IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 27, 2018, accepted November 26, 2018, date of publication December 10, 2018,
date of current version January 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2885736

Multistage Fusion With Dissimilarity Regularization for SAR/IR Target Recognition

YOUNG-RAE CHO, SEUNGJUN SHIN, SUNG-HYUK YIM, KYEONGBO KONG, HYUN-WOONG CHO, AND WOO-JIN SONG, (Member, IEEE)
Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 37673, South Korea

Corresponding author: Woo-Jin Song (wjsong@postech.ac.kr)

This research was supported by the STRL (Sensor Target Recognition Laboratory) program of Defense Acquisition Program Administration and Agency for Defense Development. This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-2012-0-00628) supervised by the IITP (Institute for Information & communications Technology Promotion). This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ''ICT Consilience Creative Progra'' (IITP-R0346-16-1007) supervised by the IITP (Institute for Information & communications Technology Promotion).

ABSTRACT In this paper, we propose dissimilarity regularization with a multistage fusion stream for a synthetic aperture radar (SAR) and infrared (IR) sensor fusion using deep learning. The multistage fusion structures are composed of multiple layers for fusing all the feature maps generated by the convolutional neural networks. The proposed structure combines feature maps of equivalent levels, ensuring that the spatial information of the corresponding levels can be utilized for fusion. Dissimilarity regularization is the sum of the normalized cross-correlation between the features generated in two different single-sensor streams. The proposed regularization is added to the conventional learning problem of a single-sensor stream, and each single-sensor stream is promoted to learn the disparate types of features for fusion. To evaluate the proposed algorithm, we compare the recognition rate of the proposed algorithm with that of the conventional fusion approaches using the SAR and IR image databases. Finally, the effects of the proposed architecture and regularization on the fusion result are analyzed.

INDEX TERMS Deep learning, target recognition, infrared, sensor fusion, synthetic aperture radar.

I. INTRODUCTION

Multimodal datasets contain information obtained from different sensors observing common phenomena [2]. Sensor fusion using these multimodal datasets can increase the recognition power compared with single-sensor datasets. To improve the discrimination ability of conventional algorithms based on single sensors, sensor fusion has been applied in various fields such as vehicle classification [3], object detection [4], action-recognition [5], skin detection [6], and others [7]–[10]. Deep learning allows computational models to be composed of multiple processing layers for learning the representations of data with multiple levels of abstraction [11]. Deep learning has been used in object detection [12]–[15], emotion recognition [16], [17], semantic segmentation [18], medical diagnosis [19], and many other domains [20]–[23]. The development of deep learning methods has been steadily widening to include multimodal domains [2]. Sensor fusion algorithms using deep learning can be classified into two fusion approaches: early and late [24]. Early fusion methods used a combined input in which the data of two modalities are integrated, and they required that all the parameters in the architecture be trained freshly using the combined input [24]. In contrast, late fusion methods only use the features extracted from pre-trained subnetworks, which have been trained for each sensor data for fusion. Moreover, in these late fusion methods the parameters of the subnetworks are fixed, and only the fusion layer that is responsible for the feature fusion is trained [24]. Late fusion methods can also use state-of-the-art pre-trained single-sensor streams. Owing to these advantages, late fusion methods have been used in various fields [13], [18], [21], [25]. However, the fusion layer of conventional late fusion methods have a simple structure consisting of only a fully connected layer. Here, the pre-trained single-sensor stream that is trained independently is exploited, without considering the relationship to the data that will be fused.

When convolutional neural networks (CNNs) are combined using the conventional late fusion methods, only vector-type information is used and a large amount of spatial information contained in the feature maps is lost. These characteristics lead to the limitation that late fusion methods cannot use all the information of the sensor data, because they only use a fully connected layer to perform the fusion. Use of conventional pre-trained networks in fusion only integrates the combined features that are suitable for single-sensor data processing. Therefore, these networks only provide a slight improvement in the fusion result. In this paper, we propose a dissimilarity regularization (DisReg) and multistage fusion stream (MFS) for resolving the demerits of late fusion methods. The MFS is a hierarchical structure that consists of multiple layers, and is able to combine single-sensor features according their levels. Conventional late fusion approaches only fuse the feature vector; however, the proposed fusion architecture can integrate all the features generated from each layer of the single-sensor streams. The features of the single-sensor streams are fused according to their level, using the convolutional layer or fully connected layer, depending on the feature type, and they are propagated to the upper layer of the MFS. In the upper layer of the MFS, the transmitted fusion features and single-sensor features having identical levels are used simultaneously as input to generate the next fusion feature. Therefore, the MFS is able to utilize the information efficiently compared with late fusion methods by using the local image features in the fusion process. DisReg is a measure of the similarity between the feature maps of two single-sensor streams; it is added to the learning problem and facilitates each single-sensor stream to learn a distinct type of feature for the fusion. The similarity is estimated using the normalized cross-correlation (NCC) during each epoch of the training, and both the CNNs are trained simultaneously. The proposed fusion algorithm is evaluated based on target recognition experiments using synthetic aperture radar (SAR) and infrared (IR) image databases. SAR and IR sensors are widely used in both military and civilian fields owing to their 24-h operation capability. A SAR image has a low resolution compared with an IR image, but it has the advantages of long-distance surveillance and the ability to operate regardless of the weather conditions. In contrast, IR images have higher resolution than SAR images and can acquire the target information during the day and night. However, IR images are affected by atmospheric conditions. Therefore, the fusion of SAR and IR sensors has tremendous potential for performance improvement. In this study, we conducted two types of experiments. First, we compared the recognition rate obtained using the proposed algorithm with various model parameters. Second, we compared the recognition rate calculated by the proposed algorithm with those generated by conventional fusion algorithms. The superiority of the proposed algorithm was demonstrated in the above experiments by the achievement of a significant improvement in the recognition rate.

II. RELATED WORK

In speech recognition, an autoencoder has been used for the fusion of a high-level representation of audio and video to generate a shared representation [13]: two sparse restricted Boltzmann machines (RBMs) are trained separately, and then a bimodal deep belief network (DBN) model is trained in a greedy layer-wise approach. Bimodal deep learning has been proposed to fuse the disparate modalities [17], [26]. Image and text data used in [26] are combined to extract a unified representation and demonstrate that the multimodal deep belief machine (DBM) can successfully learn the generated model of the joint space of the image and text inputs. For emotion recognition, a combination of the physiological signals using a bimodal deep autoencoder has been developed [17]. A shared representation that contains the correlation between the EEG signals and eye movement is generated by a bimodal deep autoencoder. The cross-weights of the sensor modalities are proposed for the gradual learning of the interactions between the modalities [27]. In [27], it has been proven theoretically that a multimodal framework including cross-weights can provide the intra-modality information of the shared representation. For achieving object detection using a late fusion network, representations of the color (red-green-blue, RGB) and depth (D) of images have been fused [12]. This method trains two CNNs, one for RGB and one for D, and then consecutively combines them using a late fusion network. A fusion method based on deep learning has been developed to detect pedestrians in RGB-D images [28]. In [29], the method adds a gating network to the structure to integrate all the decisions of a single-sensor stream. The reliability of the decision output extracted from each single-sensor stream is set based on the integrated feature vector. In [30], for action recognition, a two-stream approach has been proposed for combining spatial and temporal information; this method performs spatially varying soft-gating on CNN feature maps and amplifies the activation of the last convolutional layer of the spatial CNN by the magnitude of the optical flow to emphasize the motion information. In [18], the RGB, D, and near-IR data are combined to achieve a semantic understanding of scenes. CNNs were fused using both early fusion and late fusion methods, and compared with conventional deep learning algorithms. Several regularizations have been developed to improve fusion results via deep learning. To estimate the joint distribution of the multimodal data, in [31] a training method that minimizes the variation in the information was proposed. In [32], structure regularization was applied for training an autoencoder for the objective of selecting only a few input nodes to discourage the learning of the weak correlations between different sensory modes. A regularization for feature/class relationships [33] uses both the regularizations for training the fusion layers; each regularization explores both inter-feature and inter-class relationships and yields an improved recognition result. In this paper, we describe the details of the proposed fusion algorithm based on a target recognition task; however, the proposed algorithm can be applied in the fusion of CNNs for various purposes.

III. SINGLE-SENSOR STREAM

For ease of understanding, in this paper a single-sensor stream is referred to as a single CNN. The CNN is a deep learning structure specialized for image-type data processing using the dot product of the convolutional layer. The CNN can effectively process image-type data, but it uses fewer parameters than networks composed of fully connected layers [34]. The transition function of the l th layer of the CNN is described in (1) as follows:

$$a_l = W_l * a_{l-1} + b_l \tag{1}$$

where W_l is the weight of the l th convolutional layer, b_l is its bias, a_{l-1} is the input of the layer, and a_l is its output.

