation 6 can be converted to an equivalent regression problem for depth normalized translational EMF and rotational EMF, denoted as ξ_t and ξ_ω respectively. We hypothesize that regression with the EMF constraints from Equation 1 will be more robust than direct 6DoF ego-motion prediction methods in presence of variations due to depth and dynamic segments [2], [3], [5].

Regression of high dimensional output is a difficult problem. However, significant progress has been made using deep neural networks and generative models [4], [6], [37], [56]. For structured data such as EMF, the complexity of regression can be greatly reduced by expressing the target as a weighted linear combination of basis vectors drawn from a pre-computed dictionary. Then the regression will be a much simpler task of estimating the basis coefficients, which usually has orders of magnitude lower dimension than the target.

Suppose $\tilde{\xi}_t$ is the prediction for depth normalized translational EMF obtained as linear combination of basis vectors from a dictionary T. And $\tilde{\xi}_\omega$ is the prediction for rotational EMF calculated similarly from a dictionary R.

$$\tilde{\xi}_t = \sum_{j=1}^{m} \alpha_j T_j \tag{8}$$

$$\tilde{\xi}_\omega = \sum_{j=1}^{n} \beta_j R_j \tag{9}$$

where α_j and β_j are the coefficients and $m, n << N$. Small values of m, n not only lead to computational efficiency, but they also allow each basis vector to be meaningful and generic.

On the other hand, having too few active basis vectors is counterproductive for predictions on unseen data with non-Gaussian variations. For example, PCA finds a small set of uncorrelated basis vectors, however, it requires that the important components of the data have the largest variance. Therefore, in presence of non-Gaussian noise with high variance, the principal components deviate from the target distribution and generalize poorly to unseen data [57]. Furthermore, a smaller dictionary is more sensitive to corruption of the coefficients due to noisy input.

Therefore, for high dimensional and noisy data, a redundant decomposition of the Equations 8, 9 is preferred. Dictionaries with linearly dependent bases are called overcomplete and they have been used widely for noise removal applications [39]–[41] and in signal processing [42], [58]. Overcomplete representations are preferred due to flexibility of representation for high dimensional input, robustness, and sparse activation [39].

Despite the flexibility provided by overcompleteness, there is no guarantee that a large set of manually picked linearly dependent basis vectors will fit to the structure of the underlying input distribution [39]. Therefore, an overcomplete dictionary must be learned from the data such that the basis vectors encode maximum structure in the distribution. However, the under-determined problem of finding a large overcomplete dictionary becomes unstable when the input data are inaccurate or noisy [59]. Nevertheless, the ill-posedness can be greatly diminished using a sparsity prior on the activations of the basis vectors [41]–[43]. Considering sparse activation prior, the decomposition in Equation 8 is constrained by,

$$\|\alpha\|_0 < k \tag{10}$$

$\|\alpha\|_0$ is the L0 (pseudo)norm of α and denotes the number of non-zero basis coefficients, with an upper bound k. The

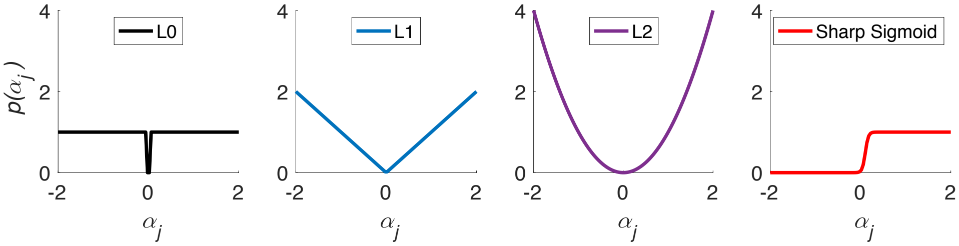


Fig. 2. L0, L1, and L2 norm penalties and the proposed sharp sigmoid penalty for basis coefficient α_j. It can be observed that for $\alpha_j \geq 0$, the sharp sigmoid penalty approximates the L0 penalty and is continuous and differentiable. The sharp sigmoid function shown above corresponds to $Q = 25$ and $B = 30$. The L1 and L2 norm penalties enforce shrinkage on larger values of α_j. Moreover, for a set of coefficients, L1 and L2 norm penalties cannot indicate the number of $\alpha_j > 0$ due to not having any upper bound.

decomposition for $\tilde{\xi}_\omega$ in Equation 9 is similarly obtained and will not be stated for brevity.

Therefore, the objective function to solve for basis T and co-efficients α can be written as,

$$\underset{T,\alpha}{\operatorname{argmin}} \|\xi_t - \sum_{j=1}^{m} \alpha_j T_j\|_1 \quad \text{subject to} \quad \|\alpha\|_0 < k \tag{11}$$

We use L1 norm for the reconstruction error term since it is robust to input noise [60]. In contrast, the more commonly used L2 norm overfits to noise, since it results in large errors for outliers [61]. As the ξ_t components can be noisy, L1 norm of reconstruction error is more suitable in our case.

The regularizer in Equation 11, known as best variable selector [62], requires a pre-determined upper bound k, which may not be the optimal for all samples in a dataset. Therefore, a penalized least squares form is preferred for optimization.

$$\underset{T,\alpha}{\operatorname{argmin}} \|\xi_t - \sum_{j=1}^{m} \alpha_j T_j\|_1 + \lambda_s \|\alpha\|_0 \tag{12}$$

The penalty term in Equation 12 is computed as $\|\alpha\|_0 = \sum_{j=1}^{m} 1(\alpha_j \neq 0)$, where $1(.)$ is the indicator function. However, the penalty term results in 2^m possible states of the coefficients α and the exponential complexity is not practical for large values of m, as in the case of overcomplete basis [63]. Further, the penalty function is not differentiable and cannot be solved using gradient based methods.

Although functionally different, the penalty function in Equation 12 is commonly approximated using a L1 norm penalty, which is differentiable and results in a computationally tractable convex optimization problem.

$$\underset{T,\alpha}{\operatorname{argmin}} \|\xi_t - \sum_{j=1}^{m} \alpha_j T_j\|_1 + \lambda_s \|\alpha\|_1 \tag{13}$$

Penalized regression of the form in Equation 13 is known as Lasso [64], where the penalty $\|\alpha\|_1 = \sum_{j=1}^{m} |\alpha_j|_1$ shrinks the coefficients toward zero and can ideally produce a sparse solution. However, Lasso operates as a biased shrinkage operator as it penalizes larger coefficients more compared to smaller coefficients [63], [65]. As a result, it prefers solutions with many small coefficients than solutions with fewer large coefficients. When input has noise and correlated variables, Lasso results in a large set of activations, all shrunk toward zero, to minimize the reconstruction error [63].

