2018 IEEE Winter Conference on Applications of Computer Vision

An Order Preserving Bilinear Model for Person Detection in Multi-Modal Data

Oytun Ulutan*[1], Benjamin S. Riggan[2], Nasser M. Nasrabadi[3] and B. S. Manjunath[1]

[1]Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA
[2]US Army Research Lab, Adelphi, MD
[3]West Virginia University, Morgantown, WV

Abstract

We propose a new order preserving bilinear framework that exploits low-resolution video for person detection in a multi-modal setting using deep neural networks. In this setting cameras are strategically placed such that less robust sensors, e.g. geophones that monitor seismic activity, are located within the field of views (FOVs) of cameras. The primary challenge is being able to leverage sufficient information from videos where there are less than 40 pixels on targets, while also taking advantage of less discriminative information from other modalities, e.g. seismic. Unlike state-of-the-art methods, our bilinear framework retains spatio-temporal order when computing the vector outer products between pairs of features. Despite the high dimensionality of these outer products, we demonstrate that our order preserving bilinear framework yields better performance than recent orderless bilinear models and alternative fusion methods. Code is available at https://github.com/oulutan/OP-Bilinear-Model

1. Introduction

Human detection is a frequently studied problem, especially in the context of surveillance applications [3, 5, 6, 25]. In our work, we are interested in cases where visual detectors fail due to insufficient number of pixels on the target (i.e., low resolution). Therefore, our objective is to provide a detection framework that is robust to challenging conditions, such as few pixels on target, by leveraging multi-modal sensor data.

Low-resolution videos can be generated from a scenario where a high resolution camera with a wide field of view (FOV) placed close to a power source but far away from the field with targets. This requires visual detection frameworks to search for small (few pixels) objects on a large field. Seismic sensors on the other hand can provide reliable information about their close surroundings and can easily be distributed on a large field. This allows the data from a seismic sensor to improve the detection of cameras in regions where camera view and sensor range intersects.

In this work, we consider a typical surveillance setting (e.g., border patrol) where multiple sensors and cameras are used to monitor a particular area. Traditional methods for person detection that rely only upon visual cues tend to perform poorly on low resolution imagery data from our dataset. For this reason, we aim to jointly leverage corresponding sensor (e.g., seismic) and imaging data (Fig. 1).

In this context, we propose a new order-preserving bilinear fusion model for person detection, leveraging pairwise interactions between convolutional features in a new way. We demonstrate that sparse feature selection combined with bilinear fusion selects the optimal combinations of spatio-temporal features. We show that the proposed fusion method is differentiable and the final model is end-to-end trainable. The performance of our fusion model is tested in a new multi-modal person detection dataset with syncronized seismic sensors and video cameras [18]. The dataset is available through requests[1]. Our experimental results show that our model achieves better detection accuracy and reduced false positive rates compared to the state of the art fusion methods.

2. Related Work

In a surveillance setting, traditional detection methods for multimodal sensor data depend on hand-crafted features such as frequency domain analysis [6, 25], Symbolic Dynamic Filtering [3], and Cepstral features [20]. Damarla *et al*. [5] extracts and fuses hand-crafted features from multiple different modalities for person detection. Recently, with the advances of computational hardware and the increase

*ulutan@ece.ucsb.edu

[1]The dataset can be obtained by sending an email to benjamin.s.riggan.civ@mail.mil

978-1-5386-4886-5/18/$31.00 ©2018 IEEE
DOI 10.1109/WACV.2018.00132

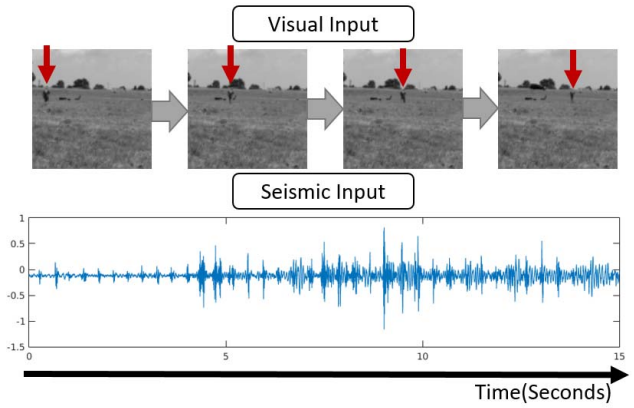

Figure 1. An example of time synchronized seismic and visual data. Frames are cropped centering the seismic sensor's location. As a person gets closer to the center of image, the amplitude of the seismic signals increase. Red arrows indicate the person.

of available data, feature learning has been integrated with classification to achieve end-to-end trainable systems [14].

Ngiam *et al*. [19] analyzed the relations between different modalities in deep networks and showed that cross-modality feature learning can improve single modality performance. Riggan *et al*. [22] used Coupled AutoEncoders for cross-modal face recognition fusing visible and thermal imaging. [9, 27] achieved fusion by concatenating features from CNNs trained on RGB and depth images.

Fusing different features extracted from a single modality has been achieved using multiple different methods which are also applicable to multi-modal fusion. [26, 32] achieved late fusion between optical flow and RGB by averaging the confidence scores of single CNNs for video classification. Karpathy *et al*. [12] analyzed concatenating features from different time instances and trained fully connected layers to fuse information over time in a video.