IV. DISSIMILARITY REGULARIZATION

Before describing DisReg, we first introduce the learning problem for a single-sensor stream. The typical learning problem for target recognition, meaning the cross-entropy loss, is defined in (2) as follows:

$$\operatorname{argmin}_{W_i} -\frac{1}{N}\sum_{n_i=1}^{N}[\, y_{n_i}\log(\,\hat{y}_{n_i}) + (1-y_{n_i})\log(1-\hat{y}_{n_i})] + \lambda_1\|W_i\|_2^2 \tag{2}$$

where $i \in [1, 2]$ is the index of a single-sensor stream, y_{n_i} is the label of the input image, $\hat{y}_{n_i}$ is the prediction result of i th single-sensor stream, N is the size of the mini-batch, and W_i is the weights of the i th single-sensor stream. Conventional learning problems for a single-sensor stream did not consider fusion with other single-sensor streams. The fusion of two single-sensor streams would be a difficult task if the features extracted from each single-sensor stream have a high similarity with respect to each other. For successful fusion, we propose DisReg to extract the feature map possessing the unique characteristics of the given single-sensor data. These features cannot be observed with other sensor data. By inducing this mutually complementary feature learning during training, each single-sensor stream subsequently serves complementary features to the MFS. Let the set of the feature maps of the i th single-sensor stream be as follows:

$$F_i = \left\{F_{(i,1)}, F_{(i,2)}, \cdots, F_{(i,L)}\right\} \tag{3}$$

where L is the number of convolutional layers in a single-sensor stream. Term $F_{(i,l)}$, $i \in [1, 2]$, is the feature map generated from the l th convolutional layer of the i th single-sensor stream, with the shape of $d_l @ (h_l \times w_l)$, where d_l is the number of the feature maps generated from the l th convolutional layer, and h_l and w_l are the height and width of the feature map, respectively. The similarity between F_1 and F_2 can be estimated using normalized cross-correlation (NCC). NCC has been widely used in broad image processing applications such as object recognition [35], face recognition [36], motion analysis [37], and for patch matching [38]. The NCC of two gray-scale images with the same size f(x,y), and g(x,y) can be estimated as follows:

$$\mathrm{NCC}(\mathrm{f(x,y)}, \mathrm{g(x,y)}) = \frac{1}{p}\frac{\sum_{\mathrm{x,y}}(\mathrm{f(x,y)}-\bar{\mathrm{f}})(\mathrm{g(x,y)}-\bar{\mathrm{g}})}{\sigma_{\mathrm{f}}\sigma_{\mathrm{g}}} \tag{4}$$

where p is the total number of pixels in the image, $\bar{\mathrm{f}}$, $\bar{\mathrm{g}}$ are the intensity averages, and σ_{f}, σ_{g} are the standard deviations of the intensity of f(x,y), and g(x,y), respectively. The NCC is an index varying between $+1$ and -1, depending on the degree of similarity between the two images. The NCC has a value of $+1$ for identical images, and -1 for inverted images. To measure the similarity between two single-sensor streams, we measure the similarity between the feature maps generated by the two single-sensor streams. The feature maps generated from the l th layer are composed of d_l features for each sensor stream. Let $F_{(1,l,m)}$ and $F_{(2,l,n)}$ denote the specific m th and n th feature of $F_{(1,l)}$, and $F_{(2,l)}$, then the similarity between $F_{(1,l,m)}$, and $F_{(2,l,n)}$ is estimated in (5) as follows:

$$\begin{aligned}&\mathrm{NCC}(F_{(1,l,m)}, F_{(2,l,n)})\\ &\quad= \frac{1}{h_l \times w_l}\\ &\quad\times\frac{\sum_{h,w}(F_{(1,l,m)}(h,w)-\bar{F}_{(1,l,m)})(F_{(2,l,n)}(h,w)-\bar{F}_{(2,l,n)})}{\sigma_{F_{(1,l,m)}}\sigma_{F_{(2,l,n)}}}\end{aligned} \tag{5}$$

Dissimilarity regularization ($\mathcal{L}_{DisReg}$) is defined as the summation of the NCC values estimated for each layer of the single-sensor stream as follows:

$$\mathcal{L}_{DisReg}(F_1, F_2) = \sum_{l=1}^{L}\sum_{m=1}^{d_l}\sum_{n=1}^{d_l}\mathrm{NCC}\left(F_{(1,l,m)}, F_{(2,l,n)}\right) \tag{6}$$

where L is the number of convolutional layers of a single-sensor stream, and d_l is the number of feature maps generated from the l th layer of a single-sensor stream. We define the new learning problems for the single-sensor stream using the amended typical learning equation of $\mathcal{L}_{DisReg}$ as follows:

$$\operatorname{argmin}_{W_i} -\frac{1}{N}\sum_{n_i=1}^{N}[\, y_{n_i}\log(\,\hat{y}_{n_i}) + (1-y_{n_i})\log(1-\hat{y}_{n_i})] + \lambda_1\|W_i\|_2^2 + \lambda_2\mathcal{L}_{DisReg}(F_1, F_2) \tag{7}$$

DisReg added in equation (7) causes two single-sensor streams to learn the patterns that are distinct from each other before fusing them. This learning problem is designed to assist each single-sensor stream to learn features that are discriminative and suitable for complementary fusion processes.

V. MULTISTAGE FUSION STREAM

Recently, deep learning structures such as DenseNet [39], ResNet [40], and single shot multibox detector (SSD) [41] have been proposed to utilize the features from several different layers and achieve improved results. These methods

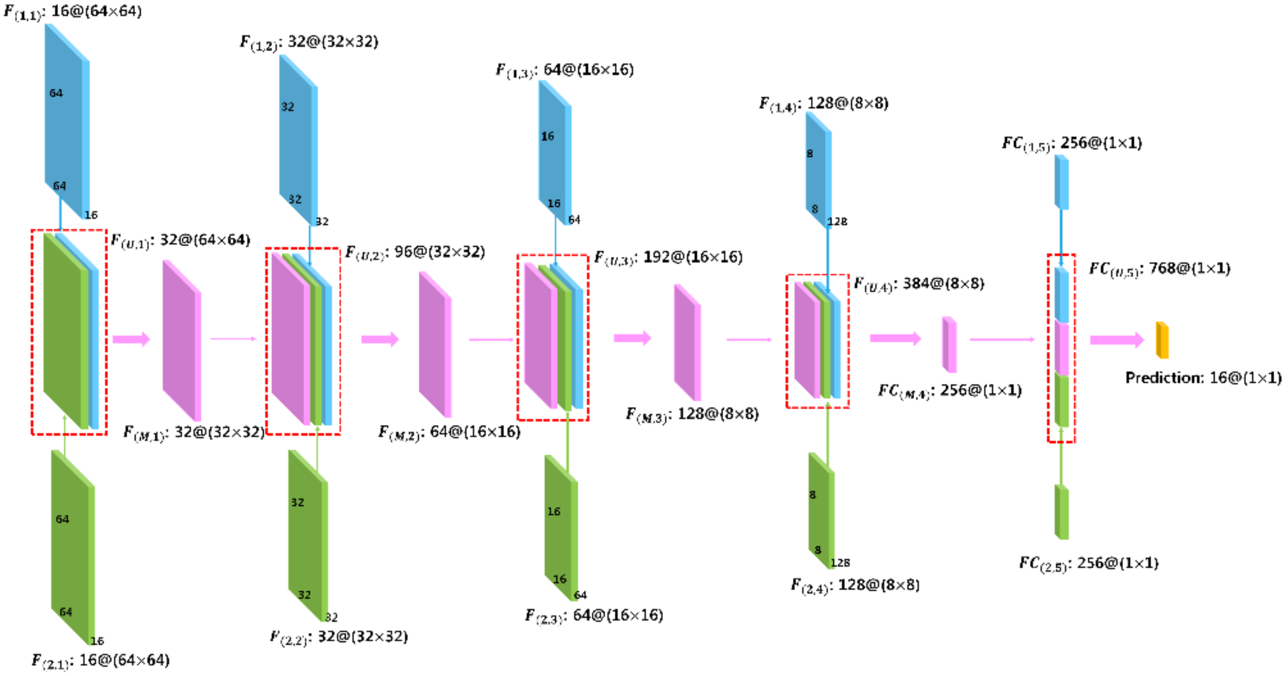


FIGURE 1. The structure of the multistage fusion stream architecture used in this paper. Pre-trained single-sensor streams transimit their feature map to multistage fusion stream according to level of features. Each layer of multistage fusion stream exploits integrated feature, $F_{(U,l)}$ that contain only features of equivalent levels to generate the new fusion feature map. The feature map, $F_{(M,l)}$ is used as an input to the next layer of multistage fusion stream with single-sensor feature. $FC_{(U,5)}$ is feature vector that represents the information of all stage of not only fusion stream, but also single-sensor streams, and so has good discrimination power. Bold arrow: transition function, Narrow arrow: transmission, Red box: integrated feature $F_{(U,i)}$, Blue color: features of 1 st single-sensor stream, Green color: features of 2 nd single-sensor stream, Pink color: features of fusion stream.

combine the features of different levels at a certain layer to utilize the various scales and resolution information of the features. However, conventional fusion algorithms based on deep learning focus on exploiting the late fusion structure rather than using various local image features. Motivated by these results, we proposed an MFS that can use the features generated at all the stages of the single-sensor streams to take advantage of all the different levels of the features. To exploit the spatial information in the feature map effectively, MFS uses a convolutional layer to fuse the feature maps of the single-sensor streams. MFS generates the fusion feature map at each level and propagates it to the upper layer via a hierarchical structure. We evaluated the recognition rate of the proposed fusion method and compared it with that of conventional fusion methods. First, we describe the details of MFS.

A. MULTISTAGE FUSION STREAM ARCHITECTURE

MFS is a fusion structure that allows the equivalent levels of the feature maps extracted from a single-sensor stream to be fused to generate the fusion feature (Fig. 1). The proposed fusion architecture generates the fusion feature in each layer of the MFS using the single-sensor features transmitted from each single-sensor stream and the fusion feature generated from the previous layer. The generated fusion features are exploited with the equivalent levels of the single-sensor features as the input to the upper layer. The features of the various levels that are extracted from the single-sensor streams are combined to yield the fusion feature through the convolutional layer or a fully connected layer depending on their feature type. Each fusion information is propagated to a higher-level layer by utilizing the above hierarchical structure. The top fusion feature containing the new local image feature (which cannot be acquired by using only a single-sensor stream), is concatenated with the feature vector of each single-sensor stream and used for the recognition task. To generate the fusion feature in each layer of the MFS, we integrated the features of the equivalent level into one feature. Let $F_{(U,l)}$ be the integrated feature map that is used as the input to the l th layer of the fusion stream. It can be defined as follows:

$$F_{(U,l)} = \begin{cases} [F_{(1,l)}, F_{(2,l)}] & \text{where } l = 1 \\ [F_{(1,l)}, F_{(2,l)}, F_{(M,l-1)}] & \text{where } L \geq l \geq 2 \end{cases} \tag{8}$$

Feature maps $F_{(1,l)}$, and $F_{(2,l)}$ have a shape of $d_l@(h_l \times w_l)$, where d_l is the number of features of $F_{(i,l)}$. An integrated feature is formed by accumulating the equivalent levels of the feature in the dimension of the number of features, d_l. The integrated feature has identical feature size $(h_l \times w_l)$, with an increased number of features $3 \times d_l$. To propagate the previous fusion information and exploit the single-sensor feature, the l th layer of the fusion stream uses all these features by integrating them into feature map $F_{(U,l)}$.