To perform best variable selection through a gradient based optimization, we propose to use a penalty function that approximates L0 norm for rectified input based on the generalized logistic function with a high growth rate, which we call as sharp sigmoid penalty and is defined for the basis coefficient α_j as,

$$p(\alpha_j) = \frac{1}{1 + Qe^{-B\alpha_j}} \tag{14}$$

where, Q determines the response at $\alpha = 0$ and B determines the growth rate. The Q and B hyperparameters are tuned within a finite range such that i) zero activations are penalized with either zero or a negligible penalty and ii) small magnitude activations are penalized equally as the large magnitude activations (like L0). The sharp sigmoid penalty is continuous and differentiable for all input values, making it a well suited sparsity regularizer for gradient based optimization methods. Thus, the objective function with sharp sigmoid sparsity penalty can be written as,

$$\underset{T,\alpha}{\operatorname{argmin}} \|\xi_t - \sum_{j=1}^{m} \alpha_j T_j\|_1 + \lambda_s \sum_{j=1}^{m} \frac{1}{1 + Qe^{-B\alpha_j}} \tag{15}$$

Figure 2 shows that the sharp-sigmoid penalty approximates number of non-zero coefficients in rectified α. It provides a sharper transition between 0 and 1 compared to the sigmoid function and does not require additional shifting and scaling. To achieve dropout like weight regularization [66], a sigmoid derived hard concrete gate was proposed in [65] to penalize neural network connection weights. However, it does not approximate the number of non-zero weights and averages to the sigmoid function for noisy input.

IV. Joint Optimization for Basis Vectors and Coefficients

We now describe the proposed optimization method to find the basis sets T, R and coefficients α for translational and rotational EMF, based on the objective function in Equation 15. We let the optimization determine the coupling between the coefficients for rotation and translation, therefore the coefficients α are shared between T and R. We write the objective in a framework of energy function $E(\xi_t, \xi_\omega | T, R, \alpha)$ as

$$T^*, R^*, \alpha^* = \underset{T,R,\alpha}{\operatorname{argmin}} E(\xi_t, \xi_\omega | T, R, \alpha) \tag{16}$$

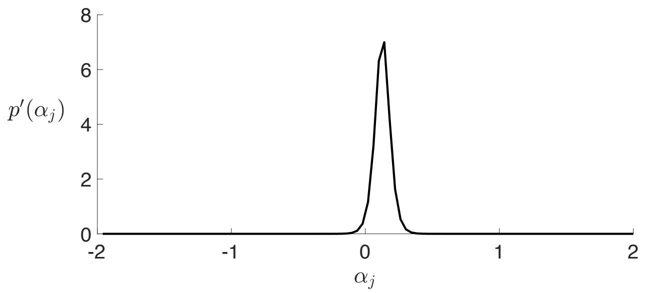


Fig. 3. Derivative of the sharp sigmoid penalty function $p(\alpha_j)$ with respect to coefficient α_j.

where

$$E(\xi_t, \xi_\omega|T, R, \alpha) = \lambda_t\|\xi_t - \sum_{j=1}^{m} \alpha_j T_j\|_1 + \lambda_\omega\|\xi_\omega - \sum_{j=1}^{m} \alpha_j R_j\|_1 + \lambda_s \sum_{j=1}^{m} \frac{1}{1 + Qe^{-B\alpha_j}} \qquad (17)$$

There are three unknown variables T, R, and α to optimize such that the energy in Equation 17 is minimal. This can be performed by optimizing over each variable one by one [43]. For example, expectation maximization procedure can be used to iteratively optimize over each unknown.

For gradient based minimization over α_j, we may iterate until the derivative of $E(\xi_t, \xi_\omega|T, R, \alpha)$ with respect to each α_j is zero. For each input optic flow, the α_j are solved by finding the equilibrium of the differential equation

$$\dot{\alpha_j} = \lambda_t T_j sgn(\xi_t - \sum_{j=1}^{m} \alpha_j T_j) + \lambda_\omega R_j sgn(\xi_\omega - \sum_{j=1}^{m} \alpha_j R_j) - \lambda_s p'(\alpha_j) \qquad (18)$$

However, the third term of this differential that imposes self-inhibition on α_j is problematic. As depicted in Figure 3, the gradient $p'(\alpha_j)$ of the sharp sigmoid penalty with respect to the coefficient is mostly zero, except for a small interval of coefficient values close to zero. As a result, the α_j values outside this interval will have no effect on the minimization to impose sparsity. The sparsity term also has zero derivatives with respect to R and T, therefore Equation 16 cannot be directly optimized over T, R, and α for sparsity when sharp sigmoid penalty is used.

Instead, we can cast it as a parameterized framework where the optimization is solved over a set of parameters θ_s that predicts the sparse coefficients α to minimize the energy form in Equation 17. This predictive model can be written as $\alpha = f_{\theta_s}(\hat{v})$. The unknown variables R and T can be grouped along with θ_s as $\theta = \{T, R, \theta_s\}$ and optimized jointly to solve the objective

$$\theta^* = \operatorname*{argmin}_{\theta} E(\xi_t, \xi_\omega, \alpha|\theta) \qquad (19)$$

where $E(\xi_t, \xi_\omega, \alpha|\theta)$ is equivalent to the energy function in Equation 17, albeit expressed in terms of variable θ.

The objective in Equation 19 can be optimized efficiently using an autoencoder neural network with θ_s as its encoder parameters and $\{T, R\}$ as its decoder parameters. The encoder output or bottleneck layer activations provide the basis coefficients α. Following this approach, we propose Sparse Motion Field Encoder (SparseMFE), which learns to predict EMF due to self rotation and translation from optic flow input. The predicted EMF allows direct estimation of 6DoF ego-motion parameters in closed form and prediction of projected object velocities or OMF via flow parsing [55].

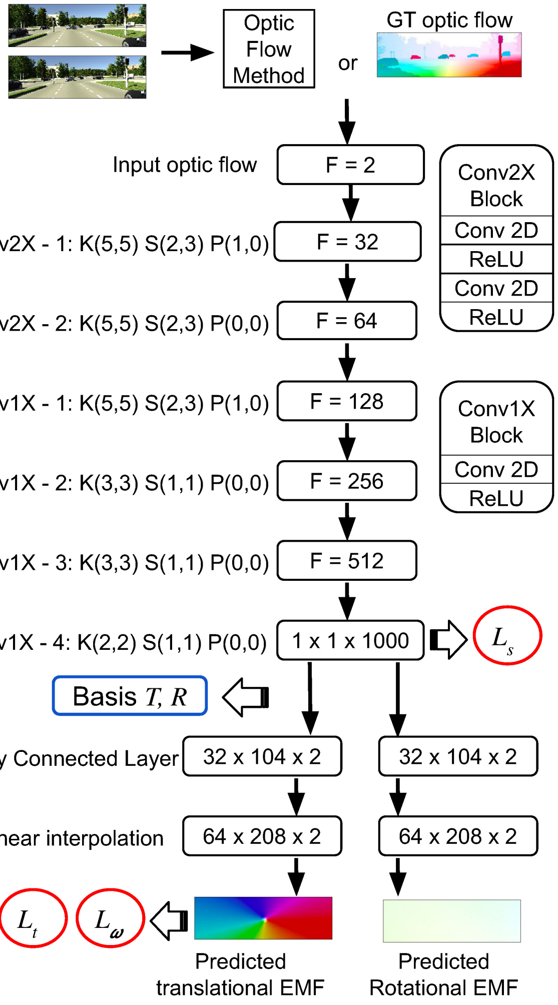


Fig. 4. Architecture of the proposed SparseMFE network. Conv blocks are fully convolutional layers of 2D convolution and ReLU operations. The receptive field size is gradually increased such that each neuron in the Conv1X-4 layer operates across the entire image. Outputs of all Conv blocks are non-negative due to ReLU operations. K, S, and P denote the kernel sizes, strides, and padding along vertical and horizontal directions of feature maps. F denotes the number of filters in each layer. The weights of the fully connected layer forms the basis for translational and rotational egomotion.