Bilinear models were first analyzed by Tenenbaum and Freeman [30] to manipulate two factors from images, style and content. Recently bilinear models have achieved success in multiple tasks. Lin *et al*. [16] fused two convolutional neural networks to obtain orderless descriptors and improved results in fine-grained visual recognition. Carreira *et al*. [4] used second order statistics of the local descriptors for semantic segmentation. RoyChowdhury *et al*. [23] used bilinear CNNs to improve results in face identification tasks. Gao *et al*. [10] improves the bilinear methods by developing a compact pooling method.

The main difference between recent bilinear methods [4, 10, 16, 23] and our method is that we use the outer product of vectors and obtain the pairwise feature interactions at each spatio-temporal indices. This is in contrast with these methods that use pooling methods over all indices and obtain an 'orderless' descriptor without preserving the order.

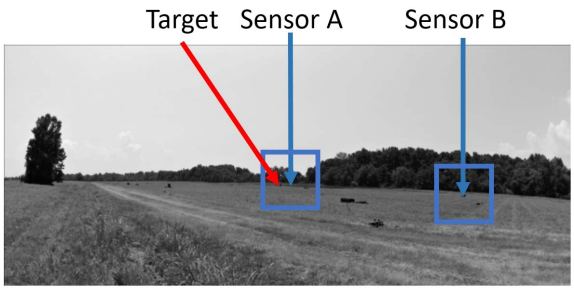

Figure 2. ROIs seen from a wider camera view. Each ROI is located around the sensor locations which are known a priori. Notice that in the figure, target is within the Sensor A's region which produces a positive sample whereas the sample from Sensor B is a negative sample. There are multiple ROIs within this camera frame but only two of them are shown.

3. Technical Approach

The goal is to detect the region of interest(ROI) with a person walking in a field that is being monitored by a multi-modal sensor network data consisting of video cameras and seismic geophones. In this context, a ROI is any contiguous set of pixels and corresponding sensor data. Detection is defined on ROIs with corresponding camera and sensor pairs. We pose this as a binary classification problem for each ROI. Fig. 2 shows example ROIs located around the sensor locations which are known a priori. The inputs to our model are a single optical flow frame and its corresponding seismic signal for the same time interval.

In the following sections, we define the problem as a general multi-modal fusion problem and derive our fusion model by explaining each of the modules.

3.1. Problem Definition

Let X and Z be two sets of local descriptors extracted from two different modalities. Each descriptor $\boldsymbol{x}_{u_x,v_x,t_x} \in X$ represents the feature vector for the spatio-temporal voxel defined by the indices u_x, v_x, t_x, and similarly for the other modality $\boldsymbol{z}_{u_z,v_z,t_z} \in Z$. Let $\boldsymbol{x}$ and $\boldsymbol{z}$ be $N \times 1$ and $M \times 1$ dimensional feature vectors respectively.

Our goal is to develop a fusion algorithm $O = f(X, Z)$ such that spatio-temporal indices are preserved. For every spatio-temporal index from both modalities, we have the output feature vector:

$$\boldsymbol{o}_{u_x,v_x,t_x,u_z,v_z,t_z} = f(\boldsymbol{x}_{u_x,v_x,t_x}, \boldsymbol{z}_{u_z,v_z,t_z}) \quad (1)$$

where $\boldsymbol{o}_{u_x,v_x,t_x,u_z,v_z,t_z} \in O$ are the local descriptors of the output. If the input modalities are synchronized in time and space then we will have $(u_x, v_x, t_x) = (u_z, v_z, t_z) = (u, v, t)$. Indices from Eq. 1 simplifies into:

$$\boldsymbol{o}_{u,v,t} = f(\boldsymbol{x}_{u,v,t}, \boldsymbol{z}_{u,v,t}) \quad (2)$$

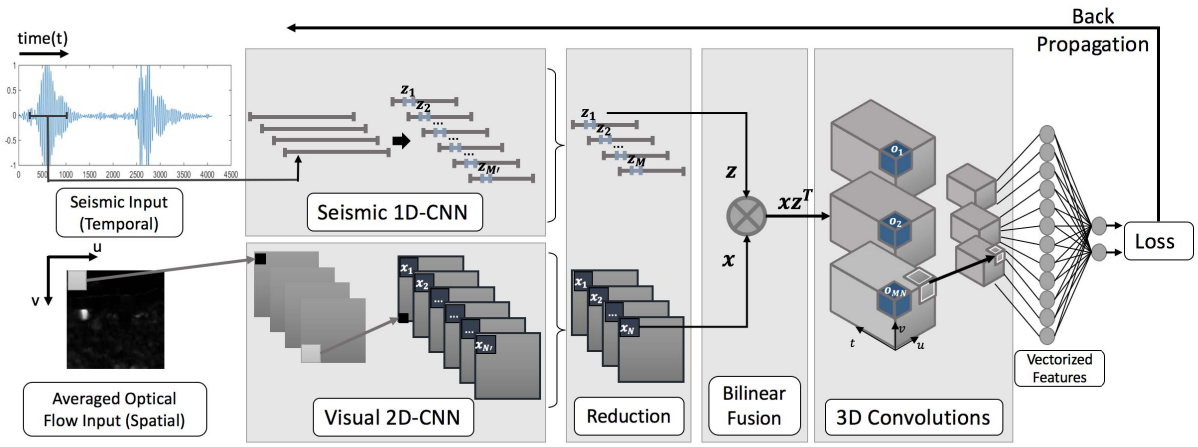

Figure 3. Order Preserving Bilinear model: Data from both modalities go through their respective CNN streams. Resulting features are compressed into lower dimensional vectors by sparse feature reduction and then fused by taking outer product at every spatio-temporal index. Since the order is preserved, 3D convolutions are leveraged. Since every module is differentiable, the whole model is trained end-to-end.

