On the role of multimodal learning in the recognition of sign language

Pedro M. Ferreira · **Jaime S. Cardoso**[1] · **Ana Rebelo**[2]

Abstract
Sign Language Recognition (SLR) has become one of the most important research areas in the field of human computer interaction. SLR systems are meant to automatically translate sign language into text or speech, in order to reduce the communicational gap between deaf and hearing people. The aim of this paper is to exploit multimodal learning techniques for an accurate SLR, making use of data provided by Kinect and Leap Motion. In this regard, single-modality approaches as well as different multimodal methods, mainly based on convolutional neural networks, are proposed. Our main contribution is a novel multimodal end-to-end neural network that explicitly models private feature representations that are specific to each modality and shared feature representations that are similar between modalities. By imposing such regularization in the learning process, the underlying idea is to increase the discriminative ability of the learned features and, hence, improve the generalization capability of the model. Experimental results demonstrate that multimodal learning yields an overall improvement in the sign recognition performance. In particular, the novel neural network architecture outperforms the current state-of-the-art methods for the SLR task.

Keywords Sign language recognition · Multimodal learning · Convolutional neural networks · Kinect · Leap motion

1 Introduction

Sign language (SL) is an integral form of communication especially used by hearing impaired people within deaf communities worldwide. It is a visual means of communication, with its own lexicon and grammar, that combines articulated hand gestures along with facial expressions to convey meaning. The population of SL speakers is extended by familiars and friends of the deaf, interpreters and the curious, who learn the language by their

Pedro M. Ferreira
pmmf@inesctec.pt

[1] INESC TEC and Universidade do Porto, Porto, Portugal

[2] INESC TEC and Univ Portucalense, Oporto, Portugal

own initiative. As most of hearing people are unfamiliar with SL, deaf people find it difficult to interact with the hearing majority. The result is the isolation of deaf communities from the overall society.

In this regard, Sign Language Recognition (SLR) has becoming an appealing research topic in modern societies. SLR systems have several applications. Their main purpose is to automatically translate the signs from video or images into the corresponding text or speech. This is important not only to bridge the communicational gap between deaf and hearing people but also to increase the amount of contents to which the deaf can access. The creation of educational tools or games for deaf and visual dictionaries of sign language are some interesting examples of SLR use cases.

SLR is a multidisciplinary challenging task since it involves several fields, such as sign capturing methods, computer vision, machine learning, human action and sign language understanding. Although several SLR systems have been proposed in the literature, there are still many opportunities for research and improvement.

1.1 Related work

The SLR task can be addressed by using wearable devices or vision-based approaches. Vision-based SLR is less invasive since there is no need to wear cumbersome devices that may affect the natural signing movement. A vision-based SLR system is typically composed by three main building blocks: (i) hand segmentation and/or tracking, (ii) feature extraction, and (iii) sign recognition. Figure 1 depicts some examples of different vision-based SLR systems, according to the data acquisition sensor.

The first vision-based SLR approaches were just based on the extraction of colour information from images or videos [1, 4]. In general, a set of relevant colour-based features is extracted to be used in a traditional classification module that provides the sign recognition. As these representations contain a 2D description of the three-dimensional hand pose, colour-based approaches often demonstrate several limitations especially when the

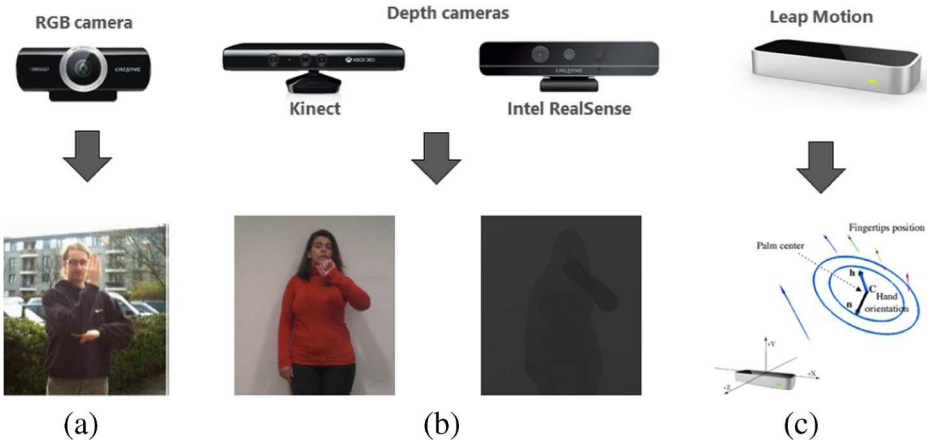


Fig. 1 Vision-based SLR systems: colour information provided by RGB cameras (**a**), colour and depth information provided by depth cameras (**b**) and the hand position and orientation provided by Leap Motion (**c**)

signs to be recognized involve complex 3D movements (i.e., in which there are several inter-occlusions between the various hand parts).