Figure 4 depicts the architecture of the proposed SparseMFE network. The network is an asymmetric autoencoder that has a multi-layer fully convolutional encoder and a single layer linear decoder. We will refer to the Conv1X-4 block at the end of the encoder consisting of $m = 1000$ neurons as the bottleneck layer of the SparseMFE network. The bottleneck layer predicts a latent space embedding of ego-motion from input optic flow. This embedding operates as coefficients α for the basis vectors of dictionaries T and R learned as the fully connected decoder weights. The outputs of all Conv block in the encoder, including the bottleneck layer neurons, are non-negative due to ReLU operations.

EMF reconstruction losses

The translational and rotational EMF reconstruction losses by SparseMFE are obtained as,

$$L_t = \|\xi_t - \tilde{\xi}_t\|_1 \tag{20}$$

$$L_\omega = \|\xi_\omega - \tilde{\xi}_\omega\|_1 \tag{21}$$

where, ξ_t is true translational EMF with $\rho = 1$ and ξ_ω is true rotational MF, obtained using Equation 2.

As most datasets contain disproportionate amount of rotation and translation, we propose to scale L_t and L_ω relative to each other, such that the optimization is unbiased. The scaling coefficients of L_t and L_ω for each input batch are calculated as,

$$\lambda_t = max(\frac{\|\xi_\omega\|_2}{\|\xi_t\|_2}, 1) \tag{22}$$

$$\lambda_\omega = max(\frac{\|\xi_t\|_2}{\|\xi_\omega\|_2}, 1) \tag{23}$$

Sparsity loss

The SparseMFE network is regularized during training for sparsity of activation of the bottleneck layer neurons. This is implemented by calculating a sparsity loss (L_s) for each batch of data and backpropagating it along with the EMF reconstruction loss during training. The value of L_s is calculated for each batch of data as the number of non-zero activations of the bottleneck layer neurons, also known as population sparsity. Although, to make this loss differentiable, we approximate a number of activations using sharp sigmoid penalty in Equation 14. The penalty L_s is calculated as,

$$L_s = \sum_{j=1}^{m} p(\alpha_j) \tag{24}$$

Combining EMF reconstruction loss and sparsity loss, the total loss for training is given by,

$$L = \lambda_t L_t + \lambda_\omega L_\omega + \lambda_s L_s \tag{25}$$

where, λ_s is a hyperparameter to scale sparsity loss.

V. Experimental results

We evaluate the performance of SparseMFE in ego-motion and object velocity prediction tasks, comapring to the baselines on real KITTI odometry dataset and synthetic MPI Sintel dataset [1], [32]. Additionally, we analyze the EMF basis set learned by SparseMFE for sparsity and overcompleteness.

The predictions for 6DoF translation and rotation parameters are computed in closed form from $\tilde{\xi}_t$ and $\tilde{\xi}_\omega$, respectively, following the continuous ego-motion formulation.

$$\tilde{t} = \tilde{\xi}_t / A \mid \rho = 1, \quad \tilde{\omega} = \tilde{\xi}_\omega / B \tag{26}$$

Projected object velocities or OMF are obtained using Equation 7.

A. Datasets

1) KITTI visual odometry dataset: We use the KITTI visual odometry dataset [1] to evaluate ego-motion prediction performance by the proposed model. This dataset provides eleven driving sequences (00-10) with RGB frames (we use only the left camera frames) and the ground truth pose for each frame. Of these eleven sequences, we use sequences 00-08 for training our model and sequences 09, 10 for testing, similar to [2], [3], [5], [26]. This amounts to approximately 20.4K frames in the training set and 2792 frames in the test set. As ground truth optic flow is not available for this dataset, we use a pretrained PWC-Net [30] model to generate optic flow from the pairs of consecutive RGB frames for both training and testing.

2) MPI Sintel dataset: MPI Sintel dataset contains scenes with fast camera and object movement and also many scenes with large dynamic regions [32]. Therefore, this is a challenging dataset for ego-motion and OMF prediction. Similar to the other pixel-wise object-motion estimation methods [11], we split the dataset such that the test set contains scenes with a different proportion of dynamic regions, in order to study the effect of moving objects on prediction accuracy. Of the 23 scenes in the dataset, we select $alley\_2 (1.8\%)$, $temple\_2 (5.8\%)$, $market\_5 (27.04\%)$, $ambush\_6 (38.96\%)$, and $cave\_4 (47.10\%)$ sequences as the test set, where the number inside the parentheses specify the percentage of dynamic regions in each sequence [11]. The rest 18 sequences are used to train SparseMFE.

B. Training

We use Adam optimizer [56] to train SparseMFE. Learning rate η is set to 10^{-4} and is chosen empirically by line search. The β_1 and β_2 parameters of Adam are set to 0.99 and 0.999, respectively. The sparsity coefficient λ_s for training is set to 10^2, whose selection criterion is described later in Section V-E.

C. Ego-motion prediction

For the KITTI visual odometry dataset [1], following the existing literature on learning based ego-motion prediction [2]–[5], [26], absolute trajectory error (ATE) metric is used for ego-motion evaluation, which measures the distance between the corresponding points of the ground truth and the predicted trajectories. In Table I, we compare the proposed model against the existing methods on the KITTI odometry dataset. Recent deep learning based SfM models for direct 6DoF ego-motion prediction are compared as baselines since their ego-motion prediction method is comparable to SparseMFE. For reference, we also compare against a state-of-the-art visual SLAM method, ORB-SLAM [17] and epipolar geometry based robust optimization methods [15], [19].

Table I shows that SparseMFE achieves the state-of-the-art ego-motion prediction accuracy on both test sequences 09 and 10 of the KITTI odometry test split compared to the state-of-the-art learning based ego-motion methods [3]–[5] and geometric ego-motion estimation baselines [15], [17], [19].

In order to investigate the effectiveness of the learned sparse representation of ego-motion, we evaluate ATE using

TABLE I
ABSOLUTE TRAJECTORY ERROR (ATE) ON THE KITTI VISUAL ODOMETRY TEST SET. THE LOWEST ATE IS DENOTED IN BOLDFACE.