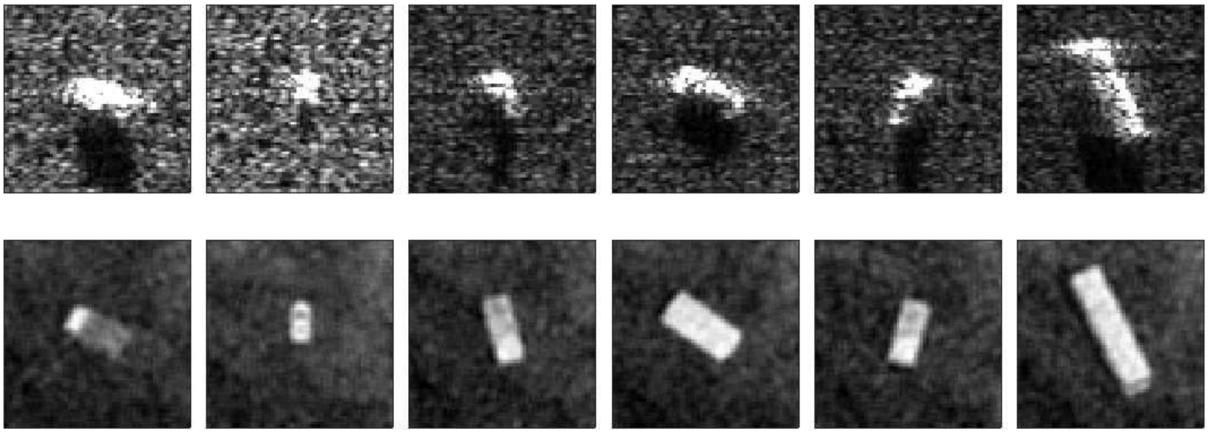


FIGURE 2. SAR/IR target chip sample, First row: SAR images, Second row: IR images.

Because no fusion feature map has the same level as the features of single-sensor streams $F_{(1,l)}$, and $F_{(2,l)}$, only the features of the single-sensor streams are used to form $F_{(U,1)}$. To match the level of the fusion feature with a feature of the single-sensor stream, the l^{th} layer of the proposed architecture has the same layer configuration as the $(l+1)^{\text{th}}$ layer of the single-sensor stream. The proposed scheme is designed to exploit a large amount of information compared to conventional methods, by using the features of various levels (including the image local information), which has not been used in late fusion methods.

B. TRAINING OF MULTISTAGE FUSION STREAM

After all the single-sensor streams are trained, the MFS is trained using the feature maps generated from the single-sensor streams. The mature feature maps extracted from each single-sensor stream are integrated by (8) and used as inputs for each stage of the MFS according to its level. The MFS is trained using the standard cross-entropy loss,

$$\operatorname{argmin}_{W_F} -\frac{1}{N}\sum_{n_F=1}^{N}[\,y_{n_F}\log(\,\hat{y}_{n_F}) + (1-y_{n_F})\log(1-\hat{y}_{n_F})] + \lambda_1\|W_F\|_2^2 \quad (9)$$

where W_F is the weight of the fusion stream, y_{n_F} is the image label, and $\hat{y}_{n_F}$ is the prediction of the MFS fusion stream. During the training of the MFS, all the trainable parameters in all the single-sensor streams are fixed; only the parameters of the MFS are trained. In summary, the learning process of the proposed fusion algorithm is divided into two stages: single sensor stream training and fusion stream training. Single-sensor streams are trained to learn complementary features using a learning problem, as expressed in (7). Once the training of the single-sensor streams is completed, they provide the features for training the MFS according to the level of the feature. The features of equivalent levels are integrated using equation (8), and the parameters of the MFS are updated using the cross-entropy loss.

TABLE 1. SAR/IR image database.

Target classes	SAR Depression angle 10°	15°	20°	25°	SAR Total	IR Depression angle 65°	70°	75°	80°	IR Total
AMX10	72	72	72	72	288	72	72	72	72	288
AMX10RC	72	72	72	72	288	72	72	72	72	288
Audi	72	72	72	72	288	72	72	72	72	288
BMP3	72	72	72	72	288	72	72	72	72	288
Bus	72	72	72	72	288	72	72	72	72	288
Clio	72	72	72	72	288	72	72	72	72	288
Fire truck	72	72	72	72	288	72	72	72	72	288
Ford transit	72	72	72	72	288	72	72	72	72	288
Jeep	72	72	72	72	288	72	72	72	72	288
Leclerc	72	72	72	72	288	72	72	72	72	288
Oil tank	72	72	72	72	288	72	72	72	72	288
Radar camo	72	72	72	72	288	72	72	72	72	288
Sa19inch camo	72	72	72	72	288	72	72	72	72	288
T72	72	72	72	72	288	72	72	72	72	288
TMM	72	72	72	72	288	72	72	72	72	288
VABOBS	72	72	72	72	288	72	72	72	72	288

VI. EXPERIMENTS

A. DATABASE

In this study, we evaluated the target-recognition accuracy of the proposed algorithm using an SAR/IR image database (Fig. 2). The SAR/IR database used in the experiment consists of images of 16 different target classes. This database contains a total 4,608 images per sensor, and the size of each image is 64×64 (Table 1). The SAR image database is collected at 10°, 15°, 20°, and 25° depression angles. To cover the full aspect angle, the target image is rotated in 5° intervals of the aspect angle from 0° to 355°, resulting in 72 images per depression angle. The IR database is collected at 65°, 70°, 75°, and 80° depression angles, and the other conditions are the same as for the SAR image database. The SAR/IR images are generated by using the SE-RAY-SAR and SE-RAY-IR simulators, which can generate very realistic images via the ray-tracing technique [42]–[46]. For training and evaluation, we separated the SAR/IR database depending on the depression angle (Table 2). The SAR image at 25° depression angle and IR image at 80° depression angle

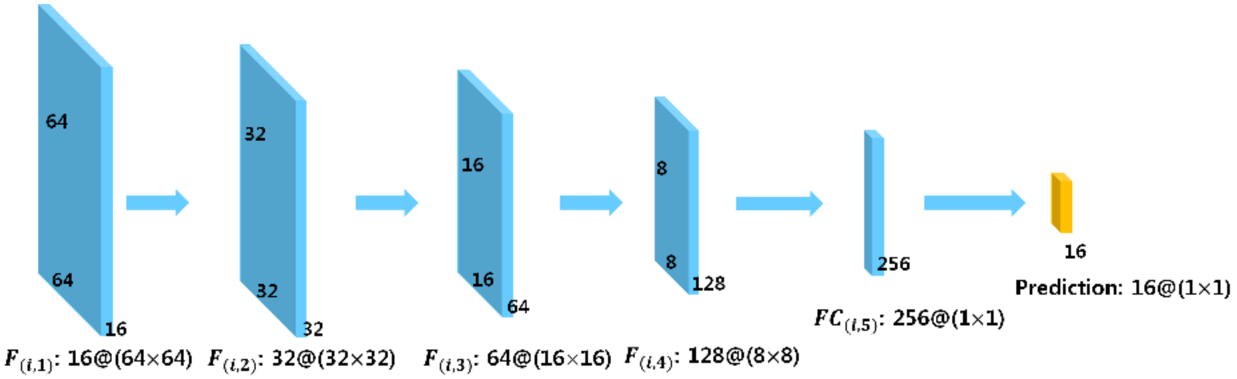


FIGURE 3. The structure of single-sensor CNN, and early fusion architecture used in this paper. First to fourth stage of single-sensor streams generate the feature map $F_{(i,l)}$, where i is index of single-sensor stream, and l is the level of the stage. The feature vector $FC_{(i,l)}$ is generated using fully connected layer. The softmax classifier is used for prediction. Bold arrow: transition function.

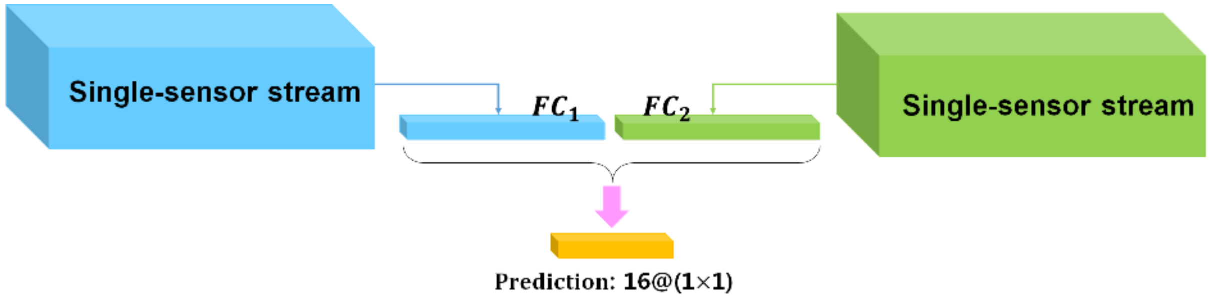


FIGURE 4. The structure of the late fusion used in this paper. Each single-sensor stream generates the feature vector FC_1, FC_2, and then concatenated to form one feature vector. Late fusion method exploits last concatenated feature vector for prediction. Bold arrow: transition function, Narrow arrow: transmission.