With the emergence of low-cost consumer depth cameras (e.g., Microsoft's Kinect™), some SLR systems have explored the 3D information for an accurate gesture recognition [5, 6, 11, 29]. This new layer of information may be particularly helpful when the position and angles of the fingers are needed with high precision.

Bergh et al. [5] demonstrated that depth information can be used together with colour information to increase the recognition accuracy, especially when there is superposition between the hands and the face. In [6], multiple depth-based descriptors are fed into a SVM classifier for gesture recognition. In a first stage, the hands are detected and segmented using both colour and depth information. Afterwards, different subsets of depth-based features, such as distance, elevation, curvature and palm area features, are extracted.

The recent introduction of Leap Motion has launched new research lines for gesture recognition.Instead of a complete depth map, the Leap Motion sensor directly provides the 3D spatial positions of the fingertips and the hand orientation with quite accuracy (see Fig. 1). One of the first studies referring to the utilization of Leap Motion for SLR has been presented in [17]. The authors stated that, although Leap Motion may have a great potential for sign recognition, it is not always able to recognize all fingers in some hand configurations (e.g., when the hand is not perpendicular to the camera). In order to overcome that limitation, Marin et al. [14, 15] combined the input data from Leap Motion with Kinect. The authors proposed a feature-level fusion approach with hand-crafted features extracted from two modalities (i.e., depth data from Kinect and Leap Motion data). The extracted features are based on the distances between the hand contour points and the hand's centroid, the curvature of the hand contour, and the convex hull of the hand shape.

More recently, Ferreira et al. [7] also explored the complementary characteristics of Kinect and Leap Motion for gesture recognition. Instead of traditional hand-crafted approaches, the authors proposed several multimodal deep learning strategies, mainly based on Convolutional Neural Networks (CNNs). The advantage is to avoid the extraction of hand-crafted features and the inherent difficulty of designing reliable features to the large variations of hand gestures. In principle, a traditional multimodal end-to-end deep neural network, as proposed in [7], should be able to encode the relationships and the complementary aspects of the input modalities (i.e., Kinect and Leap Motion). However, in practice, a multimodal deep neural network requires a lot of training data to generalize well. This is not the case of the SLR context where large multimodal datasets, with both Kinect and Leap Motion data, are scarce.

1.2 Deep multimodal regularization

In the deep multimodal learning context, an important design consideration is the formulation of well-designed loss functions along with regularization terms that enforce inter-modality and intra-modality relationships. Although the relationship between different modalities has not been thoroughly investigated in the SLR task, several deep multimodal regularization techniques have been proposed in the scope of more generic problems, such as RGB-D object recognition [12, 20, 24, 25, 27], transfer learning [3], and deep feature embeddings [10, 19].

In order to learn relationships between modalities, Sohn et al. [20] proposed a loss function that minimizes the variation of information between modalities. The underlying idea is that learning to maximize the amount of information that one data modality has about the others would allow multimodal generative models to reason about the missing data modality

given partial observations. Wu et al. [28] explored both inter-modality and intra-class relationships, for video semantic classification, by imposing trace-norm based regularizations on the shared and output layers of the neural network. Loss functions that enforce inter- and intra-modality correlations have also been proposed in [24, 25]. In particular, Wang et al. [24] proposed a multimodal fusion layer that uses matrix transformations to enforce a common part to be shared by features of different modalities while retaining modality-specific properties. Lenz et al. [12] introduced a structured regularization term in the loss function, in order regularize the number of modalities used per feature (node). In this regard, the model is able to learn correlated features between multiple input modalities, while discarding weak correlations between them.

The formulation of well-designed loss functions, along with additional regularization terms, have also been explored in many other domains, such as transfer learning [3, 23], deep feature embeddings [10, 19], and image retrieval [30] as well as to maximize domain-specific performance metrics [8, 13, 30]. A very comprehensive and recent survey on deep multimodal learning and regularization can be found in [18].

1.3 Major contributions

This paper presents a novel multimodal end-to-end neural network, called End-to-End Network with Regularization (EENReg), that explicitly models the complementary characteristics of the input modalities. Our novel architecture, along with a well-designed loss function, results in a model that jointly learns to extract representations that are specific to each modality as well as shared representations across modalities. The underlying idea is to increase the discriminative ability of the learned features by regularizing the entire learning process and, hence, improve the generalization capability of multimodal deep models. The present work expands the ideas proposed in [7], improving their results. In particular, our main novelties are:

- A comparative study between single-modality and multimodal learning techniques, in order to demonstrate the effectiveness of multimodal learning in the overall sign recognition performance;
- The introduction of a more robust hand gesture detection algorithm, which promotes an overall improvement in the sign recognition performance;
- The implementation of a more complete randomized data augmentation scheme, which allows training deeper neural networks without overfitting;
- The proposal of a novel multimodal end-to-end neural network architecture, the so-called EENReg, along with a well-designed loss function that explicitly learns to extract deep features representations that are unique and shared between modalities. By inducing the model to jointly learn both modality-specific and modality-shared features, the proposed EENReg outperforms the state-of-the-art multimodal approaches.