Method	Seq 09	Seq 10
ORB-SLAM [17]	0.064±0.141	0.064±0.130
Robust ERL [19]	0.447±0.131	0.309±0.152
8-pt Epipolar + RANSAC [15]	0.013±0.016	0.011±0.009
Zhou et al. [2]	0.021±0.017	0.020±0.015
Lee and Fowlkes [26]	0.019±0.014	0.018±0.013
Yin et al. [3]	0.012±0.007	0.012±0.009
Mahjourian et al. [5]	0.013±0.010	0.012±0.011
Godard et al. [25]	0.023±0.013	0.018±0.014
Ranjan et al. [4]	0.012±0.007	0.012±0.008
SparseMFE	**0.011±0.007**	**0.011±0.007**
SparseMFE (top 5% coefficients)	**0.011±0.007**	**0.011±0.007**
SparseMFE (top 3% coefficients)	**0.011±0.007**	**0.011±0.007**
SparseMFE (top 1% coefficients)	0.011±0.008	0.012±0.008

only a few top percentile activations of basis coefficients in the bottleneck layer of SparseMFE. This metric tells about dimensionality reduction capabilities of an encoding scheme. As shown in Table I, SparseMFE achieves state-of-the-art ego-motion prediction on both sequences 09 and 10 using only the 3% most active basis coefficients for each input frame pair. Further, when using this subset of coefficients only, the achieved ATE is equal to when using all the basis coefficients. This implies that SparseMFE is able to learn a sparse representation of ego-motion.

On the MPI Sintel dataset, we use the relative pose error (RPE) [67] metric for evaluation of ego-motion prediction, similar to the baseline method Rigidity Transform Network (RTN) [11]. SparseMFE is comparable to this parametric method without any additional iterative refinement of ego-motion. An offline refinement step can be used with SparseMFE as well. However, offline iterative refinement methods are independent of the pose prediction and therefore, cannot be compared directly.

Table II compares ego-motion prediction performance of SparseMFE against the baseline RTN [11], ORB-SLAM [17], geometric ego-motion methods [15], [19], and non-parametric baselines SRSF [27] and VOSF [28] on the Sintel test split. SparseMSE and the geometric baselines do not use depth input for ego-motion prediction, however, RTN, SRSF, and VOSF use RGBD inputs. For a fair comparison with RTN, both methods obtain optic flow using PWC-net [30]. SparseMFE achieves the lowest overall rotation prediction error compared to the existing methods, even when using only RGB frames as input. Although, VOSF [28] achieves the lowest overall translation prediction error, it uses depth as an additional input to predict ego-motion.

D. Object-motion prediction

We quantitatively and qualitatively evaluate SparseMFE on object-motion prediction using the Sintel test split. We compare to RTN [11] and Semantic Rigidity [68] as the state-of-the-art learning based baselines and SRSF [27] and VOSF [28] as non-parametric baselines for obejct-motion evaluation. RTN [11] trained using the Things3D dataset [69] for generalization is also included. The standard end-point-error (EPE) metric is used, which measures the euclidean distance between the ground truth and the predicted 2D flow vectors generated by moving objects. These 2D object flow vectors are herein referred to as OMF and with a different terminology "projected scene flow" in [11]. Table III shows that SparseMFE achieves the state-of-the-art OMF prediction accuracy on four out of five test sequences. The other methods become progressively inaccurate with larger dynamic regions. On the other hand, SparseMFE maintains OMF prediction accuracy even when more than 40% of the scene is occupied by moving objects, as in case of the *cave_4* sequence.

Figure 5 depicts qualitative OMF performance of SparseMFE on each of the five sequences from the Sintel test split. Dynamic region mask is obtained by thresholding the residual optic flow from Equation 7. While SparseMFE successfully recovers OMF for fast moving objects, it is possible that some rigid background pixels with faster flow components are classified as dynamic regions, as for the examples from *market_5* and *cave_4* sequences. This can be avoided by using more data for training, since these background residual flows are generalization errors stemming from ego-motion prediction and are absent in training set predictions.

We show qualitative object-motion prediction results on real world KITTI benchmark [70] in Figure 6, which illustrates effective dynamic region prediction compared to ground truth dynamic region masks. The benchmark does not provide ground truth OMF, which are difficult to obtain for real world scenes.

E. Sparsity analysis

We analyze the effect of using the sparsity regularizer in the encoding of ego-motion. The proposed sharp sigmoid penalty in Equation 14 is compared against L1 and L2 norm sparsity penalties commonly used in sparse feature learning methods [71], [72]. ReLU non-linearity at the bottleneck layer was proposed for sparse activations [53]. Since the bottleneck layer of Sparse MFE uses ReLU non-linearity, we also compare the case where no sparsity penalty is applied.

Figure 7 depicts the effectiveness of the proposed sharp sigmoid penalty in learning a sparsely activated basis set for ego-motion prediction. Figure 7(a) shows number of nonzero activations in the bottleneck layer on Sintel test split when the network is trained using different sparsity penalties. Sharp sigmoid penalty results in sparse and stable activations of basis coefficients for all Sintel test sequences. On the contrary, L0 and L1 norm penalties find dense solutions where large basis subsets are used for all sequences. Figure 7(b) shows the activation heatmap of the bottleneck layer for the *market_5* frame in Figure 5 for the tested sparsity penalties. L0 and L1 penalties do not translate to the number of nonzero activations, rather work as a shrinkage operator on activation magnitude, to result in large number of small activations in the bottleneck layer. On the other hand, the proposed sharp sigmoid penalty activates only a few neurons in that layer.

We conducted ablation experiments to study the effectiveness of L1, L2, and sharp sigmoid penalties in learning a

TABLE II
RELATIVE POSE ERROR (RPE) COMPARISON ON THE SINTEL TEST SET. THE LOWEST AND THE SECOND LOWEST RPE ON EACH SEQUENCE ARE DENOTED USING BOLDFACE AND UNDERLINE, RESPECTIVELY. ★ DENOTES THAT A METHOD USES RGBD INPUT FOR EGO-MOTION PREDICTION.

	dynamic region <10%				dynamic region 10% - 40%				dyn. reg. >40%		All	
	alley_2		temple_2		market_5		ambush_6		cave_4		Average	
	RPE(t)	RPE(r)	RPE(t)	RPE(r)	RPE(t)	RPE(r)	RPE(t)	RPE(r)	RPE(t)	RPE(r)	RPE(t)	RPE(r)
ORB-SLAM [17]	0.030	0.019	0.174	0.022	0.150	0.016	0.055	0.028	**0.017**	0.028	0.089	0.022
Robust ERL [19]	**0.014**	0.022	0.354	0.019	0.259	0.035	0.119	0.107	0.018	0.046	0.157	0.041
8-pt + RANSAC [15]	0.058	**0.002**	0.216	**0.006**	0.087	0.012	0.096	0.041	0.018	0.019	0.095	0.013
SRSF [27]★	0.049	0.014	0.177	0.012	0.157	0.011	0.067	0.073	0.022	0.015	0.098	0.018
VOSF [28]★	0.104	0.032	**0.101**	0.016	**0.061**	**0.001**	**0.038**	**0.019**	0.044	**0.005**	**0.075**	0.014
RTN [11]★	0.035	0.028	0.159	0.012	0.152	0.021	0.046	0.049	0.023	0.021	0.088	0.022
SparseMFE	0.020	0.005	0.172	0.010	0.202	0.011	0.087	0.041	0.025	0.011	0.103	**0.012**