TABLE 2. Database setup for experiment.

	SAR		IR	
Data split	Training	Test	Training	Test
Depression angle	10°, 15°, 20°	25°	65°, 70°, 75°	80°
Total	3456	1152	3456	1162

are used for the test, and the remaining images are used for training each single-sensor stream. We augment the training data by vertical and horizontal flipping to increase the size of the training database, and thus, 10,368 images per sensor are used for training.

B. EXPERIMENTAL SETTING

To analyze the effect of DisReg on a single-sensor stream, we trained two single-sensor streams with and without DisReg for each sensor, and then evaluated the recognition rates of these streams. In addition, we trained the MFS with and without DisReg and examined the target-recognition accuracy. For the MFS with DisReg, we trained the single-sensor streams with DisReg using (7) and used them to train MFS. The MFS without DisReg used the single-sensor streams that were trained using (2). For comparison with conventional algorithms, the following approaches were implemented, and a target recognition experiment was conducted.

- **Early fusion**
  The early fusion architecture combines the information of two image-type data at the pixel-level and uses a combined image as the input. The architecture is identical to that of the single-sensor stream used in this study (Fig. 3). The early fusion architecture is trained using the conventional learning method in (2) to update all the parameters in the architecture.
- **Late fusion**
  Here, the fusion layer fuses the feature vectors generated by each single-sensor stream (Fig. 4). Each single-sensor stream is trained independently using (2). The learning method used for training the fusion layers is composed of a cross-entropy error term and the typical L2 regularization term. Weights and biases of the single-sensor streams are maintained during the training of the fusion layers.
- **Autoencoder-based fusion (AE-based fusion)**
  The fully connected layer for fusion in the late fusion is trained by the encoder-decoder scheme [47] to generate a shared representation from the feature vector of each single-sensor stream. Each single-sensor stream is trained using (2), identical to the late fusion case, and the parameters of the single-sensor streams are fixed during the training of the fusion layer. After the training

of the fusion layer is completed, a fine-tuning process is applied to the fusion network for the recognition task.

- **Convolutional autoencdoer-based fusion (CAE-based fusion)**
  The encoder-decoder scheme of a convolutional autoencoder (CAE) exploits the convolutional layer instead of the fully connected layer. We applied CAE-based fusion [48] to the MFS to impose the inner relationship between two feature maps on the corresponding fusion feature map. We used a greedy layer-wise approach to train each fusion layer; this is the same training method as used for AE. After the fusion layer was trained, we removed the decoder scheme and fine-tuned the fusion stream for the given task.
- **Fusion using regularizations for feature/class relationship [33]**
  In [33], regularization of the feature relationships and class relationships was proposed to train the fusion layer in the late fusion structure and improve the categorization result. In this study, we used both regularizations and applied them to the late fusion layer. As in [33], we only updated the parameters of the fusion layer using the cost function to which the features and class relationship regularization were added.

In this experiment, we used an architecture with a low degree of freedom to prevent overfitting. The single-sensor streams consist of four stages of convolutional layers and one fully connected layer for feature vector generation. The softmax classifiers are applied for the final prediction. The filter size of the convolutional layers from the first to the third stages is 5×5, and only the last convolutional layer uses a filter size of 3×3. The last convolutional stage generates a 128@(8×8) feature map, and the fully connected layer converts the last feature map to a feature vector size of 256. For maximum pooling, we set both the window sizes and stride value as 2×2. ReLU [49] and batch normalization layers [50] were installed for nonlinearity and fast convergence of the training. The weights and biases were initialized using the Xavier initialization [51] and random uniform, respectively. We used the Adam optimizer [52] for backpropagation, and $\beta_1 = 0.9$, $\beta_2 = 0.999$. For batch normalization [50], we set the decay for the moving average as 0.9. The mini-batch size of all the tested streams was 32. In addition to the previous research [1], all the structures were trained with the scale parameters of the batch normalization, and the learning parameter was set to decrease in intervals of 500 epochs by 0.9 times for learning stability. In the experiments, we set the weight of L2 regularization to $\lambda_1 = 10^{-4}$. For fusion using regularization for the feature/class relationship, we set the weights of both the class relationship regularization and feature regularization [33] as 10^{-6}. Terms λ_1 and λ_2 are factors that determine the proportion of L2 regularization, and DisReg in the cost function. For the proposed algorithm, we set $\lambda_1 = 10^{-4}$ and $\lambda_2 = 10^{-8}$. We selected a small value of λ_2 owing to the large scale of DisReg. All the hyperparameters were determined experimentally, which showed fast convergence and stability of the training of all the streams we tested.

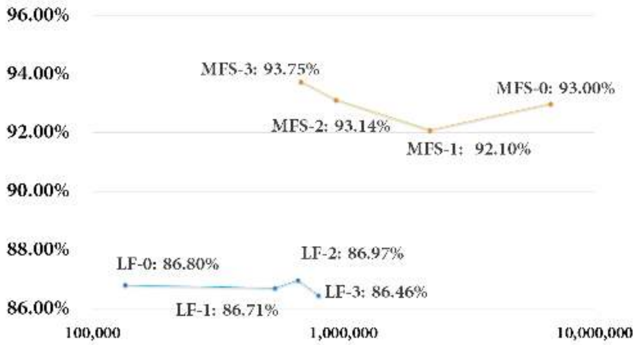


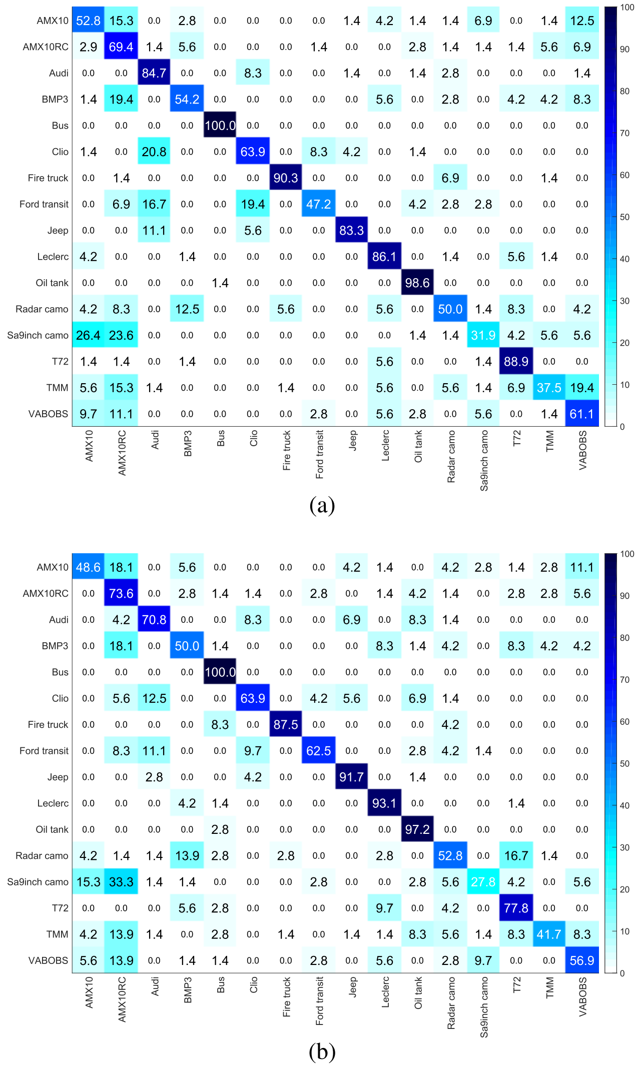
FIGURE 5. Recognition rates of late fusion methods and multistage fusion streams depending on the number of parameters.

C. RECOGNITION RATE FOR VARIOUS MODEL PARAMETERS

The proposed fusion structure may require numerous additional parameters; this can increase the total number of parameters. In this part of the study, we measured the recognition rate of the proposed fusion structure and late fusion method as a function of the total number of parameters to analyze the necessity and efficiency of the additional parameters in the MFS. The MFS structures were reduced by substituting the fully connected layer into the convolutional layer for feature extraction, whereas the late fusion structure was expanded with an additional fully connected layer (Table 3). Late fusion only exploits the feature vector. Therefore, it is inevitable that this method suffers from information loss. However, even if the structure is expanded, there is little effect on the recognition rate as a function of the number of parameters (Fig. 5). In the MFS, the number of variables used is significantly reduced by replacing the fully connected layer with the convolutional layers, but the recognition rates are maintained above 92%. Among the tested structures, MFS-3 exhibits the highest recognition rate of 93.75% using 675,328 parameters. In this experiment, the recognition rates of the MFS methods are higher than those of the late fusion methods. The latter are approximately 86%, but even the lowest recognition rate among the four different proposed fusion structures achieves 92%. All the MFS series result in an improved recognition rate compared with the late fusion series, but this improvement is not simply caused by the increase in the number of parameters. It can be seen that the use of various levels of features for fusion rather than the number of parameters helps in improving the recognition rate when comparing MFS-3 and LFS-3. The proposed fusion algorithm acquires the local features of the images and prevents information loss by fusing the features generated by the various levels of the single-sensor stream. Because it is impossible for late fusion to exploit the local feature information, there is no noticeable improvement, even if the structures are enlarged.

TABLE 3. The total number of parameters for fusion structures. LF: Late Fusion, MFS: Multistage Fusion Streams.