Our work is inspired by the recent works on transfer learning [3] and local similarity-aware deep feature embeddings [10], which explore the complementary properties between the source and target domains. However, we extent their ideas for supervised deep multimodal learning, in particular, for the SLR task, which implied an entire refinement of the neural network architecture, loss function, and regularization terms.

The paper is organized in six sections including the Introduction (Section 1). Section 2 presents a pre-processing step for segmenting the hands from the noisy background, before sign recognition. The implemented single-modality and conventional multimodal SLR methodologies are fully described in Sections 3 and 4, respectively. The proposed EENReg

model, which is the major contribution of the paper, is presented in Section 5. Section 6 reports the experimental evaluation of the proposed methodologies. Finally, conclusions and some topics for future work are presented in Section 7.

2 Pre-processing for hand detection

Both Kinect modalities, colour and depth, require a pre-processing step in order to segment the hands, from the noisy backgroundof the image, before feature extraction and sign recognition. As illustrated in Fig. 2, the developed hand segmentation method exploits both colour and depth information of Kinect.

In a first step, a skin colour segmentation, in the YCbCr colour space, is performed to roughly distinguish skin pixels from background pixels. The YCbCr colour space was adopted since it is perceptually uniform and separates luminance and chrominance, which makes this colour space suitable for skin colour detection [9]. The YCbCr colour space comprises three channels, representing the luminance component (Y) and the chrominance components (Cb and Cr). The conversion from RGB to YCbCr is simply defined as follows:

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{1}$$

$$C_b = (B - Y) \cdot 0.564 + 128 \tag{2}$$

$$C_r = (R - Y) \cdot 0.713 + 128 \tag{3}$$

For illumination-invariance, the implemented skin colour segmentation method just makes use of both chrominance components (CbCr). In the CbCr subspace, the distribution of skin and background colours is modelled each one by a multivariate Gaussian mixture

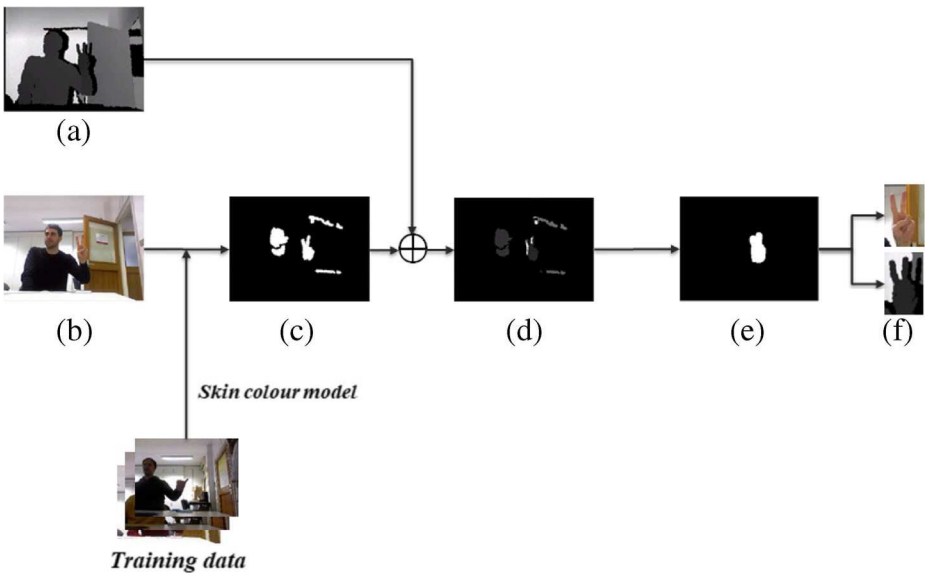


Fig. 2 Hand detection methodology: input depth image (**a**), input colour image (**b**), skin colour segmentation (**c**), filtered depth map (**d**), hand segmentation result (**e**) and the cropped colour and depth images (**f**)

model S and B, respectively. Therefore, the probability that a pixel j with colour value X_j belongs to the skin colour model S is defined as:

$$p(X_j \mid S) = \sum_{i=1}^{k} \gamma_i p(X_j \mid S_i) = \sum_{i=1}^{k} \frac{\gamma_i}{(2\pi)^{l/2} |\Sigma_{S_i}|^{1/2}} \exp\left[-\frac{1}{2} (X_j - \mu_{S_i})^T \Sigma_{S_i}^{-1} (X_j - \mu_{S_i})\right], \quad (4)$$

where l denotes the feature space dimension, k represents the number of Gaussian components of S, each one characterized by its mean vector μ_{S_i}, covariance matrix Σ_{S_i} and proportions γ_i. Likewise, the probability of a pixel belonging to the background colour model B is modelled in a similar manner.