Furthermore, if we let modality X to be a spatial signal and modality Z to be a temporal signal. That gives $t_x = 1$ and $u_z = v_z = 1$ and simplifies the Eq. 1 into:

$$\boldsymbol{o}_{u,v,t} = f(\boldsymbol{x}_{u,v}, \boldsymbol{z}_t) \tag{3}$$

Eq. 3 defines the local descriptor which is the output of the fusion method. Note that in both Eq. 2 and Eq. 3, the calculation of $\boldsymbol{o}_{u,v,t}$, depends on the input values at indices u, v, t, which gives an ordered descriptor. Ordered descriptors allow us to exploit the relations between neighboring terms by using methods such as 3D convolutions. The goal is to detect targets using spatial images and temporal seismic sensor data, which fits into the formulation in Eq. 3.

Our model is organized into four sub-components as shown in Fig. 3: **1)** input sensor signals are processed by dedicated CNNs for each modality (Section 3.2); **2)** at each spatial and temporal index, feature vectors are compressed in their depth dimension (Section 3.3); **3)** outer product is used in each spatio-temporal index to obtain the bilinear feature vector (Section 3.4); and **4)** 3D convolutions are used to leverage neighborhood relations of spatio-temporally ordered terms (Section 3.5).

3.2. CNN Features

In previous works, CNNs have been shown to extract useful features for variety of tasks on spatial [11], temporal [2] and spatio-temporal [26] modalities. CNNs extract local feature vectors at each spatio-temporal index (u, v, t). The size of the vector depends on the number of filters in the last convolutional layer, i.e., depth of the layer. For each modality at each index u, v, t we have:

$$\boldsymbol{x}'_{u,v} = [x'_1, x'_2, ...x'_{N'}]^T, \tag{4}$$

$$\boldsymbol{z}'_t = [z'_1, z'_2, ...z'_{M'}]^T. \tag{5}$$

The prime ($'$) notations refer to the values before feature selection.

3.3. Sparse Feature Selection

The proposed fusion method, explained in Section 3.4, generates a high dimensional vector. Using high dimensional vectors are computationally challenging and can be prone to overfitting due to increased number of parameters. Within these large number of features, we want to prioritize which feature pairs are more useful (further discussed in Section 3.4.2). Therefore, we implement an efficient way to perform spatio-temporal feature selection by combining sparse 1×1 convolutions with bilinear fusion. Moreover, this method maintains spatio-temporal order. The goal is to compress the input vector to reduce the dimensions from Eq. 4. From here on, we generically use the term '**reduction**' to represent both feature selection and dimensionality reduction operations. We define our reduction function $r(.)$ as:

$$r(\boldsymbol{x}'_{u,v}) = \boldsymbol{x}_{u,v} = [x_1, x_2, ..., x_N]^T, \tag{6}$$

where $N < N'$ so that we obtain a more compact feature vector and we define the each reduced component x_i as the linear combinations of the original vector:

$$x_i = ReLU(\sum_{k=1}^{N'} w_{ik}^x x'_k) = max(0, \sum_{k=1}^{N'} w_{ik}^x x'_k), \tag{7}$$

where weights w^x_{ik} are learned over the training and the norm of the weights are regularized using $L1$ normalization. Compared to $L2$ normalization or without normalization, $L1$ normalization generates a more sparse set of weights which forces the network to 'choose' the features that will be included in the summation. By $L1$ regularization, the weights $|w^x_{ik}|$ are mostly close to zero except a few weights that are multiplying essential set of features x'_k. This is similar to LASSO [35, 17] and provides a feature selection operation. Similarly for the second modality, reducing the vector from Eq. 5:

$$r(\boldsymbol{z'_t}) = \boldsymbol{z}_t = [z_1, z_2, ...z_M]^T, \tag{8}$$

$$z_i = ReLU(\sum_{k=1}^{M'} w^z_{ik} z'_k) = max(0, \sum_{k=1}^{M'} w^z_{ik} z'_k). \tag{9}$$

3.4. Order Preserving Bilinear Fusion

Reduced CNN features (Eq. 6 and Eq. 8) are fed into the fusion layer. At each spatial and temporal index, local feature vectors from both modalities are fused by taking the outer product. The fusion function at each spatio-temporal index $u \in U, v \in V, t \in T$ can be written as:

$$\boldsymbol{o}_{u,v,t} = f(\boldsymbol{x}_{u,v}, \boldsymbol{z}_t) = vectorize(\boldsymbol{x}_{u,v} \boldsymbol{z}_t^T) \tag{10}$$