Fusion architecture	Layer configurations	The number of parameters
LF-0	(512 x 512) / (256 x 16)	135,168
LF-1	(512 x 512) / (512 x 256) / (256 x 16)	532,480
LF-2	(512 x 512) / (512 x 512) / (512 x 16)	659,456
LF-3	(512 x 512) / (512 x 512) / (512 x 512) / (512 x 16)	794,624
MFS-0	32@(5x5) / 64@(5x5) / 128@(5x5) / ((128x8x8) x 256) / (768 x 16)	6,708,736
MFS-1	32@(5x5) / 64@(5x5) / 128@(3x3) / 128@(3x3) / (2048 x 256) / (768 x 16)	2,211,328
MFS-2	32@(5x5) / 64@(5x5) / 128@(3x3) / 128@(3x3) / 64@(3x3) / (768 x 16)	933,376
MFS-3	32@(5x5) / 64@(5x5) / 128@(3x3) / 64@(3x3) / 64@(3x3) / (768 x 16)	675,328



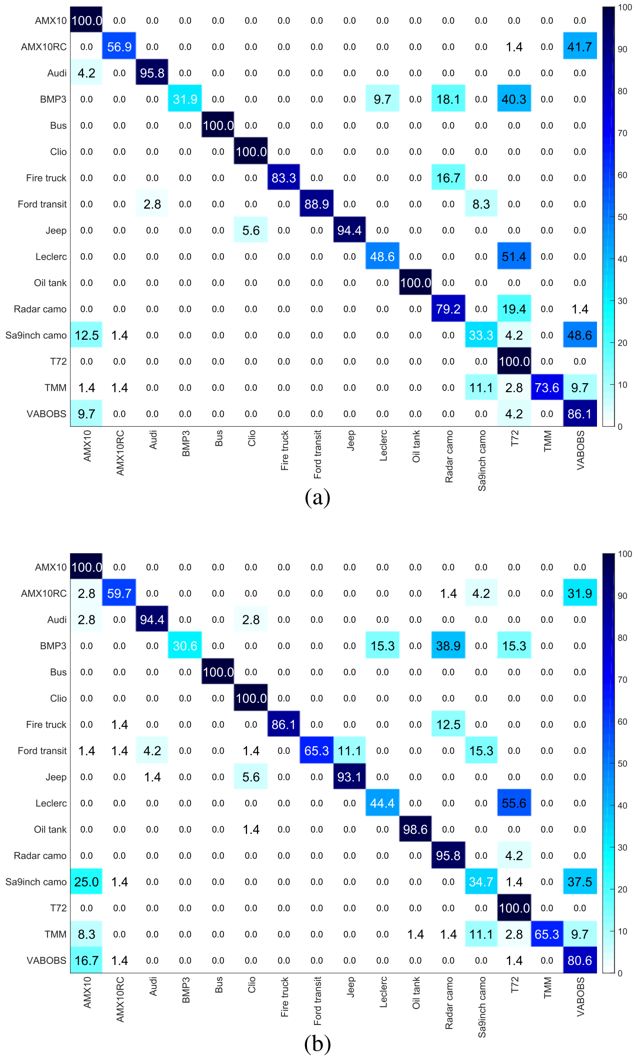
	AMX10	AMX10RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9inch camo	T72	TMM	VABOBS
AMX10	52.8	15.3	0.0	2.8	0.0	0.0	0.0	0.0	1.4	4.2	1.4	1.4	6.9	0.0	1.4	12.5
AMX10RC	2.9	69.4	1.4	5.6	0.0	0.0	0.0	1.4	0.0	0.0	2.8	1.4	1.4	1.4	5.6	6.9
Audi	0.0	0.0	84.7	0.0	0.0	8.3	0.0	0.0	1.4	0.0	1.4	2.8	0.0	0.0	0.0	1.4
BMP3	1.4	19.4	0.0	54.2	0.0	0.0	0.0	0.0	0.0	5.6	0.0	2.8	0.0	4.2	4.2	8.3
Bus	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Clio	1.4	0.0	20.8	0.0	0.0	63.9	0.0	8.3	4.2	0.0	1.4	0.0	0.0	0.0	0.0	0.0
Fire truck	0.0	1.4	0.0	0.0	0.0	0.0	90.3	0.0	0.0	0.0	0.0	6.9	0.0	0.0	1.4	0.0
Ford transit	0.0	6.9	16.7	0.0	0.0	19.4	0.0	47.2	0.0	0.0	4.2	2.8	2.8	0.0	0.0	0.0
Jeep	0.0	0.0	11.1	0.0	0.0	5.6	0.0	0.0	83.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Leclerc	4.2	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	86.1	0.0	1.4	0.0	5.6	1.4	0.0
Oil tank	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	98.6	0.0	0.0	0.0	0.0	0.0
Radar camo	4.2	8.3	0.0	12.5	0.0	0.0	5.6	0.0	0.0	5.6	0.0	50.0	1.4	8.3	0.0	4.2
Sa9inch camo	26.4	23.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	1.4	31.9	4.2	5.6	5.6
T72	1.4	1.4	0.0	1.4	0.0	0.0	0.0	0.0	0.0	5.6	0.0	0.0	1.4	88.9	0.0	0.0
TMM	5.6	15.3	1.4	0.0	0.0	0.0	1.4	0.0	0.0	5.6	0.0	5.6	1.4	6.9	37.5	19.4
VABOBS	9.7	11.1	0.0	0.0	0.0	0.0	0.0	2.8	0.0	5.6	2.8	0.0	5.6	0.0	1.4	61.1

(a)

	AMX10	AMX10RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9inch camo	T72	TMM	VABOBS
AMX10	48.6	18.1	0.0	5.6	0.0	0.0	0.0	0.0	4.2	1.4	0.0	4.2	2.8	1.4	2.8	11.1
AMX10RC	0.0	73.6	0.0	2.8	1.4	1.4	0.0	2.8	0.0	1.4	4.2	1.4	0.0	2.8	2.8	5.6
Audi	0.0	4.2	70.8	0.0	0.0	8.3	0.0	0.0	6.9	0.0	8.3	1.4	0.0	0.0	0.0	0.0
BMP3	0.0	18.1	0.0	50.0	1.4	0.0	0.0	0.0	0.0	8.3	1.4	4.2	0.0	8.3	4.2	4.2
Bus	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Clio	0.0	5.6	12.5	0.0	0.0	63.9	0.0	4.2	5.6	0.0	6.9	1.4	0.0	0.0	0.0	0.0
Fire truck	0.0	0.0	0.0	0.0	8.3	0.0	87.5	0.0	0.0	0.0	0.0	4.2	0.0	0.0	0.0	0.0
Ford transit	0.0	8.3	11.1	0.0	0.0	9.7	0.0	62.5	0.0	0.0	2.8	4.2	1.4	0.0	0.0	0.0
Jeep	0.0	0.0	2.8	0.0	0.0	4.2	0.0	0.0	91.7	0.0	1.4	0.0	0.0	0.0	0.0	0.0
Leclerc	0.0	0.0	0.0	4.2	1.4	0.0	0.0	0.0	0.0	93.1	0.0	0.0	0.0	1.4	0.0	0.0
Oil tank	0.0	0.0	0.0	0.0	2.8	0.0	0.0	0.0	0.0	0.0	97.2	0.0	0.0	0.0	0.0	0.0
Radar camo	4.2	1.4	1.4	13.9	2.8	0.0	2.8	0.0	0.0	2.8	0.0	52.8	0.0	16.7	1.4	0.0
Sa9inch camo	15.3	33.3	1.4	1.4	0.0	0.0	0.0	2.8	0.0	0.0	2.8	5.6	27.8	4.2	0.0	5.6
T72	0.0	0.0	0.0	5.6	2.8	0.0	0.0	0.0	0.0	9.7	0.0	4.2	0.0	77.8	0.0	0.0
TMM	4.2	13.9	1.4	0.0	2.8	0.0	1.4	0.0	1.4	1.4	8.3	5.6	1.4	8.3	41.7	8.3
VABOBS	5.6	13.9	0.0	1.4	1.4	0.0	0.0	2.8	0.0	5.6	0.0	2.8	9.7	0.0	0.0	56.9

(b)

FIGURE 6. Recognition rate of SAR single-sensor stream (a) trained using basic learning method, and (b) trained using the DisReg-based method.



	AMX10	AMX10RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9inch camo	T72	TMM	VABOBS
AMX10	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AMX10RC	0.0	56.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	0.0	41.7
Audi	4.2	0.0	95.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BMP3	0.0	0.0	0.0	31.9	0.0	0.0	0.0	0.0	0.0	9.7	0.0	18.1	0.0	40.3	0.0	0.0
Bus	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Clio	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fire truck	0.0	0.0	0.0	0.0	0.0	0.0	83.3	0.0	0.0	0.0	0.0	16.7	0.0	0.0	0.0	0.0
Ford transit	0.0	0.0	2.8	0.0	0.0	0.0	0.0	88.9	0.0	0.0	0.0	0.0	8.3	0.0	0.0	0.0
Jeep	0.0	0.0	0.0	0.0	0.0	5.6	0.0	0.0	94.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Leclerc	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	48.6	0.0	0.0	0.0	51.4	0.0	0.0
Oil tank	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
Radar camo	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	79.2	0.0	19.4	0.0	1.4
Sa9inch camo	12.5	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	33.3	4.2	0.0	48.6
T72	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0
TMM	1.4	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	11.1	2.8	73.6	9.7
VABOBS	9.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.2	0.0	86.1

(a)

	AMX10	AMX10RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9inch camo	T72	TMM	VABOBS
AMX10	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AMX10RC	2.8	59.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	4.2	0.0	0.0	31.9
Audi	2.8	0.0	94.4	0.0	0.0	2.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BMP3	0.0	0.0	0.0	30.6	0.0	0.0	0.0	0.0	0.0	15.3	0.0	38.9	0.0	15.3	0.0	0.0
Bus	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Clio	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fire truck	0.0	1.4	0.0	0.0	0.0	0.0	86.1	0.0	0.0	0.0	0.0	12.5	0.0	0.0	0.0	0.0
Ford transit	1.4	1.4	4.2	0.0	0.0	1.4	0.0	65.3	11.1	0.0	0.0	0.0	15.3	0.0	0.0	0.0
Jeep	0.0	0.0	1.4	0.0	0.0	5.6	0.0	0.0	93.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Leclerc	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	44.4	0.0	0.0	0.0	55.6	0.0	0.0
Oil tank	0.0	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	98.6	0.0	0.0	0.0	0.0	0.0
Radar camo	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	95.8	0.0	4.2	0.0	0.0
Sa9inch camo	25.0	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	34.7	1.4	0.0	37.5
T72	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0
TMM	8.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	1.4	11.1	2.8	65.3	9.7
VABOBS	16.7	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	0.0	80.6

(b)

FIGURE 7. Recognition rate of IR single sensor stream (a) trained using basic learning method, and (b) trained using the DisReg-based method.