After obtaining the skin model S and the background model B, the skin colour segmentation is performed by maximum likelihood classification of pixels within a test image. That is, a pixel with colour value X is classified as skin pixel if the following condition is verified:

$$p(X \mid S) > p(X \mid B) \quad (5)$$

As illustrated in Fig. 2c, the skin colour segmentation process results in a binary mask of the skin coloured objects present in the image (i.e., hand, face or other uncovered body parts). This binary mask is then used to filter the depth map, in order to only retain depth samples associated with skin coloured objects (see Fig. 2d). The underlying assumption is that the closest skin coloured object of the image corresponds to the hand, as the signer is typically the nearest object to the camera.

After this stage, hand segmentation is performed on the filtered depth map using a region growing technique. First, a search for the region with the minimum depth value D_{min} on the filtered depth map is performed. The corresponding region R_{min} is chosen as the seed region for the hand detection process, if its area is greater than a threshold T_{area}; otherwise the next closest region is selected. The area criterion is used so that the selected R_{min} does not correspond to an isolated artefact due to measurement noise. In the next step, the neighbouring pixels are examined and added to the seed region R_{min} based on a homogeneity criterion (i.e., if the depth value difference between those pixels and R_{min} does not exceed a threshold T_{depth}). This process is applied iteratively until no more pixels satisfy the homogeneity criterion. As illustrated in Fig. 2e, the segmented hand is then represented by all pixels that have been merged during this iterative procedure.

Once the segmentation process is completed, the original colour and depth images are both cropped by the bounding box of the segmented sign and, then, these resulting cropped images are resized to the average sign size of the training set (see Fig. 2f).

Fig. 3 Illustration of the background suppression methodology for a given colour image: original cropped colour image (**a**), Euclidean distance map of each pixel to the segmentation mask centroid (**b**), distance transform of the segmentation mask (**c**), linear combination of the two distance maps (**d**) and its application on the cropped colour image (**e**)

To further reduce of the influence of the background in the recognition task, a background suppression methodology is applied to the cropped images (see Fig. 3). First, a Euclidean distance map of each pixel to the segmentation mask centroid as well as the distance transform of the segmentation mask are computed (Fig. 3b and c, respectively). These maps are linear combined and, then, multiplied with the cropped image. As illustrated in Fig. 3e, the final result is the fading out of the background pixels according to their distance to the segmentation centroid, while it keeps the foreground pixels unchanged.

Finally, the image inputs are normalized to ensure that each pixel (i.e., input parameter) has a similar data distribution and, hence, make converge faster while training the models. Data normalization is done by subtracting the mean from each pixel, and then dividing the result by the standard deviation. For more pre-processing scenarios in deep learning, the reader should consider the following research works [26, 31].

3 Single-modality sign recognition

In this section, the implemented single-modality methodologies for SLR are presented. For both Kinect modalities (colour and depth), we resorted to a deep learning strategy based on convolutional neural networks (CNNs); whereas for Leap Motion we implemented a traditional machine learning pipeline with hand-crafted feature extraction. This choice was motivated by the different nature of the data of these modalities. As the leap motion data is already at a high semantic level (i.e., well structured features), a shallow classifier is suitable for making predictions.

3.1 Kinect modalities (colour and depth)

3.1.1 CNN architecture

The implemented neural network follows the traditional CNN architecture for classification [21]. It starts from several sequences of convolution-convolution-pooling layers to fully connected layers. More especifically, the implemented CNN is composed by six convolutional layers, three fully connected layers (or dense layers) and two max-pooling layers. The number of filters is doubled after each pooling operation. Finally, the last layer of the CNN is a softmax output layer, which contains the output probabilities for each class label.

The output node that produces the largest probability is chosen as the overall classification. The architecture of the implemented CNN is illustrated in Fig. 4.

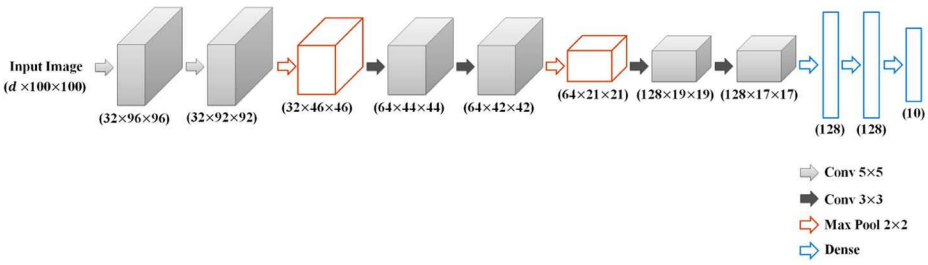


Fig. 4 The architecture of the implemented CNN model for single-modality sign recognition, using colour *(d* = 3*)* or depth *(d* = 1*)*