At each index, we have length N vector $\boldsymbol{x_{u,v}}$ and length M vector $\boldsymbol{z_t}$. Outer product between these feature vectors generate the $N \times M$ second order pairwise features matrix:

$$\boldsymbol{x}_{u,v} \boldsymbol{z}_t^T = \begin{bmatrix} x_1 z_1 & x_1 z_2 & ... & x_1 z_M \\ x_2 z_1 & x_2 z_2 & ... & x_2 z_M \\ \vdots & & \ddots & \vdots \\ x_N z_1 & x_N z_2 & ... & x_N z_M \end{bmatrix}. \tag{11}$$

We stack the rows together in lexicographical order, i.e., $N \times M$ dimensional matrix into an $MN \times 1$ vector. This gives the fused feature vector at each spatio-temporal index.

$$\begin{aligned} &\boldsymbol{o}_{u,v,t} = [o_1, o_2, ...o_{MN}]^T = \\ &\begin{bmatrix} x_1 z_1 & ... & x_1 z_M & ... & x_N z_1 & ... & x_N z_M \end{bmatrix}^T \end{aligned} \tag{12}$$

We repeat this operation for each spatial index u, v and temporal index t and obtain the fused second order feature vector at every combination of indices u, v, t.

3.4.1 Differentiability for Backpropagation

This fusion operation is differentiable for gradient operations and it is end-to-end trainable. In this section we show how the gradient can be backpropagated to each modality stream. Let L denote the cross-entropy loss function. Then by chain rule, we obtain:

$$\frac{\partial L}{\partial \boldsymbol{x}_{u,v}} = \frac{\partial L}{\partial \boldsymbol{o}_{u,v,t}} \frac{\partial \boldsymbol{o}_{u,v,t}}{\partial \boldsymbol{x}_{u,v}} = \frac{\partial L}{\partial \boldsymbol{o}_{u,v,t}} \begin{bmatrix} \frac{\partial o_1}{\partial x_1} & ... & \frac{\partial o_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial o_{MN}}{\partial x_1} & ... & \frac{\partial o_{MN}}{\partial x_N} \end{bmatrix} \tag{13}$$

where $\frac{\partial L}{\partial \boldsymbol{o}_{u,v,t}}$ can be calculated using chain rule of derivatives for layers between loss L and the outer product. Each partial derivative in the matrix can be written as:

$$\frac{\partial o_p}{\partial x_r} = \frac{\partial (x_s z_q)}{\partial x_r} \tag{14}$$

where $p = 1, .., MN$, $q = 1, .., M$, $r = 1, .., N$ and $s = 1, .., N$. For $s = r$, this simplifies into:

$$\frac{\partial o_p}{\partial x_r} = \frac{\partial (x_r z_q)}{\partial x_r} = z_q \tag{15}$$

For $s \neq r$, Eq. 14 becomes 0. Gradients before this layer can also be calculated by the regular CNN chain rule. $\frac{\partial L}{\partial \boldsymbol{z}_t}$ can be calculated similarly for the second modality.

3.4.2 Effects of Feature Selection

The outer product generates a high dimensional feature vector at each index $\boldsymbol{o}_{u,v,t}$. To handle the high dimensionality, we pool the convolutional features before the outer product operation by feature selection (Section 3.3). When $\sum_{k=1}^{N'} w^x_{ik} x'_k > 0$ and $\sum_{l=1}^{M'} w^z_{jl} z'_l > 0$, multiplying the terms from Eq. 7 and Eq. 9 yields for each $x_i z_j$ in Eq. 12:

$$x_i z_j = \sum_{k=1}^{N'} w^x_{ik} x'_k \times \sum_{l=1}^{M'} w^z_{jl} z'_l = \sum_{k=1}^{N'} \sum_{l=1}^{M'} w^{xz}_{kl} x'_k z'_l \tag{16}$$

where each $w^{xz}_{kl} = w^x_{ik} w^z_{jl}$. Otherwise, the $x_i z_j = 0$. This shows that output of reduced fusion operation is linear combinations of the second order interactions of the original feature vectors before the feature selection operation x'_k, z'_l.

Weights of 1×1 convolutions w^x_{ik}, w^z_{jl} are trained with $L1$ regularization, hence they are individually sparse (Section 3.3). Therefore, this ensures that when multiplied, the produced set of weights are also sparse and the product $w^x_{ik} w^z_{jl}$ is non-zero only if corresponding features k, l from each modality x'_k, z'_l are individually important for the task which is similar to sparse representations[21].

3.5. 3D Convolutions

Since the outer product operation is repeated for every combination of the spatial (u, v) and temporal (t) indices, output of the fusion operation is a spatio-temporal feature

tensor as shown in Fig. 3. This tensor allows us to use shared weights that stride across spatial and temporal dimensions, i.e., 3D convolutions, to reduce the total number of parameters and chances of overfitting by exploiting spatio-temporal correlations. In the tensor, at every spatio-temporal index, we have a feature vector of length MN which is the output of the outer product between length M vector $\boldsymbol{x}$ and length N vector $\boldsymbol{z}$. In the 3D convolutions, this dimension corresponds to the depth of input. The intuition behind keeping the spatio-temporal order is that certain activations in certain combinations of spatial and temporal indices complement each other. By having all the second order pairs as features at each index, we can find feature pairs that are sufficiently discriminative.