D. TARGET RECOGNITION

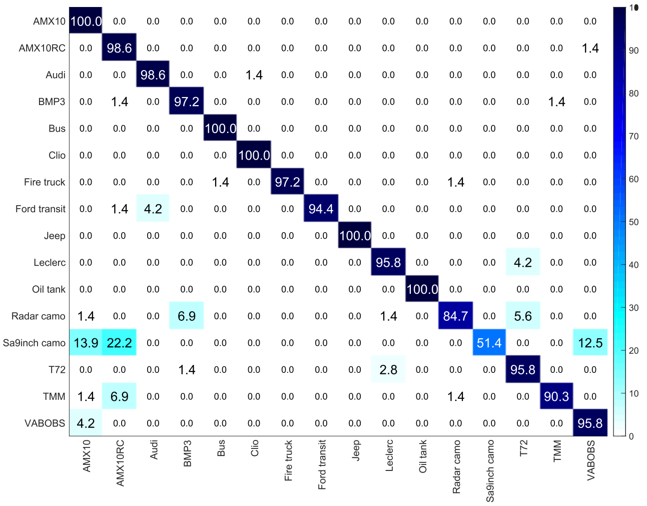
First, the recognition results of the single-sensor streams were compared (Table 4). The single-sensor stream without DisReg shows a recognition rate of 79.5% for the SAR sensor (Fig. 6a) and 68.8% for the IR sensor (Fig. 7a). In contrast, the single-sensor streams trained using DisReg achieve a recognition rate of 78.0% for the SAR sensor (Fig. 6b) and 68.4% for the IR sensor (Fig. 7b). The results indicate that the single-sensor stream using DisReg undergoes some degradation in accuracy. The MFS with DisReg exhibits a higher recognition rate than the MFS without DisReg (Table 5). Contrary to the result of single-sensor streams, the use of DisReg increases the recognition rate of

TABLE 4. Recognition rate[%] of single-sensor streams.

	AMX10	AMX10 RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9Inch Camo	T72	TMM	VABOBS	**Average**
IR	100.0	56.9	95.8	31.9	100.0	100.0	83.3	88.9	94.4	48.6	100.0	79.2	33.3	100.0	73.6	86.1	79.5
SAR	52.8	69.4	84.7	54.2	100.0	63.9	90.3	47.2	83.3	86.1	98.6	50.0	31.9	88.9	37.5	61.1	68.8
IR with DisReg	100.0	59.7	94.4	30.5	100.0	100.0	86.1	65.2	93.0	44.4	98.6	95.8	34.7	100.0	65.2	80.5	78.0
SAR with DisReg	48.6	73.6	70.8	50	100.0	63.8	87.5	62.5	91.6	93.0	97.2	52.7	27.7	77.7	41.6	56.9	68.4

TABLE 5. Recognition rate[%] of fusion streams.

	AMX10	AMX10 RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9Inch Camo	T72	TMM	VABOBS	**Average**
Early fusion	69.4	70.8	83.3	47.2	98.6	70.8	95.8	76.3	91.6	86.1	98.6	62.5	31.9	90.2	41.6	54.1	73.0
Late fusion	98.6	72.2	98.6	56.9	100.0	100.0	95.8	91.6	98.6	76.3	100.0	79.1	54.1	100.0	73.6	93.0	86.8
AE-based fusion	100.0	75.0	98.6	48.6	100.0	100.0	86.1	95.8	100.0	84.7	100.0	84.7	54.1	98.6	81.9	90.2	87.4
CAE-based fusion	95.8	79.1	98.6	52.7	100.0	100.0	97.2	94.4	98.6	77.7	100.0	79.1	54.1	100.0	77.7	94.4	87.5
[33] fusion	94.4	70.8	98.6	68.0	100.0	100.0	98.6	88.8	100.0	84.7	100.0	86.1	52.7	98.6	86.1	97.2	89.0
Multistage without DisReg	97.2	80.5	98.6	70.8	100.0	100.0	98.6	95.8	100	73.6	100.0	81.9	65.2	98.6	76.3	94.4	89.6
Multistage with DisReg	100.0	98.6	98.6	97.2	100.0	100.0	97.2	94.4	100.0	95.8	100.0	84.7	51.3	95.8	90.2	95.8	93.7



	AMX10	AMX10RC	Audi	BMP3	Bus	Clio	Fire truck	Ford transit	Jeep	Leclerc	Oil tank	Radar camo	Sa9inch camo	T72	TMM	VABOBS
AMX10	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AMX10RC	0.0	98.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4
Audi	0.0	0.0	98.6	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BMP3	0.0	1.4	0.0	97.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	0.0
Bus	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Clio	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fire truck	0.0	0.0	0.0	0.0	1.4	0.0	97.2	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0
Ford transit	0.0	1.4	4.2	0.0	0.0	0.0	0.0	94.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Jeep	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Leclerc	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	95.8	0.0	0.0	0.0	4.2	0.0	0.0
Oil tank	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
Radar camo	1.4	0.0	0.0	6.9	0.0	0.0	0.0	0.0	0.0	1.4	0.0	84.7	0.0	5.6	0.0	0.0
Sa9inch camo	13.9	22.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	51.4	0.0	0.0	12.5
T72	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	2.8	0.0	0.0	0.0	95.8	0.0	0.0
TMM	1.4	6.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	0.0	0.0	90.3	0.0
VABOBS	4.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	95.8

FIGURE 8. Recognition results of the proposed fusion method: 93.75%.

the fusion streams. We also compared the recognition rate of the proposed fusion algorithm with conventional fusion algorithms (Table 5, Figs. 8 and 9). The early fusion method shows the lowest recognition rate (73.0%) among the fusion algorithms (Fig. 9a) The result of early fusion is lower than that of the IR single stream, implying that early fusion fails to achieve an improvement in the accuracy. During the learning of all the parameters, it fails to learn the discriminative features from the combined input image. The late fusion results achieve 86.8% recognition rate (Fig.9b). The late fusion methods use pre-trained single-sensor streams. Therefore, it reaches a higher recognition rate than the early fusion methods by exploiting the mature features extracted from each single-sensor stream for fusion. AE-based fusion achieves an 87.4% recognition rate (Fig. 9c). By learning the correlation between the features generated from each single-sensor stream, AE-based fusion yields a higher recognition rate than late fusion, which simply combines the features. CAE-based fusion attains recognition rate of 87.5%, which is slightly higher than that of AE-based fusion (Fig.9d). Further, it learns the correlation between the feature maps by using a convolutional layer. These feature maps containing spatial information improve the recognition rate by performing the fusion using more information than used in AE-based fusion. Among the conventional fusion methods, the method that uses feature/class relationship regularization [33] achieves the highest recognition rate of 89.0% (Fig.9e). The encoder-decoder scheme uses a construction process to learn the shared representation that represents the correlation indirectly. However, the fusion method that uses relationship regularization directly learns the correlation between the features and classes, and thus [33], shows an improved discrimination result. The proposed algorithm (MFS with DisReg) achieves a higher recognition result than conventional fusion algorithms (Fig. 8). From the learning process of a single-sensor stream, DisReg assigns complementary attributes to the features of each single-sensor stream, and the MFS fuses all the features generated from the single-sensor streams at each stage. The proposed scheme can use all the information of single-sensor streams for fusion, and thereby achieve the highest recognition rate.