TABLE III
END POINT ERROR (EPE) COMPARISON OF OMF PREDICTION ON THE SINTEL TEST SPLIT. THE LOWEST EPE PER SEQUENCE IS DENOTED IN BOLDFACE.

	dynamic region <10%		dynamic region 10% - 40%		dynamic region >40%	All
	alley_2	temple_2	market_5	ambush_6	cave_4	Average
SRSF [27]	7.78	15.51	31.29	39.08	13.29	18.86
VOSF [28]	1.54	8.91	35.17	24.02	9.28	14.61
Semantic Rigidity [68]	0.48	5.19	13.02	19.11	6.50	7.39
RTN (trained on Things3D [69]) [11]	0.52	9.82	16.99	52.21	5.07	11.88
RTN [11]	0.48	**3.27**	11.35	19.08	4.75	6.12
SparseMFE	**0.29**	4.59	**11.27**	**4.82**	**0.93**	**4.32**

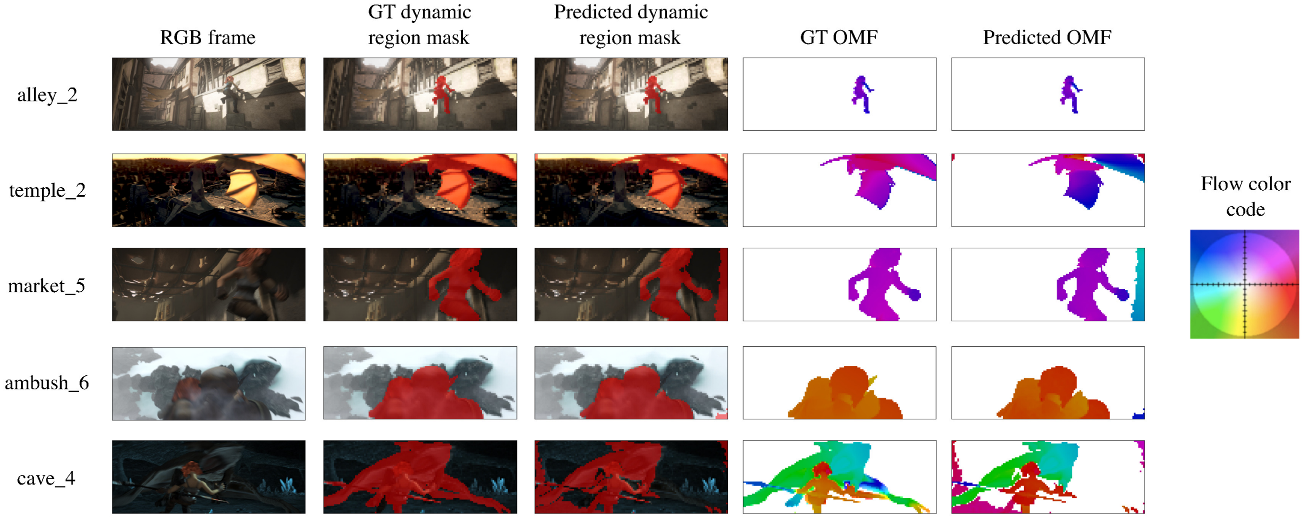


Fig. 5. Qualitative results of SparseMFE on Sintel test split. The red colored overlay denotes the dynamic region masks.

sparse representation of ego-motion. Figure 8 depicts qualitative OMF and dynamic mask prediction performance on the *alley*_2 test frame from Figure 5 by SparseMFE instances trained using either L1, L2, or sharp sigmoid penalties, with or without ablation. During ablation, we use only a fraction of the top bottleneck neuron activations (coefficients) and set the others to zero. The results show that sharp sigmoid penalty based training provides stable OMF and dynamic mask prediction using only top 1% activations, whereas L2 sparsity penalty based training results in loss of accuracy as neurons are removed from bottleneck layer. L1 penalty based training results in erroneous OMF and mask predictions for this example. Another ablation study depicted in Figure 9 shows that SparseMFE trained using sharp sigmoid sparsity penalty is more robust to random removal of neurons from the bottleneck layer compared to when trained using L1 and L2 norm sparsity penalties.

To study the effect of the sparsity loss coefficient λ_s on ego-motion prediction, we conducted a study by varying λ_s during training and using only a fraction of the most activated bottleneck layer neurons for ego-motion prediction during test and setting the rest to zero. Figure 10 depicts the effect of ablation on the ego-motion prediction accuracy during test, for λ_s values in the set $\{10^e | 0 \leq e < 4, e \in \mathbb{Z}\}$. As can be seen, $\lambda_s = 10^2$ achieves the smallest and stable ATE for different amount of ablation. For smaller λ_s values,

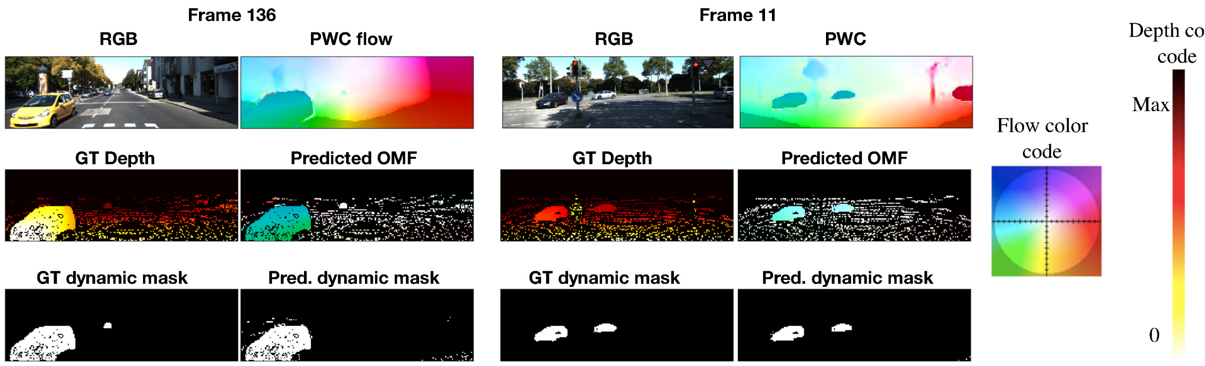
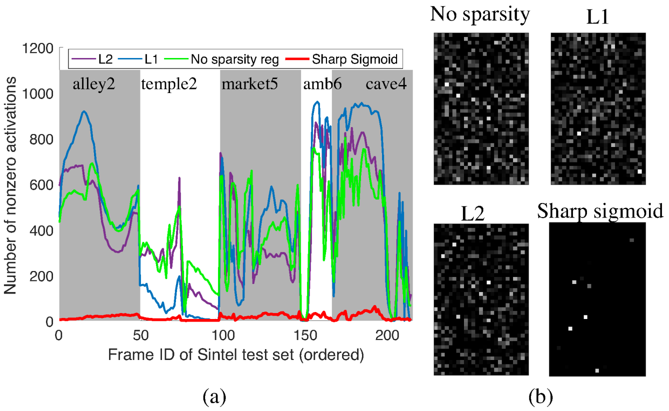


Fig. 6. Qualitative results of SparseMFE on KITTI benchmark real world frames. Ground truth OMF is not available, however, ground truth dynamic region masks are provided in the benchmark. The ground truth depth map is sparse, and the pixels where depth is not available are colored in black.