4. Implementation Details

The data is collected in a sensor field with 16 seismic sensors and 4 video cameras[18]. Seismic sensors are placed on a grid and the video cameras are placed outside the sensor field, observing it from different directions. In a surveillance setting, viewpoints and conditions vary for cameras and sensors, and surroundings can change the detected signature of the seismic sensors. To take this into account and to make the model generalizable, we split the data such that camera views (angle, background) and seismic sensors that are used in test set are different than the ones in training set. Each person in the field wears a GPS sensor. Using the location information we label the samples as positive when a person is within 15 meters of a seismic sensor. This results in 69483 negative and 16481 positive samples in training set and 26064 negative and 6440 positive samples in test set.

Videos are recorded at 30 frames per second at 640×360 resolution and seismic signals captured at 4096 Hz sampling rate. A 100×100 region that is centered at a seismic sensor location (known a priori) is cropped from each camera frame. From seismic signals we extract our data points as 1 second intervals with 50% overlap. For the video data, we compute optical flow(OF). For each seismic signal centered at time t, OF frames are computed from the seismic sensor's corresponding region over the time interval $[t-1, t+1]$. Magnitudes of these OF frames are averaged and used as the input to the proposed method. By averaging OF frames the spatio-temporal modality video is compressed into a spatial representation that encodes the temporal motion information. The reasoning behind this approach is mostly computational. This approach is further investigated and compared to LSTMs in Section 5.6.

To measure the performance of our methods, we report the precision, recall and F1-score values for the positive class. Recall values measure the detection accuracy whereas Precision measures the rate of false positives. In a data as unbalanced as ours, reporting both recall and precision becomes important. Since the negative class has significantly more samples than the positive class, high accuracy in detecting negative samples might still mean high false positive rates. For example 90% accuracy in negative test samples still means $26064 \times 0.10 = 2606$ false positives which is 40% of the total number of positive samples.

All models are trained using TensorFlow [1] and optimized using ADAM optimizer[13].

4.1. Single Modality CNNs

For extracting useful features from both seismic and visual data, we independently train modality specific CNNs for the detection task and analyze their performances.

Since there are no similar works using seismic sensors to be used for transfer learning, a randomly initialized 1-dimensional CNN is trained for the seismic modality. For the visual modality, we leverage the Inception V3 network architecture explained in [29] and initialize the network with weights that are pretrained for ImageNet [24]. Since this network is trained on RGB images and trained to detect ImageNet-specific features, we use earlier layers instead of the full architecture. Earlier layers in a CNN extract basic features such as edges, corners and these features are more generalizable. In [34] the authors quantified the generality and specificity of the layers and showed that the earlier layers are more generalizable. In [33] an OF CNN for action recognition is initialized using weights from a model trained for ImageNet. In our case, for the OF CNN we use the first five convolutional layers from Inception V3 model and initialize the weights from a ImageNet trained model.

4.2. Order Preserving Bilinear Fusion

The proposed approach (Fig. 3) consists of two dedicated streams of CNNs for each modality (Section 3.2), their corresponding sparse feature selection layers (Section 3.3), outer product between outputs of the two streams at each spatio-temporal index to preserve the order (Section 5.2), 3D convolutions (Section 3.5) and a final fully connected layer for classification. We refer this model as Order Preserving (OP) Bilinear Model.

Architectures used for the modality dedicated CNN streams are the same architectures as the single modality models defined in previous section. This allows us to initialize the model weights with pretrained weights from single modality models. Each CNN stream is followed by sparse feature selection and the fusion is achieved by order preserving outer product operation. Since the proposed outer product fusion is differentiable, as shown in Section 3.4.1, the whole model is fine-tuned in an end-to-end fashion.

5. Experiments and Results

In the following sections, we conduct a series of experiments to analyze the performance of each module in our

Model	Recall	Precision	F1-Score
Seismic	0.90	0.87	0.89
Seismic Reduced	0.89	0.86	0.88
Visual	0.82	0.89	0.86
Visual Reduced	0.78	0.89	0.83
OP-Bilinear Fusion	**0.97**	**0.96**	**0.96**

Table 1. Precision, Recall and F1-Score values for single modality models and the proposed fusion method.

Distances From **Cameras**(meters)	50-80	80-110	110-140
Visual Reduced	0.96	0.93	0.74
OP-Bilinear Fusion	**0.98**	**0.96**	**0.95**
Distances From **Sensors**(meters)	0-5	5-10	10-15
Seismic Reduced	0.96	0.93	0.80
OP-Bilinear Fusion	**0.99**	**0.97**	**0.93**

Table 2. Recall rates for different distances from the cameras and seismic sensors. Even though the performance of OP-Bilinear model also decreases with range, the change is not as significant since it incorporates the information from the complementary modality.

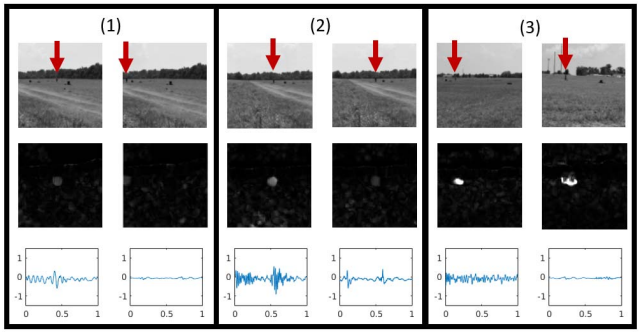

Figure 4. Examples of correct detections from the OP-Bilinear Model where single modality models fail. Red arrows indicate the targets.