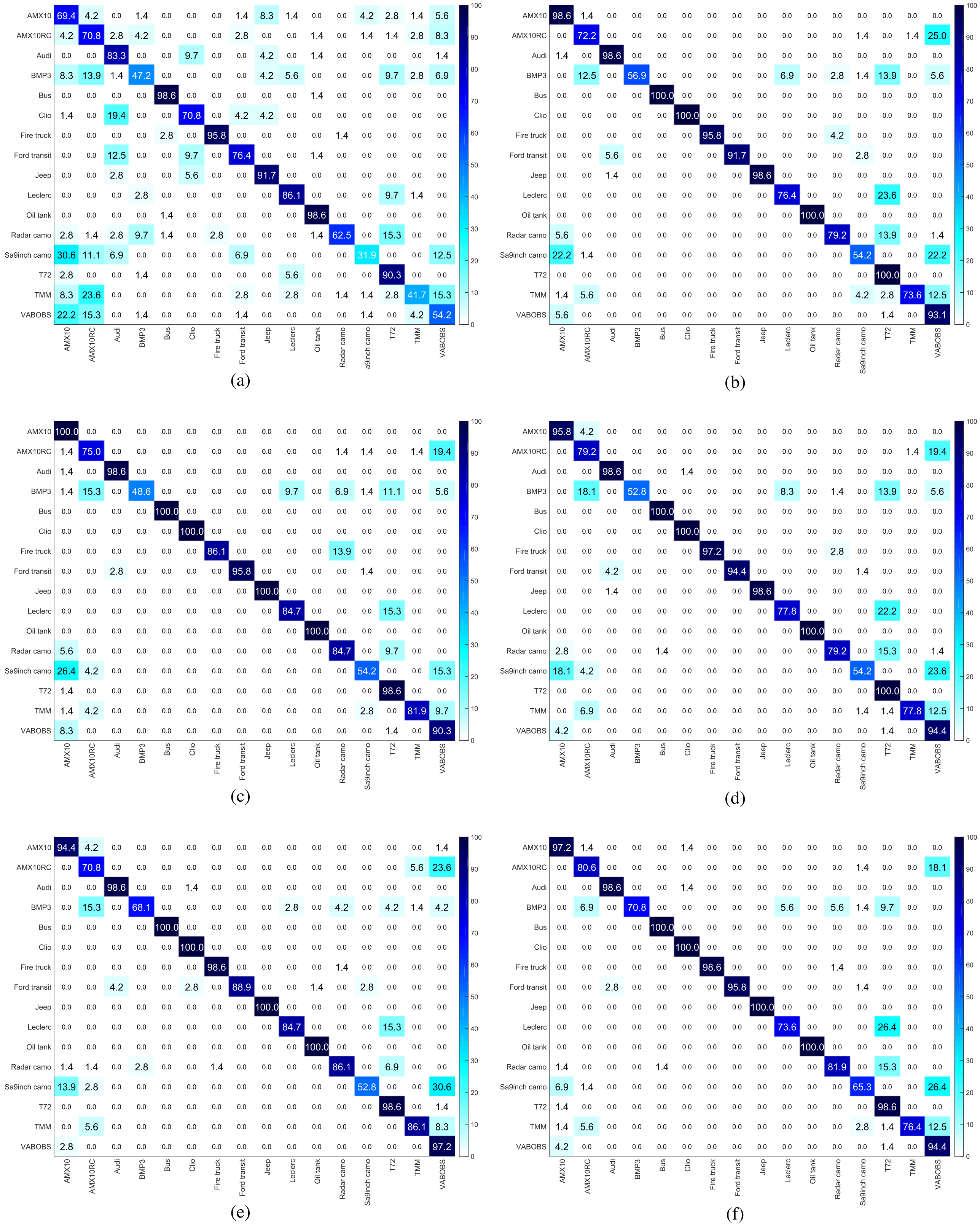


FIGURE 9. Recognition rate of the fusion approaches. (a) Recognition result of early fusion: 73.09%. (b) Recognition result of late fusion: 86.80%. (c) Recognition result of AE-based fusion: 87.41%. (d) Recognition result of CAE-based fusion: 87.50%. (e) Recognition result of [33] fusion: 89.06%. (f) Recognition result of MFS without DisReg: 89.49%.

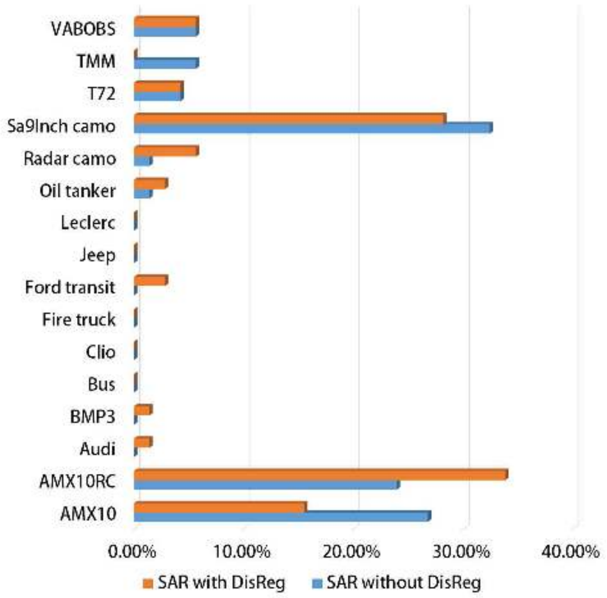


FIGURE 10. Recognition rate of SAR single-sensor stream for Sa19inch camo target with and without DisReg.

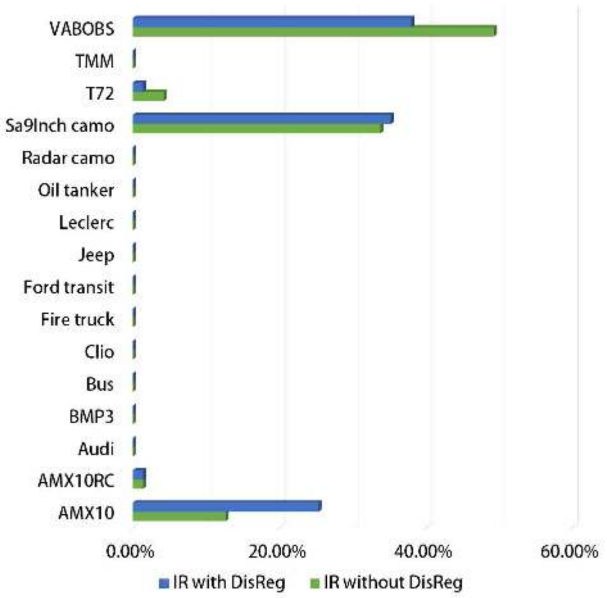


FIGURE 11. Recognition rate of IR single-sensor stream for Sa19inch camo target with and without DisReg.

VII. DISCUSSION

The conventional single-sensor streams are trained to extract the features that are appropriate for processing the given single-sensor information for a specific purpose. These features, trained using conventional learning problems, are only suitable for single-sensor information processing, and not fusion. DisReg promotes all the single-sensor streams to train the features that generate mutually complementary effects in the fusion by allowing them to learn different types of features from each other. Therefore, in single-sensor streams these features may result in a degraded outcome compared to conventional features. This can be confirmed based on the recognition test result that a single-sensor stream with DisReg yields a slightly degraded recognition result than conventional single-sensor streams (Table 4). However, only the fusion algorithm that exploits the single-sensor streams trained using the proposed regularization achieves a higher recognition rate than others, demonstrating that the features trained with DisReg are more effective in fusion. Nevertheless, the proposed method does not achieve an improvement in the recognition rate of the Sa19inch camo class. In general, for achieving high recognition rates the features to be used should be able to represent the consistent patterns of the data of a certain target. However, in the case of the Sa19inch camo class, each basic single-sensor stream shows a very low recognition rate of approximately 30%, which implies that the basic single-sensor streams fail to learn the consistent pattern of the Sa19inch camo class data (Figs. 10 and 11). When the single-sensor streams have difficulty in learning a pattern to represent a particular target, the learning method using DisReg does not assist them in improving the recognition rate. This is because backpropagation in DisReg (that aids each single-sensor stream to learn distinct types of features) is not identical to the training feature for learning the pattern, as in case of cross-entropy. Therefore, if basic single-sensor streams fail to learn the representative pattern of the data, then they may degrade the recognition result even though the complementary properties are imposed on the features by DisReg. Except this class, the recognition rate of most targets is improved because the advantages of fusion exceed the degradation of the single-sensor stream. Conventional algorithms have been developing the structure and regularization that are suitable for the fusion process for improving the fusion result. For fusion, the architecture should be able to exploit the various features effectively, and regularization should be designed to cause each single-sensor stream to learn mutually complementary features. The proposed algorithm utilizes the different levels of features using multistage structures and induce each single-sensor stream to learn the features that are suitable for the fusion process by DisReg. Therefore, the proposed fusion algorithm achieves a higher recognition rate compare with the conventional fusion methods (Table 8).

VIII. CONCLUSION

We developed fusion architecture and regularization for fusion using deep learning. The local image features generated from each single-sensor stream were fused on each layer of the proposed fusion architecture. The proposed regularization is based on the normalized cross-correlation between the feature maps of two single-sensor streams, and it induces each single-sensor stream to learn the complementary features. To evaluate the proposed algorithm, the target-recognition accuracy of the proposed algorithm was compared with that of the conventional fusion approaches using an

SAR/IR image database. In addition, we measured recognition rates for the proposed algorithm with several model parameter settings, to clarify the effect of the local image fusion scheme used in the proposed fusion stream. The effects of DisReg on the single-sensor streams and fusion stream are comprehensively discussed based on the experiment results. The proposed fusion approach achieves the highest recognition rate and is insensitive to parameter reduction. The proposed fusion architecture and learning method for a single-sensor stream are not limited by the sensor-types and their combinations. Therefore, it is expected that the proposed fusion architecture can be applied to various fields besides the target recognition field. We believe that the proposed algorithm can be exploited for both surveillance operations using SAR/IR sensor fusion, and for various civil research fields.

ACKNOWLEDGMENT

This paper was presented at the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Daegue, Republic of Korea, November 2017 [1].

REFERENCES

[1] Y.-R. Cho, S. Shin, S.-H. Yim, H.-W. Cho, and S. Woo-Jin, ''Multistage fusion and dissimilarity regularization for deep learning,'' in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst. (MFI)*, Nov. 2017, pp. 586–591.

[2] D. Ramachandram and G. W. Taylor, ''Deep multimodal learning: A survey on recent advances and trends,'' *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 96–108, Nov. 2017.

[3] A. Klausne, A. Tengg, and B. Rinner, ''Vehicle classification on multisensor smart cameras using feature- and decision-fusion,'' in *Proc. 1st ACM/IEEE Int. Conf. Distrib. Smart Cameras (ICDSC)*, Sep. 2007, pp. 67–74.

[4] W. He, W. Feng, Y. Peng, Q. Chen, G. Gu, and Z. Miao, ''Multi-level image fusion and enhancement for target detection,'' *Opt.-Int. J. Light Electron Opt.*, vol. 126, no. 11, pp. 1203–1208, 2015.

[5] C. Chen, R. Jafari, and N. Kehtarnavaz, ''A survey of depth and inertial sensor fusion for human action recognition,'' *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 4405–4425, 2017.

[6] M. Shoyaib, M. Abdullah-Al-Wadud, and O. Chae, ''A skin detection approach based on the Dempster–Shafer theory of evidence,'' *Int. J. Approx. Reason.*, vol. 53, no. 4, pp. 636–659, 2012.