(a) (b)

Fig. 7. Neuron activation profile in the bottleneck layer on Sintel test split for different types of sparsity regularization. (a) Number of nonzero activations in the bottleneck layer for frame sequences in the Sintel test split. Line colors denote the sparsity regularization used. (b) Activation heatmap of the bottleneck for the *market_5* frame shown in Figure 5. All experiments are conducted after the network has converged to a stable solution.

the prediction becomes inaccurate as more bottleneck layer neurons are removed. Although $\lambda_s = 10^3$ provides stable prediction, it is less accurate than $\lambda_s = 10^2$. The stability to ablation of neurons for larger λ_s values is a further indication of the effectiveness of the sharp sigmoid sparsity penalty in learning a sparse basis set of ego-motion.

F. Ablation of other loss terms

TABLE IV
ATE ON THE KITTI VISUAL ODOMETRY TEST SET

Method	Seq 09	Seq 10
SparseMFE (w/o translation loss)	0.776±0.192	0.554±0.242
SparseMFE (w/o rotation loss)	0.019±0.013	0.017±0.014
SparseMFE (Full)	**0.011±0.007**	**0.011±0.007**

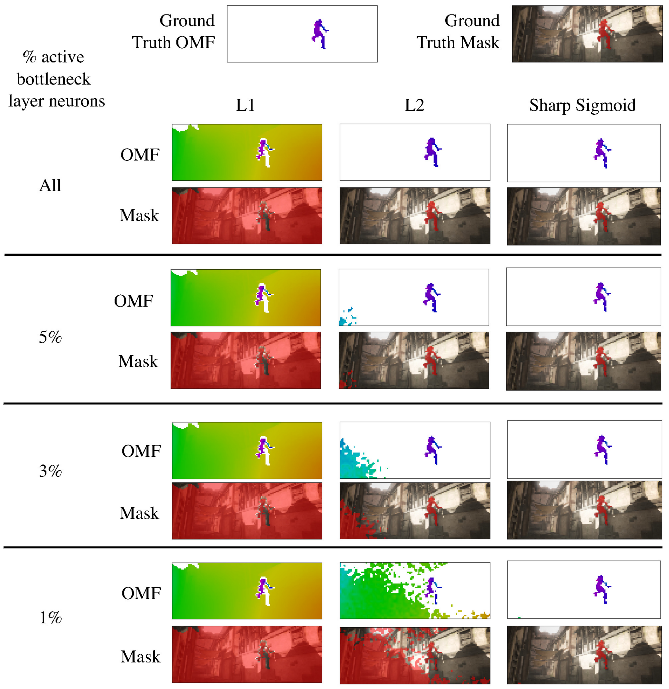


Fig. 8. Qualitative OMF and dynamic mask prediction results comparing L1, L2, and Sharp Sigmoid sparsity penalties, in terms of their robustness to removal of bottleneck layer neurons during testing.

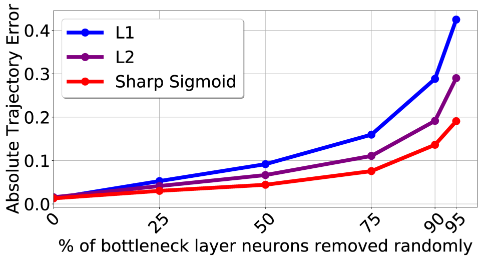


Fig. 9. Ablation study comparing L1, L2, and sharp sigmoid sparsity penalties for ego-motion inference on KITTI test sequence 10.

Similar to the ablation study for sparsity, we ablated the translational ($\lambda_t = 0$) and rotational ($\lambda_w = 0$) EMF reconstruction loss terms of the objective function in Equation 25 to evaluate the contribution of these terms to the overall performance of the proposed method. As shown in Table IV, removal of the translational loss term or the rotational loss term during training reduces the test accuracy of ego-motion prediction. Moreover, the translational loss term contributes

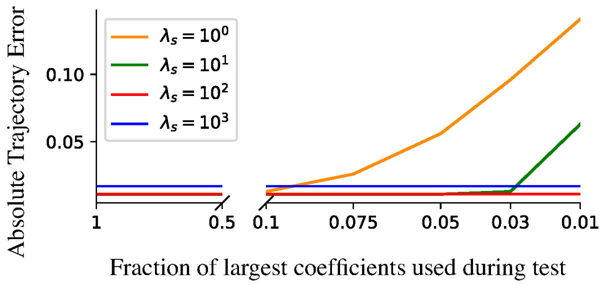


Fig. 10. Ablation experiment to study the effect of the sparsity loss coefficient λ_s on ego-motion prediction. During test, only a fraction of the bottleneck layer neurons are used for ego-motion prediction based on activation magnitude and the rest are set to zero. ATE is averaged over all frames in KITTI test sequences 09 and 10.

more than the rotational loss term toward the ego-motion prediction accuracy on the KITTI dataset.

G. The learned basis set

We visualize the EMF basis sets R and T learned by SparseMFE in Figure 11 by projecting them onto the three dimensional Euclidean space in the camera reference frame using Equation 26. It can be seen that the learned R and T are overcomplete, i.e. redundant and linearly dependent [39], [43]. The redundancy helps in two ways, first, to use different basis subsets to encode similar ego-motion so that the individual bases are not always active. Second, if some basis subsets are turned off or get corrupted by noise, the overall prediction is still robust [39], [40]. Moreover, a pair of translational and rotational bases share the same coefficient to encode ego-motion. In that sense, the bottleneck layer neurons are analogous to the parietal cortex neurons of the primate brain that jointly encode self rotation and translation [73].

An observation from Figure 11 is that the learned basis sets can be skewed if the training dataset does not contain enough ego-motion variations. In most sequences of the VKITTI dataset, the camera mostly moves with forward translation (positive Z axis). The learned translation basis set from VKITTI dataset in Figure 11(f) shows that most bases lie in the positive Z region, denoting forward translation. Although the KITTI dataset has similar translation bias, we augment the dataset with backward sequences. As a result, the translation basis set learned from the KITTI dataset does not have a skew toward forward translation, as shown in Figure 11(b).

H. Running time

TABLE V
COMPARISON OF AVERAGE INFERENCE SPEED (FRAMES PER SECOND)

Method	SparseMFE	Lv [11]	Yin [3]	Ranjan [4]
Frames/sec	9.89	9.65	10.24	10.26
Method	ERL [19]	8-pt [15]	Zhou [2]	VOSF [28]
Frames/sec	4.18	4.61	10.21	12.5

Table V lists the average inference throughput of our proposed method and the comparison methods for frames of size 256 × 832 pixels. All methods were run on a system with 12-core Intel i7 CPU of 3.5 GHz frequency, 32GB RAM, and two Nvidia GeForce 1080Ti GPUs. We implemented our method using PyTorch. For the other methods, we used the source codes released by the authors. As indicated, SparseMFE provides moderate frames per second throughput compared to the baselines, slightly slower than [2]–[4], [28] and faster than [11], [15], [19]. The proposed method first computes optic flow using PWCnet [30] to predict ego and object-motion, which limits the throughput. However, improved ego and object-motion accuracy and sparse representation make SparseMFE a favorable solution for practical applications.