Model	Recall	Precision	F1-Score
End-to-End OP-Bilinear	0.95	0.95	0.95
OP-Bilinear Fusion	**0.97**	**0.96**	**0.96**

Table 3. Precision, Recall and F1-Score values for different initialization methods.

method. First, we report experiments on the single modality CNNs and analyze the effects of dimensionality reduction. Then, we demonstrate the superior performance of the proposed bilinear fusion method compared to single modality models and alternative fusion methods. Furthermore, we compare the order-preserving methods that exploit 3D convolutions with their fully connected counterparts. Finally, we compare our visual approach with a LSTM approach.

5.1. Impact of Sparse Feature Reduction

For each modality, two different models are trained. Initial models use convolutional layers followed by fully connected layers. These models are labeled as 'Seismic' and 'Visual' in the tables. Additionally, we train models with the sparse feature selection method explained in Section 3.3. We add the feature selection layer between convolutional and fully connected layers. These models are labeled as 'Seismic Reduced' and 'Visual Reduced' in the tables.

Table 1 implies that sparse feature selection (reduced models) from Section 3.3 provide a slight trade-off in performance for computation efficiency for computing bilinear features. In the Visual CNN, the reduction in number of parameters are significant with this reduction method.

5.2. Fusion Compared to Single Modalities

Table 1 compares the proposed fusion method against single modality models and shows that the fusion method provides the best performance in accuracy (Recall) and false positive rate (Precision). Fig. 5 compares the method with other select models by plotting Precision-Recall curves. This plot demonstrates that our model is the best performing classifier since OP-Bilinear curve achieves the best Precision-Recall trade-off at every point.

Fig. 4 shows 3 sets of data samples. The first set shows the cases where both Visual and Seismic models fail but the fusion model correctly detects the target. In both samples, OF captures a weak motion and seismic sensor captures noise-like signals, but the fusion method detects the person nevertheless. The second set shows the samples where Visual model fail but Seismic and OP-Bilinear models correctly detects the target. Similarly, the third set shows the samples where Seismic model fails but Visual and OP-Bilinear model detects the target. This demonstrates that the fusion model achieves robust detection even when the input from a single sensor deteriorates.

We further compare the fusion model to the single modality models. As the distance between the target and the sensors increase, the performance deteriorates. Table 2 demonstrates that the proposed OP-Bilinear Fusion model is more robust to distance. The fusion model can effectively incorporate the information from the complementary modality when one modality degrades with range.

5.3. Effects of Initialization

In Section 3.4.1, we have derived the gradient for the proposed outer product operation. Since the gradient exists, the whole model is end-to-end trainable. In the previous section, we showed the results of the proposed method by initializing the model with single modality CNN model weights and fine-tuning the whole model. To investigate end-to-end training, we train a model using the same architecture, except the filter weights for the model are randomly

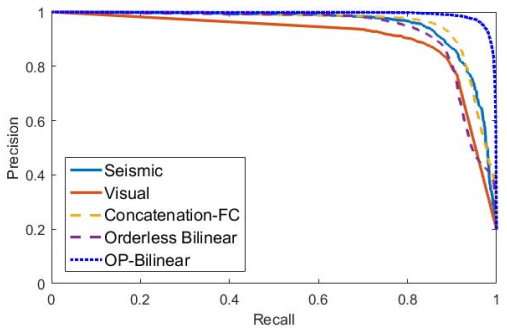

Figure 5. Precision-Recall curves show that OP-Bilinear Fusion achieves the best detection rate and fewest false positives.

initialized. Table 3 compares the performance of the end-to-end trained network with the model that is fine-tuned on pre-trained weights. This shows that pre-training achieves a slightly better performance than random initialization.

5.4. Comparisons with Fusion Methods

We compare our proposed OP-Bilinear Model with multiple late fusion approaches, feature concatenation approaches and state of the art Orderless Bilinear methods.

Average Fusion: We compare our results with a simple confidence score averaging late fusion method. This is a widely used method due to its simplicity [12, 26, 32]. In this method, we take the confidence scores from individually trained models 'Seismic' and 'Visual' from Section 4.1 and average them to get the final score for each datapoint. Results are labeled as 'Average Fusion' in Table 4.

Dempster Shafer Fusion: We compare our results with a more sophisticated late fusion method, Demster Shafer theory [7]. This theory is a framework for reasoning with uncertainty and generally applicable to sensor fusion models. We implement this model similar to [15]. We assume more uncertainty for visual modality than seismic modality, e.g. 35% versus 15%, due to noise and resolution. The results of this framework are shown in Table 4 with label 'Demster Shafer Fusion'.

Compared with these late fusion methods, our proposed fusion method is able to model the relations between the modalities and achieve better performance. Table 4 demonstrates that the proposed OP-Bilinear fusion model achieves higher detection rate (Recall) with lower false positive rate (Precision).