For training the model, the goal is to minimize the categorical cross-entropy, a commonly used loss function for classification tasks, which is given by:

$$L = -\sum_{i=0}^{N} \mathbf{y}_i^{\top} \log \hat{\mathbf{y}}_i, \tag{6}$$

where $\mathbf{y}_i$ is a column vector denoting the one-hot encoding of the class label for input i and $\hat{\mathbf{y}}_i$ are the softmax predictions of the model. The Nesterov's Accelerated Gradient Descent with momentum was used for optimization. During the training stage, several regularization techniques were applied to prevent overfitting (i.e., dropout, l_2-norm, and data augmentation). The implemented regularization techniques are fully described in Section 3.1.2.

3.1.2 Regularization

Dropout is a popular regularization technique introduced to prevent overfitting [22]. At each training stage, individual units are either "dropped out" or kept according to a defined probability p, so that a reduced network is left. Note that at each stage only the reduced network is trained on the data. Then, the removed units are reinserted into the network with their original weights. By avoiding training all units on all training data, dropout decreases overfitting in neural networks. In practice, dropout was applied to the fully connected layers of the implemented CNN.

Data augmentation is the process of increasing, artificially, the number of training samples, by means of different image transformations and noise addition. In here, a randomized data augmentation scheme based on both geometric and colour transformations is applied during the training step. The underlying idea is to increase the robustness of the CNN model to the wide range of hand gestures positions, poses, viewing angles as well as to different illumination conditions and contrasts. The data augmentation process is applied in an online-fashion, within every iteration, to a random half of the images of each mini-batch.

Specifically, the considered geometric transformations are obtained through the following randomized affine image warping:

$$\begin{bmatrix} x^{j} \\ y^{j} \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & k_1 \\ k_2 & 1 \end{bmatrix} \begin{bmatrix} x - t_1 \\ y - t_2 \end{bmatrix}, \tag{7}$$

where θ is the rotation angle, k_1 and k_2 are the skew parameters along the x and y directions. t_1 and t_2 denote both translation parameters and s is the scale factor. It is import to note that the values of these parameters are randomly selected from predefined sets (those sets are listed in Section 6). Pixels mapped outside the original image are assigned with the pixel values of their mirrored position.

The other type of image augmentation focuses on randomly normalizing the contrast of each channel in the training images. Formally, let S_c be the c-th channel of the input image, the new intensity value at each pixel in channel c is simply given by:

$$S_c^{j} = \begin{cases} 0 & , \text{ if } S_c < S_c(p_L) \\ \dfrac{S_c - S_c(p_L)}{S_c(p_H) - S_c(p_L)} & , \text{ if } S_c(p_L) \leq S_c \leq S_c(p_H) \\ 1 & , \text{ if } S_c > S_c(p_H) \end{cases}, \tag{8}$$

where p_L and p_H represent the lower and higher histogram percentiles that are randomly selected for the colour transformation, respectively. This scheme simulates the scenario that the input images are acquired with different intensities, contrasts and illumination conditions.

Fig. 5 Illustration of the implemented randomized data augmentation process: original colour images (top row) along with the corresponding augmented images (bottom row)

Figure 5 illustrates the application of the implemented data augmentation procedure. Although the resulting augmented images may be highly correlated between them, this randomized augmentation scheme significantly increases the size of the training set which allows the utilization of deep CNN architectures without overfitting.

3.2 Leap Motion

Unlike Kinect, Leap Motion does not provide a complete depth map, instead it directly provides a set of relevant features of hand and fingertips. The raw data of Leap Motion include the number of detected fingers, the position of the fingertips, the palm center, the hand orientation and the hand radius [15]. From these data, 3 different types of features were computed:

1. **Fingertip distances** $D_i = \|F_i - C\|, i = 1,\ldots,N$; where N denotes the number of detected fingers and D_i represents the 3D distances between each fingertip F_i and the hand centre C;
2. **Fingertip inter-distances** $I_i = \|F_i - F_{i+1}\|, i = 1, \ldots, N-1$; represent the 3D distances between consecutive fingertips;
3. **Hand direction** O: represents the direction from the palm position toward the fingers. The direction is expressed as a unit vector pointing in the same direction as the directed line from the palm position to the fingers;

where $\|\cdot\|$ denotes the l_2-norm, corresponding to the geometric distance between the fingertips. Both distance features are normalized by signer (user), according to the maximum fingertip distance and fingertip inter-distance of each user. This normalization is performed to make those features robust to people with different hand's size. Then, these 3 sets of features are used as input into a multi-class SVM classifier for sign recognition. The block diagram of the implemented Leap Motion-based sign recognition approach is illustrated in Fig. 6.