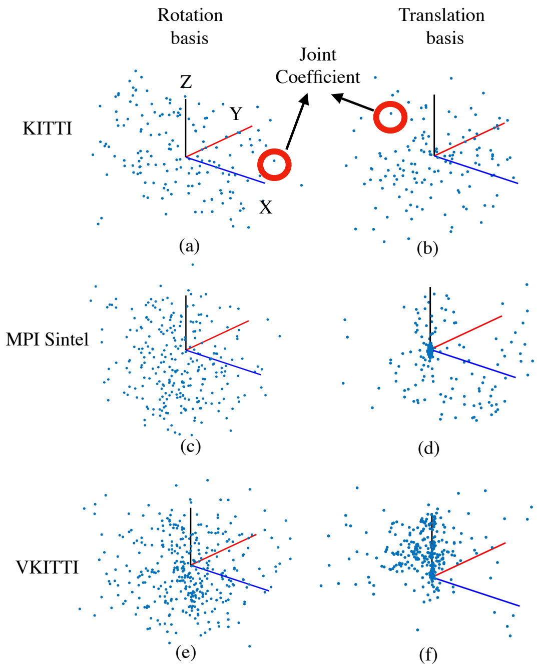


Fig. 11. Projection of the learned EMF basis set for rotational and translational ego-motion to the Euclidean space in the camera reference frame. The dots represent the learned bases and the solid lines represent the positive X, Y, and Z axes of the Euclidean space. The red circles indicate a pair of translation and rotation bases that share a same coefficient.

VI. Conclusion

Estimating camera and object velocity in dynamic scenes can be ambiguous, particularly when video frames are occupied by independently moving objects [4]. In this paper, we propose a convolutional autoencoder, called SparseMFE, that predicts translational and rotational ego-motion fields from optic flow of successive frames, from which 6DoF ego-motion parameters, pixel-wise non-rigid object motion, and dynamic region segmentation can be obtained in closed form. SparseMFE learns a sparse overcomplete basis set of ego-motion fields in its linear decoder weights, extracting the latent structures in noisy optic flow of training sequences. This is achieved using a motion field reconstruction loss and a novel differentiable sparsity penalty that approximates L0 norm for rectified input. Experimental results indicate that the learned ego-motion basis generalizes well to unseen videos in regard to the existing methods. SparseMFE achieves state-of-the-art ego-motion prediction accuracy on the KITTI dataset as well as state-of-the-art overall rotation prediction accuracy and comparable translation prediction accuracy on the MPI Sintel dataset (Tables I and II) [1], [32].

A benefit our approach, in regard to the comparison methods, is that pixel-wise object-motion can be estimated directly from predicted ego-motion field using flow parsing (Equation 7). On the realistic MPI Sintel dataset with large dynamic segments, SparseMFE achieves state-of-the-art OMF prediction performance (Table III). Moreover, compared to the baseline methods, SparseMFE object-motion prediction performance is more robust to increase in dynamic segments in videos.

Apart from achieving state-of-the-art ego and object-motion performances, our approach demonstrates an effective method for learning a sparse overcomplete basis set. This is evidenced by an ablation experiment of the basis coefficients, which shows that SparseMFE achieves state-of-the-art ego-motion prediction accuracy on KITTI odometry dataset using only the 3% most active basis coefficients, with all other coefficients set to zero (Table I and Figure 10). Moreover, the sharp sigmoid sparsity penalty proposed here is more effective in enforcing sparsity on the basis coefficients, compared to L1 and L2 norm based sparsity penalties used in common regularization methods Lasso and ridge regression, respectively (Figure 7) [64], [74]. L1 and L2 norm penalties work as shrinkage operators on the coefficient values (Figure 2). On the other hand, the differentiable sharp sigmoid penalty is uniform for most positive activations and therefore, results in fewer nonzero basis coefficients (Figures 7, 8, 9). Our approach provides a complete solution to recovering both ego-motion parameters and pixel-wise object-motion from successive image frames. Nonetheless, the regularization techniques developed in this paper are also applicable to sparse feature learning from other high dimensional data.

Acknowledgment

This work was supported in part by NSF grants IIS-1813785, CNS-1730158, and IIS-1253538.

References

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.

[2] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1851–1858.

[3] Z. Yin and J. Shi, "GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1983–1992.

[4] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black, "Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 240–12 249.

[5] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5667–5675.

[6] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, "SfM-net: Learning of structure and motion from video," *arXiv preprint arXiv:1704.07804*, 2017.

[7] Z. Kalal, K. Mikolajczyk, J. Matas *et al.*, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, p. 1409, 2012.

[8] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[9] H. Cho, Y.-W. Seo, B. V. Kumar, and R. R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2014, pp. 1836–1843.

[10] A. Byravan and D. Fox, "Se3-nets: Learning rigid body motion using deep neural networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2017, pp. 173–180.

[11] Z. Lv, K. Kim, A. Troccoli, D. Sun, J. M. Rehg, and J. Kautz, "Learning rigidity in dynamic scenes with a moving camera for 3D motion field estimation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 468–484.

[12] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems*, 2013, pp. 4363–4369.

[13] A. Bak, S. Bouchafa, and D. Aubert, "Dynamic objects detection through visual odometry and stereo-vision: a study of inaccuracy and improvement sources," *Machine Vision and Applications*, vol. 25, no. 3, pp. 681–697, 2014.

[14] G. P. Stein, O. Mano, and A. Shashua, "A robust method for computing vehicle ego-motion," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2000, pp. 362–368.

[15] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580–593, 1997.

[16] J. Fredriksson, V. Larsson, and C. Olsson, "Practical robust two-view translation estimation," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015, pp. 2684–2690.

[17] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[18] H. Strasdat, J. Montiel, and A. J. Davison, "Scale drift-aware large scale monocular SLAM," *Robotics: Science and Systems VI*, vol. 2, no. 3, p. 7, 2010.

[19] A. Jaegle, S. Phillips, and K. Daniilidis, "Fast, robust, continuous monocular egomotion computation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2016, pp. 773–780.

[20] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 0756–777, 2004.

[21] A. Concha and J. Civera, "DPPTAM: Dense piecewise planar tracking and mapping from a monocular sequence," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 5686–5693.

[22] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 834–849.

[23] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proceedings of the International Conference on Computer Vision*, 2011, pp. 2320–2327.

[24] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards internet-scale multi-view stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1434–1441.

[25] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3828–3838.

[26] M. Lee and C. C. Fowlkes, "CeMNet: Self-supervised learning for accurate continuous ego-motion estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[27] J. Quiroga, T. Brox, F. Devernay, and J. Crowley, "Dense semi-rigid scene flow estimation from RGBD images," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 567–582.

[28] M. Jaimez, C. Kerl, J. Gonzalez-Jimenez, and D. Cremers, "Fast odometry and scene flow from RGB-D cameras based on geometric clustering," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2017, pp. 3992–3999.