Concatenation-Fully Connected: Many multi-modal fusion [9, 27, 31] and feature fusion [12] methods concatenate the feature vectors from CNNs and classify the results using fully connected layer. This simple stacking of feature vectors compresses the spatial or temporal order since the features at every index are stacked into a single vector. Note that such operation does not exploit correlations in the spatial or temporal order. The output of this fusion can be expressed as:

$$\begin{aligned} \boldsymbol{o}_{u,v,t} = [o_1, o_2, ...o_{M+N}]^T = \\ \begin{bmatrix} x_1 & ... & x_N & z_1 & ... & z_M \end{bmatrix}^T \end{aligned} \tag{17}$$

and vectors at each spatial and temporal indices are also stacked into a vector as:

$$\begin{bmatrix} \boldsymbol{o}_{1,1,1} & ... & \boldsymbol{o}_{u,v,t} & ... & \boldsymbol{o}_{U,V,T} \end{bmatrix}^T \tag{18}$$

Results of this model are provided in Table 4 and Fig. 5 under the label 'Concatenation-FC'. The results show that the OP-Bilinear method achieves better performance than the Concatenation model by extracting bilinear features and preserving order.

Orderless Bilinear Descriptor: Bilinear pooling methods [16, 23, 4, 10] use sum pooling over spatial indices to pool the second order feature tensor into an orderless feature representation. Inspired by this idea, we sum the output of the outer product operation $\boldsymbol{x}_{u,v}\boldsymbol{z}_t^T$ from every spatial and temporal indices.

$$\sum_{u,v,t} \boldsymbol{x}_{u,v}\boldsymbol{z}_t^T = \begin{bmatrix} x_1z_1 & x_1z_2 & ... & x_1z_M \\ x_2z_1 & x_2z_2 & ... & x_2z_M \\ \vdots & & \ddots & \vdots \\ x_Nz_1 & x_Nz_2 & ... & x_Nz_M \end{bmatrix} \tag{19}$$

Results of these fusion models can be seen in Table 4 and Fig. 5. The results demonstrate that the proposed method achieves the highest recall and precision rate among alternative fusion methods. Additionally, we observe that Orderless Bilinear model performs worse than the Concatenation. We believe that summation approach over all the spatio-temporal indices in the former model loses the information instead of achieving fusion.

Model	Recall	Precision	F1-Score
Average Fusion [26, 12]	0.90	0.92	0.91
Dempster Shafer Fusion [15]	0.93	0.95	0.94
Concatenation-FC [27, 12]	0.91	0.89	0.90
OP-Concatenation	0.93	0.90	0.91
Orderless Bilinear [16]	0.87	0.90	0.88
OP-Bilinear Fusion	**0.97**	**0.96**	**0.96**

Table 4. Precision, Recall and F1-Score values for different fusion methods and proposed method. Cited papers use similar (multi-modal or feature) fusion methods to our experimentation models.

5.5. Impact of 3D Convolutions

In this section we investigate the merits of 3D convolutions. Since the model is order preserving (OP), output of the fusion model is a spatio-temporal tensor. This tensor allows us to leverage 3D convolutions to reduce the total number of parameters and chances of overfitting by exploiting

Model	Recall	Precision	F1-Score
Concatenation-FC	0.91	0.89	0.90
OP-Concatenation	0.93	0.90	0.91
Bilinear-FC	0.95	0.75	0.85
OP-Bilinear Fusion	**0.97**	**0.96**	**0.96**

Table 5. Precision, Recall and F1-Score values for Order Preserving (OP) fusion methods and their fully connected orderless variants. OP methods exploit 3D convolutions, other methods do not.

spatio-temporal correlations. We demonstrate this by comparing OP models that exploit 3D convolutions with corresponding fully connected models on two different fusion approaches, i.e., concatenation and bilinear feature descriptors. Table 5 demonstrates that models that preserve order achieve superior performance in both fusion approaches.

Order Preserving Concatenation: In this model, we adjust our order preserving approach to concatenation methods. We concatenate the features from each modality at every spatio-temporal index as in Eq. 17. However, instead of stacking the vectors further (as in Eq. 18), we use these concatenated vectors as spatio-temporal local descriptors with $M + N$ length feature vector $\boldsymbol{o}_{u,v,t}$ at each index (u, v, t). Since the spatio-temporal order of descriptors is preserved this allows us to use 3D convolutions to exploit correlations. Tables 4, 5 show the results of this model under the label 'OP-Concatenation' and demonstrates that order preserving concatenation performs better than simple concatenation.

Bilinear-Fully Connected: In this model, we replace the 3D convolutions from the model in Section 5.2 with fully connected layers and fine-tune the network similarly with pre-trained CNN weights. This effectively removes the weight sharing of 3D convolutions, which removes the order-preserving aspect of the model and makes the model prone to overfitting.

Table 5 demonstrates the improvement in performance with preserving order on Bilinear Feature descriptors. OP-Bilinear model results with significantly fewer false positive rates, i.e, much higher precision compared to fully-connected method.

5.6. Averaging OF and LSTM Comparison

Our visual input is the magnitudes of OF vectors averaged over a time interval. Extracting OF from low-resolution cameras generate noisy inputs. Additionally, for this application, location and existence of the motion is as important as the evolution of the motion. Spatial location of the motion captured among subsequent frames does not change drastically and averaging over a short time interval allows OF magnitudes to compress the motion captured while reducing the noise. This generates a low dimensional, compact feature description. However, a more complex and higher dimensional approach is capable of an incrementally better performance. Recurrent Neural Networks (RNNs) and Long-Short Term Memory (LSTMs) models have been shown to achieve good performance on variety of tasks [8, 28, 36]. We compare the performance of our averaged OF model with an LSTM model. In the LSTM model each input frame (OF Magnitude) goes through the convolutional part of the 'Visual' model from Section 4.1 and the outputs of the consecutive frames are fed into an LSTM cell similar to Activity Recognition model in [8]. Table 6 shows the performance of the LSTM compared to averaged OF visual model. This demonstrates that averaging reduces the dimensionality and has slightly better false positive rates compared to small improvement in detection performance of LSTMs. Additionally, for low-power strategic scenarios, processing every frame through a CNN model may not be possible(which is required in LSTM) whereas taking an average over a time interval and processing only this compact snapshot is more feasible.