[7] P. Natarajan *et al.*, ''Multimodal feature fusion for robust event detection in Web videos,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1298–1305.

[8] X. Wei, Z. Tao, C. Zhang, and X. Cao, ''Structured saliency fusion based on Dempster–Shafer theory,'' *IEEE Signal Process. Lett.*, vol. 22, no. 9, pp. 1345–1349, Sep. 2015.

[9] D. A. Dornfeld and M. DeVries, ''Neural network sensor fusion for tool condition monitoring,'' *CIRP Ann.-Manuf. Technol.*, vol. 39, no. 1, pp. 101–105, 1990.

[10] Y. Ma, J. Chen, C. Chen, F. Fan, and J. Ma, ''Infrared and visible image fusion using total variation model,'' *Neurocomputing*, vol. 202, pp. 12–19, Aug. 2016.

[11] Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' *Nature*, vol. 521, pp. 436–444, May 2015.

[12] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, ''Multimodal deep learning for robust RGB-d object recognition,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2015, pp. 681–687.

[13] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, ''Multimodal deep learning,'' in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 689–696.

[14] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno, and T. Ogata, ''Audio-visual speech recognition using deep learning,'' *Appl. Intell.*, vol. 42, no. 4, pp. 722–737, 2015.

[15] Y. Mroueh, E. Marcheret, and V. Goel, ''Deep multimodal learning for audio-visual speech recognition,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 2130–2134.

[16] M. Ghayoumi and A. K. Bansal, ''Multimodal architecture for emotion in robots using deep learning,'' in *Proc. Future Technol. Conf. (FTC)*, 2016, pp. 901–907.

[17] W. Liu, W.-L. Zheng, and B.-L. Lu. (2016). ''Multimodal emotion recognition using multimodal deep learning.'' [Online]. Available: https://arxiv.org/abs/1602.08225

[18] A. Valada, G. L. Oliveira, T. Brox, and W. Burgard, ''Deep multispectral semantic scene understanding of forested environments using multimodal fusion,'' in *Proc. Int. Symp. Exp. Robot. (ISER)*, Tokyo, Japan, 2016, pp. 465–477. [Online]. Available: https://link.springer.com/book/10.1007/978-3-319-50115-4

[19] P. Mamoshina, A. Vieira, E. Putin, and A. Zhavoronkov, ''Applications of deep learning in biomedicine,'' *Mol. Pharmaceutics*, vol. 13, no. 5, pp. 1445–1454, 2016.

[20] S. Song *et al.*, ''Multimodal multi-stream deep learning for egocentric activity recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun./Jul. 2016, pp. 24–31.

[21] W. Ouyang, X. Chu, and X. Wang, ''Multi-source deep learning for human pose estimation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2329–2336.

[22] M. O. Simón *et al.*, ''Improved rgb-dt based face recognition,'' *IET Biometrics*, vol. 5, no. 4, pp. 297–303, 2016.

[23] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, ''Vision meets robotics: The KITTI dataset,'' *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.

[24] J. Wagner, V. Fischer, M. Herman, and S. Behnke, ''Multispectral pedestrian detection using deep fusion convolutional neural networks,'' in *Proc. 24th Eur. Symp. Artif. Neural Netw., Comput. Intell. Mach. Learn. (ESANN)*, 2016, pp. 509–514.

[25] C. Feichtenhofer, A. Pinz, and A. Zisserman, ''Convolutional two-stream network fusion for video action recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1933–1941.

[26] N. Srivastava and R. R. Salakhutdinov, ''Multimodal learning with deep Boltzmann machines,'' in *Proc. Adv. neural Inf. Process. Syst.*, 2012, pp. 2222–2230.

[27] S. Rastegar, M. Soleymani, H. R. Rabiee, and S. M. Shojaee, ''MDL-CW: A multimodal deep learning framework with crossweights,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2601–2609.

[28] O. Mees, A. Eitel, and W. Burgard, ''Choosing smartly: Adaptive multimodal fusion for object detection in changing environments,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 151–156.

[29] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, ''Adaptive mixtures of local experts,'' *Neural Comput.*, vol. 3, no. 1, pp. 79–87, 2012.

[30] E. Park, X. Han, T. L. Berg, and A. C. Berg, ''Combining multiple sources of knowledge in deep CNNS for action recognition,'' in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–8.

[31] K. Sohn, W. Shang, and H. Lee, ''Improved multimodal deep learning with variation of information,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2141–2149.

[32] L. Ma, Z. Chen, L. Xu, and Y. Yan, ''Multimodal deep learning for solar radio burst classification,'' *Pattern Recognit.*, vol. 61, pp. 573–582, Jan. 2017.

[33] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang, ''Exploiting feature and class relationships in video categorization with regularized deep neural networks,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 352–364, Feb. 2017.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[35] J. Ooi and K. Rao, ''New insights into correlation-based template matching,'' in *Proc. SPIE, Appl. Artif. Intell. IX*, vol. 1468, pp. 740–751, Mar. 1991. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/1468/0000/New-insights-into-correlation-based-template-matching/10.1117/12.28670.short?SSO=1

[36] R. Brunelli and T. Poggio, ''Face recognition: Features versus templates,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 10, pp. 1042–1052, Oct. 1993.

[37] C. E. Costa and M. Petrou, ''Automatic registration of ceramic tiles for the purpose of fault detection,'' *Mach. Vis. Appl.*, vol. 11, no. 5, pp. 225–230, 2000.

[38] A. Subramaniam, P. Balasubramanian, and A. Mittal, "NCC-net: Normalized cross correlation based deep matcher with robustness to illumination variations," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1944–1953.

[39] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *Proc. IEEE CVPR*, vol. 1, Jun. 2017, no. 2, p. 3.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[41] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 21–37. [Online]. Available: https://link.springer.com/book/10.1007%2F978-3-319-46448-0 and http://www.eccv2016.org/

[42] J.-J. Won, S. Kim, Y. Cho, W.-J. Song, and S.-H. Kim, "Synthetic SAR/IR database generation for sensor fusion-based A.T.R.," in *Proc. 12th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI)*, Oct. 2015, pp. 421–424.

[43] J. Latger, T. Cathala, N. Douchin, and A. Le Goff, "Simulation of active and passive infrared images using the se-workbench," in *Proc. Defense Secur. Symp.*, 2007, p. 654302.

[44] T. Cathala, N. Douchin, A. Joly, and S. Perzon, "The use of se-workbench for aircraft infrared signature, taken into account body, engine, and plume contributions," in *Proc. SPIE, Infrared Imag. Syst., Design, Anal., Modeling, Test. XXI*, vol. 7662, p. 76620U, Apr. 2010. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/7662/76620U/The-use-of-SE-WORKBENCH-for-aircraft-infrared-signature-taken/10.1117/12.850691.short

[45] S. Kim, W.-J. Song, and S.-H. Kim, "Robust ground target detection by SAR and IR sensor fusion using Adaboost-based feature selection," *Sensors*, vol. 16, no. 7, p. 1117, 2016.

[46] H.-W. Cho, Y.-R. Cho, W.-J. Song, and B.-K. Kim, "Image matting for automatic target recognition," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 5, pp. 2233–2250, Oct. 2017.

[47] M. A. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

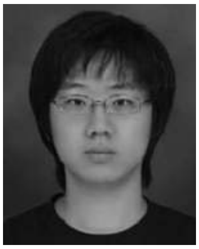
[48] J. Masci, U. Meier, D. Cire an, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Proc. Artif. Neural Netw. Mach. Learn. (ICANN)*, 2011, pp. 52–59.

[49] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.

[50] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[51] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

[52] D. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980



YOUNG-RAE CHO received the B.S. degree in electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2012, where he is currently pursuing the Ph.D. degree. Since 2012, he has been a Research Assistant with the Department of Electrical Engineering, POSTECH. His research areas of interest include image processing, radar signal processing, adaptive signal processing, machine learning, and deep learning.

SEUNGJUN SHIN received the B.S. and Ph.D. degrees in electrical engineering from the Pohang University of Science Technology (POSTECH), Pohang, South Korea, in 2012 and 2018, respectively. Since 2012, he has been a Research Assistant with the Department of Electrical Engineering, POSTECH. His research interests include multimedia signal processing for display and image signal processing.

SUNG-HYUK YIM received the B.S. degree in electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2012, where he is currently pursuing the Ph.D. degree. Since 2012, he has been a Research Assistant with the Department of Electrical Engineering, POSTECH. His research areas of interest include image processing and adaptive signal processing.

KYEONGBO KONG received the B.S. degree in electronics engineering from Sogang University, Seoul, South Korea, in 2015, and the M.S. degree in electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2017. Since 2015, he has been a Research Assistant with the Department of Electrical Engineering, POSTECH, where he is currently pursuing the Ph.D. degree. His current research interests include image processing, computer vision, machine learning, and deep learning.

HYUN-WOONG CHO received the B.S. and Ph.D. degrees in electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2009 and 2017, respectively. Since 2009, he has been a Research Assistant with the Department of Electrical Engineering, POSTECH. His research areas of interest include image processing, radar signal processing, and machine learning.

WOO-JIN SONG was born in Seoul, South Korea, in 1956. He received the B.S. and M.S. degrees in electronics engineering from Seoul National University, Seoul, in 1979 and 1981, respectively, and the Ph.D. degree in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1986. From 1981 to 1982, he was with the Electronics and Telecommunication Research Institute, Daejeon, South Korea. In 1986, he was employed by Polaroid Corporation as a Senior Engineer, working on digital image processing, where he was promoted to a Principal Engineer, in 1989. In 1989, he joined the faculty of the Pohang University of Science and Technology (POSTECH), Pohang, Korea, where he is currently a Professor of electronic and electrical engineering. His current research interests are in the areas of digital signal processing, in particular, radar signal processing, signal processing for digital television and multimedia products, and adaptive signal processing.