[29] D. J. Heeger and A. D. Jepson, "Subspace methods for recovering rigid motion I: Algorithm and implementation," *International Journal of Computer Vision*, vol. 7, no. 2, pp. 95–117, 1992.

[30] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8934–8943.

[31] T. Zhang and C. Tomasi, "On the consistency of instantaneous rigid motion estimation," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 51–79, 2002.

[32] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proceedings of the European Conference on Computer Vision*, 2012, pp. 611–625.

[33] A. Giachetti, M. Campani, and V. Torre, "The use of optical flow for road navigation," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 1, pp. 34–48, 1998.

[34] J. Campbell, R. Sukthankar, I. Nourbakhsh, and A. Pahwa, "A robust visual odometry and precipice detection system using consumer-grade monocular vision," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005, pp. 3421–3427.

[35] M. J. Black and P. Anandan, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Computer Vision and Image Understanding*, vol. 63, no. 1, pp. 75–104, 1996.

[36] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia, "Every pixel counts: Unsupervised geometry learning with holistic 3D motion understanding," *arXiv preprint arXiv:1806.10556*, 2018.

[37] H.-Y. F. Tung, A. W. Harley, W. Seto, and K. Fragkiadaki, "Adversarial inverse graphics networks: Learning 2D-to-3D lifting and image-to-image translation from unpaired supervision," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4364–4372.

[38] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade, "Three-dimensional scene flow," in *Proceedings of the IEEE International Conference on Computer Vision*, 1999, pp. 722–729.

[39] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.

[40] E. Simoncelli, W. Freeman, E. Adelson, and D. Heeger, "Shiftable multiscale transforms," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 587–607, 1992.

[41] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[42] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, 2005.

[43] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[44] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.

[45] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2011, pp. 215–223.

[46] A. Makhzani and B. J. Frey, "Winner-take-all autoencoders," in *Advances in Neural Information Processing Systems*, 2015, pp. 2791–2799.

[47] H. Lee, C. Ekanadham, and A. Y. Ng, "Sparse deep belief net model for visual area V2," in *Advances in Neural Information Processing Systems*, 2008, pp. 873–880.

[48] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[49] G. Cheng, P. Zhou, and J. Han, "Duplex metric learning for image set classification," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 281–292, 2017.

[50] P. Zhou, J. Han, G. Cheng, and B. Zhang, "Learning compact and discriminative stacked autoencoder for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4823–4833, 2019.

[51] J. Han, D. Zhang, X. Hu, L. Guo, J. Ren, and F. Wu, "Background prior-based salient object detection via deep reconstruction residual," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 8, pp. 1309–1321, 2014.

[52] J. Deng, Z. Zhang, E. Marchi, and B. Schuller, "Sparse autoencoder-based feature transfer learning for speech emotion recognition," in *Proceedings of the Humaine Association Conference on Affective Computing and Intelligent Interaction*, 2013, pp. 511–516.

[53] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2011, pp. 315–323.

[54] T. Zhang and C. Tomasi, "Fast, robust, and consistent camera motion estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1999, pp. 164–170.

[55] P. A. Warren and S. K. Rushton, "Optic flow processing for the assessment of object movement during ego movement," *Current Biology*, vol. 19, no. 18, pp. 1555–1560, 2009.

[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[57] R. A. Choudrey, "Variational methods for bayesian independent component analysis," Ph.D. dissertation, University of Oxford, 2002.

[58] P. Tseng, "Further results on stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 888–899, 2009.

[59] B. Wohlberg, "Noise sensitivity of sparse signal representations: Reconstruction error bounds for the inverse problem," *IEEE Transactions on Signal Processing*, vol. 51, no. 12, pp. 3053–3060, 2003.

[60] P. J. Huber, *Robust Statistics*. John Wiley & Sons, 2004.

[61] V. Barnett, T. Lewis, and F. Abeles, "Outliers in Statistical Data," *Physics Today*, vol. 32, p. 73, 1979.

[62] A. Miller, *Subset Selection in Regression*. Chapman and Hall/CRC, 2002.

[63] D. Bertsimas, A. King, and R. Mazumder, "Best subset selection via a modern optimization lens," *The Annals of Statistics*, vol. 44, no. 2, pp. 813–852, 2016.

[64] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[65] C. Louizos, M. Welling, and D. P. Kingma, "Learning sparse neural networks through L_0 regularization," *arXiv preprint arXiv:1712.01312*, 2017.

[66] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[67] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573–580.

[68] J. Wulff, L. Sevilla-Lara, and M. J. Black, "Optical flow in mostly rigid scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4671–4680.

[69] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4040–4048.

[70] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3061–3070.

[71] N. Jiang, W. Rong, B. Peng, Y. Nie, and Z. Xiong, "An empirical analysis of different sparse penalties for autoencoder in unsupervised feature learning," in *Proceedings of the International Joint Conference on Neural Networks*, 2015, pp. 1–8.

[72] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, no. Nov, pp. 1457–1469, 2004.

[73] A. Sunkara, G. C. DeAngelis, and D. E. Angelaki, "Joint representation of translational and rotational components of optic flow in parietal cortex," *Proceedings of the National Academy of Sciences*, vol. 113, no. 18, pp. 5077–5082, 2016.

[74] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

Hirak J. Kashyap is a PhD candiate in Computer Science at University of California, Irvine and part of Cognitive Anteater Robotics Lab (CARL). His research interests are brain inspired neural models of computer vision and machine learning. Previously he obtained an MTech with Silver Medal in Computer Science and Engineering from National Institute of Technology, Rourkela, India/Instituto Superior Tecnico, Portugal in 2014. Prior to that he obtained a BTech with Gold Medal from Tezpur University, India in 2012, where he also worked as a machine learning research fellow during 2014-15.

Charless C. Fowlkes is a Professor in the Department of Computer Science and director of the Computational Vision Lab at the University of California, Irvine. Prior to joining UC Irvine, he received his PhD in Computer Science from UC Berkeley in 2005 and a BS with honors from Caltech in 2000. Dr. Fowlkes is the recipient of the Helmholtz Prize in 2015 for fundamental contributions to computer vision in the area of image segmentation and grouping, the David Marr Prize in 2009 for work on contextual models for object recognition, and a National Science Foundation CAREER award. He currently serves on the editorial boards of IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE-TPAMI) and Computer Vision and Image Understanding (CVIU).

Jeffrey L. Krichmar received a B.S. in Computer Science in 1983 from the University of Massachusetts at Amherst, a M.S. in Computer Science from The George Washington University in 1991, and a Ph.D. in Computational Sciences and Informatics from George Mason University in 1997. He currently is a professor in the Department of Cognitive Sciences and the Department of Computer Science at the University of California, Irvine. Krichmar has nearly 20 years experience designing adaptive algorithms, creating neurobiologically plausible network simulations, and constructing brain-based robots whose behavior is guided by neurobiologically inspired models. His research interests include neurorobotics, embodied cognition, biologically plausible models of learning and memory, neuromorphic applications and tools, and the effect of neural architecture on neural function. He is a Senior Member of IEEE.