Model	Recall	Precision	F1-Score
Visual	0.82	0.89	0.86
LSTM	0.86	0.86	0.86

Table 6. Comparison of the visual and LSTM model.

6. Conclusions

In this work, we introduced an OP-Bilinear Fusion method to jointly leverage sensor data and imagery. By conducting a series of experiments we analyzed the impact of each module. We demonstrated that our feature selection algorithm makes the fusion method feasible by effectively reducing dimensionality with only a small tradeoff in single modality detection performance. We showed that our fusion model performs improves performance over models trained on single modalities and demonstrated that the fusion is beneficial. We compared the proposed fusion method with the traditional multi-modal and feature fusion methods and achieved better performance with the proposed method. Finally, we compared our approach of averaging OF frames to a more complicated LSTM approach and showed that by averaging multiple OF frames the sequence information is not lost and the model performs similarly.

7. Acknowledgements

The authors thank Dr. Thyagaraju Damarla at Army Research Lab for providing the dataset and guidance in processing the data. Research was supported in part by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053 (the ARL Network Science CTA). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here in.

References

[1] M. Abadi, A. Agarwal, et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10):1533–1545, 2014.

[3] S. Bahrampour, A. Ray, S. Sarkar, T. Damarla, and N. M. Nasrabadi. Performance comparison of feature extraction algorithms for target detection and classification. *Pattern Recognition Letters*, 34(16):2126–2134, 2013.

[4] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *European Conference on Computer Vision*, pages 430–443. Springer, 2012.

[5] R. Damarla and D. Ufford. Personnel detection using ground sensors. In *Defense and Security Symposium*, pages 656205–656205. International Society for Optics and Photonics, 2007.

[6] T. Damarla, A. Mehmood, and J. Sabatier. Detection of people and animals using non-imaging sensors. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pages 1–8. IEEE, 2011.

[7] A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. *The annals of mathematical statistics*, pages 325–339, 1967.

[8] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[9] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard. Multimodal deep learning for robust rgb-d object recognition. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 681–687. IEEE, 2015.

[10] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell. Compact bilinear pooling. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

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

[14] 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.

[15] K. Lee, B. S. Riggan, and S. S. Bhattacharyya. An accumulative fusion architecture for discriminating people and vehicles using acoustic and seismic signals. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2017.

[16] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1449–1457, 2015.

[17] L. Meier, S. Van De Geer, and P. Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

[18] S. M. Nabritt, T. Damarla, and G. Chatters. Personnel and vehicle data collection at aberdeen proving ground (apg) and its distribution for research. Technical report, Army Research Lab Adelphi, MD Sensors and Electron Devices Directorate, 2015.

[19] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696, 2011.

[20] N. H. Nguyen, N. M. Nasrabadi, and T. D. Tran. Robust multi-sensor classification via joint sparse representation. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, pages 1–8. IEEE, 2011.

[21] V. M. Patel and R. Chellappa. Sparse representations, compressive sensing and dictionaries for pattern recognition. In *Pattern Recognition (ACPR), 2011 First Asian Conference on*, pages 325–329. IEEE, 2011.

[22] B. S. Riggan, C. Reale, and N. M. Nasrabadi. Coupled auto-associative neural networks for heterogeneous face recognition. *IEEE Access*, 3:1620–1632, 2015.

[23] A. RoyChowdhury, T.-Y. Lin, S. Maji, and E. Learned-Miller. Face identification with bilinear cnns. *arXiv preprint arXiv:1506.01342*, 2015.

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[25] J. M. Sabatier and A. E. Ekimov. Range limitation for seismic footstep detection. In *SPIE Defense and Security Symposium*, pages 69630V–69630V. International Society for Optics and Photonics, 2008.

[26] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[27] R. Socher, B. Huval, B. P. Bath, C. D. Manning, and A. Y. Ng. Convolutional-recursive deep learning for 3d object classification. In *NIPS*, volume 3, page 8, 2012.

[28] N. Srivastava, E. Mansimov, and R. Salakhutdinov. Unsupervised learning of video representations using lstms. In *ICML*, pages 843–852, 2015.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. *arXiv preprint arXiv:1512.00567*, 2015.

[30] J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural computation*, 12(6):1247–1283, 2000.

[31] J. Wagner, V. Fischer, M. Herman, and S. Behnke. Multispectral pedestrian detection using deep fusion convolutional neural networks. In *European Symp. on Artificial Neural Networks (ESANN)*, 2016.

[32] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao. Towards good practices for very deep two-stream convnets. *arXiv preprint arXiv:1507.02159*, 2015.

[33] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*, pages 20–36. Springer, 2016.

[34] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3320–3328. Curran Associates, Inc., 2014.

[35] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[36] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.