4 Conventional multimodal sign recognition

The data provided by Kinect and Leap Motion have quite complementary characteristics, since while Leap Motion provides few accurate and relevant key-points, Kinect produces

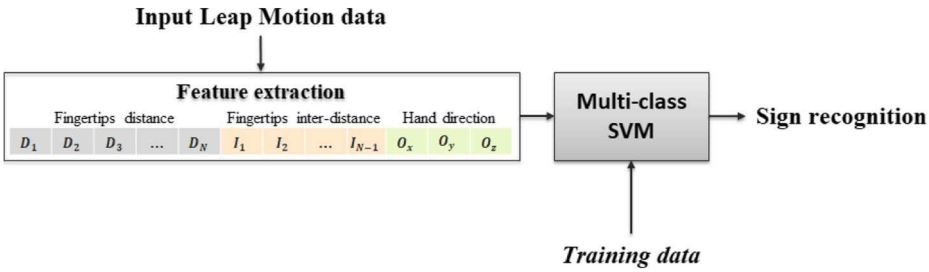


Fig. 6 Single-modality sign recognition methodology of Leap Motion data

both a colour image and a complete depth map with a large number of less accurate 3D points. Therefore, we intend to exploit them together for SLR purposes.

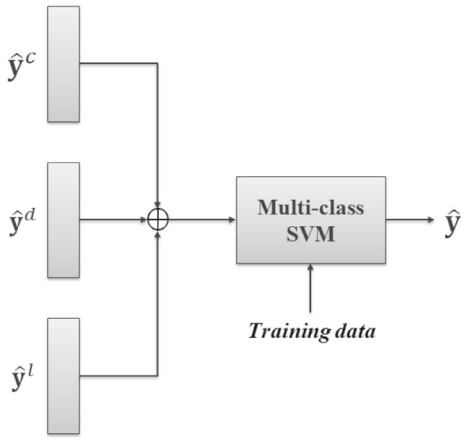
According to the level of fusion, multimodal fusion techniques can be roughly grouped into two main categories: (i) decision-level and (ii) feature-level fusion techniques [16]. As described in the following, we propose multimodal approaches of each fusion category for the SLR task, making use of 3 modalities (i.e., colour, depth and Leap Motion data).

Throughout the rest of the paper, let $\mathbf{X} = \{(x_i^c, x_i^d, x_i^l, y_i)\}_{i=1}^{N}$ denote the labeled multi-modal dataset of N samples used in this work, where x_i^c, x_i^d and x_i^l represent the i-th colour, depth and leap motion sample, respectively, and y_i denotes the ground-truth class labels.

4.1 Decision-level fusion

The purpose of decision-level fusion is to learn a specific classifier for each modality and, then, to find a decision rule between them. In this paper, we apply this concept making use of the output class probabilities of the models designed individually for each modality under analysis. Then, two main kinds of decision rules, to combine these class probabilities, were implemented: 1) pre-defined decisions rules, and 2) decision rules learned from the data. Let $\hat{\mathbf{y}}^c$, $\hat{\mathbf{y}}^d$ and $\hat{\mathbf{y}}^l$ be the predictions of colour, depth and leap motion modalities, respectively; then, the decision-level fusion schemes is illustrated in Fig. 7.

Fig. 7 Decision-level fusion, in which the decision rule is learned from the data. ⊕ is an aggregate operator representing the concatenation of the modality-specific class probabilities



4.1.1 Pre-defined decision rules

Herein, two different pre-defined decision rules were implemented. In the first approach, the final prediction is given by the argument that maximizes the averaged class probabilities. In the second approach, the final prediction is given by the model with the maximum confidence. The confidence of a model in making a prediction is measured by its highest class probability.

4.1.2 Learned decision rule

The underlying idea of this approach is to learn a decision rule from the data. Therefore, a descriptor that concatenates the class probabilities, extracted from the individual models of each modality, is created and, then, used as input into a multiclass SVM classifier for sign recognition.

4.2 Feature-level fusion

In general, feature-level fusion is characterized by three phases: (i) learning a feature representation/embedding, (ii) supervised training, and (iii) testing [16]. According to the order in which phases (i) and (ii) are made, feature-level fusion techniques can be roughly divided into two main groups: 1) End-to-end fusion, where the representation and the classifier are jointly learned; and 2) Multi-step fusion, where the representation is first learned and then the classifier is learned from it.

4.2.1 End-to-end fusion

The underlying idea of this approach is to jointly learn a multimodal deep feature representation $\mathbf{h}^m$ and a classifier $G(\mathbf{h}^m)$

that maps from the multimodal representation $\mathbf{h}^m$ to the task-specific predictions $\hat{\mathbf{y}}$. In our scenario, the neural network has three input-specific pipes, one for each data type: (i) colour $\mathbf{x}^c$, (ii) depth $\mathbf{x}^d$ and (iii) leap motion $\mathbf{x}^l$. Therefore, the multimodal feature embedding is simply given by the concatenation of the embeddings of each modality, such that:

$$\mathbf{h}^m = \overset{\Sigma}{\boldsymbol{f}^c(\mathbf{x}^c) \square \boldsymbol{f}^d(\mathbf{x}^d) \square \boldsymbol{f}^l(\mathbf{x}^l)}, \tag{9}$$

where $\boldsymbol{f}^c(\mathbf{x}^c)$, $\boldsymbol{f}^d(\mathbf{x}^d)$ and $\boldsymbol{f}^l(\mathbf{x}^l)$ denote the deep feature representations of colour, depth and leap motion modalities, respectively, and $\square$ represents the concatenation operation. While the embeddings of colour $\boldsymbol{f}^c(\mathbf{x}^c)$ and depth $\boldsymbol{f}^d(\mathbf{x}^d)$ are both learned by a CNN, the leap motion embedding $\boldsymbol{f}^l(\mathbf{x}^l)$ is learned by a classical multilayer neural network (NN) with two hidden layers (each one with 128 neurons). All the layers are trained together end-to-end. The architecture of the implemented end-to-end multimodal neural network is represented in Fig. 8a.

4.2.2 Multi-step fusion

As in the end-to-end approach, a multimodal representation $\mathbf{h}^m$ is created, by concatenating the modality-specific representations $\boldsymbol{f}^c(\mathbf{x}^c)$, $\boldsymbol{f}^d(\mathbf{x}^d)$ and $\boldsymbol{f}^l(\mathbf{x}^l)$. However, in this case, these representations are first learned individually. In particular, the representations $\boldsymbol{f}^c(\mathbf{x}^c)$ and $\boldsymbol{f}^d(\mathbf{x}^d)$ correspond to the activations extracted from the penultimate dense layer of

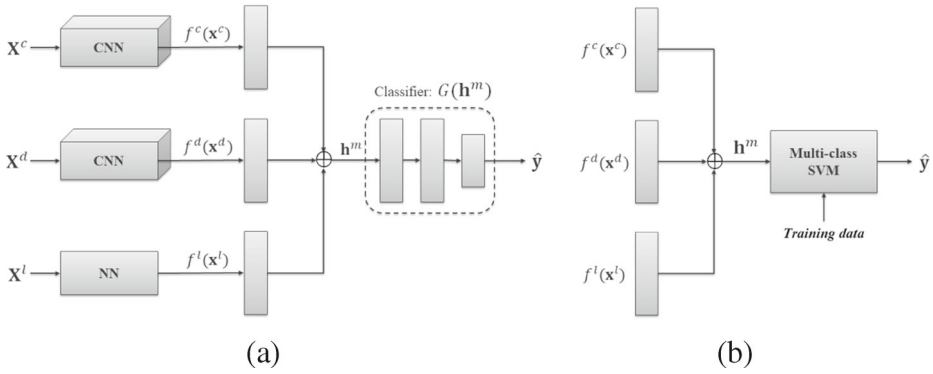


Fig. 8 Feature-level fusion schemes: end-to-end feature fusion (**a**) and multi-step feature fusion (**b**). $\square$ represents a concatenation operator

each modality-specific CNN, and $\boldsymbol{f}^l(\mathbf{x}^l)$ corresponds to the features extracted from the leap motion data (see Section 3). Then, for sign recognition, the multimodal representation vector $\mathbf{h}^m$ is fed into an additional classifier (i.e., a multi-class SVM). The multi-step feature-level fusion scheme is depicted in Fig. 8b.

5 Proposed multimodal end-to-end fusion with regularization

Ideally, the end-to-end network, as described in Section 4.2.1, should be able to encode the most relevant aspects of the input modalities for the classification task. However, in practice, training a multimodal end-to-end network with multiple input-specific pipes without overfitting is very difficult, mainly due to its huge number of parameters and, especially, if we have to deal with small datasets.

Rather than adopting a conventional multimodal learning structure that involves simple feature- or decision-level fusions, our goal is to further explore the implicit dependence between different modalities. In this regard, we propose a novel multimodal end-to-end architecture, called End-to-End Network with Regularization (EENReg), that explicitly models what is unique and shared between modalities. The underlying idea is that the desired multimodal features should comprise the agreement or shared properties between different modalities, while retaining the modality-specific properties that can only be captured by each modality individually. By imposing such regularization in the learning process, the model's ability to extract meaningful features for the classification should improve.

To induce the model to extract both modality-specific and modality-shared features, the EENReg network is composed by three private streams that are specific to each modality and three shared streams between modalities. In addition, the loss function is defined in a such manner that encourages independence between these private and shared representations. The result is a model that produces shared representations that are similar for all modalities and private representations that are modality-specific. The classifier is then trained on these private and shared representations to enhance discriminative capability of the model.

5.1 Architecture

As depicted in Fig. 9, the architecture of the EENReg comprises three private streams that are specific to each modality, three shared streams between modalities and a classifier.

While the purpose of each private stream is to transform the data of each modality into a new modality-specific feature representation, the purpose of each shared stream is to perform a mapping from each input modality to a shared representation between modalities. Therefore, the architecture of each stream consists of several sequences of convolution-convolution-pooling layers, for a typical CNN feature extraction, with a dense layer on top of that. In particular, each multimodal stream has the same architecture of the implemented CNN model for single-modality sign recognition (see Fig. 4 for more details). By concatenating the shared and modality-specific feature representations, a multimodal feature representation is, then, created.

Finally, a classifier that simply comprises three fully connected layers is fed with the multimodal feature representation. The last layer is a softmax output layer, which contains the output probabilities for each class label.

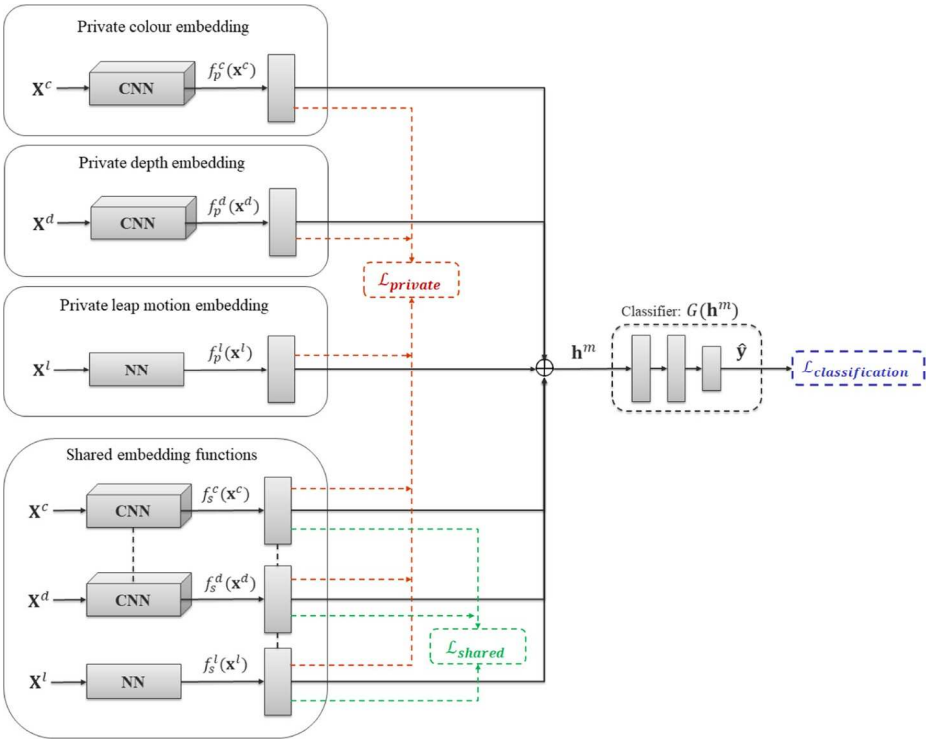


Fig. 9 The architecture of the EENReg model that explicitly learns to extract deep feature representations that are unique and shared between modalities

5.2 Learning

Let $f_s^m(\mathbf{x})$ be an embedding function that maps from an input sample $\mathbf{x}$ to a shared feature representation of modality m. Also, let $f_p^m(\mathbf{x})$ be an embedding function that maps from a sample $\mathbf{x}$ to a private feature representation that is specific to its modality. In order to maintain feature comparability, the representations $f_s^m(\mathbf{x})$ and $f_p^m(\mathbf{x})$ are first normalized onto the unit hypersphere, i.e., $\|f(\mathbf{x})\|_2 = 1$. Then, the EENReg model is trained by minimizing the following loss function:

$$L = L_{classification} + \alpha L_{private} + \beta L_{shared}, \tag{10}$$

where α, β are the weights that control the interaction of the loss terms. The classification loss, $L_{classification}$, trains the model to predict the output labels and corresponds to the categorical cross-entropy as defined in (6).

The purpose of the private loss $L_{private}$ is to encourage the shared and private representations of each modality to encode different aspects of the inputs. Therefore, $L_{private}$ is defined by imposing orthogonality between the shared and the private representations of each modality, such that:

$$\begin{aligned} L_{private} = \alpha_c \sum_{i=1}^{N} \langle f_p^c(x_i^c), f_s^c(x_i^c) \rangle^2 + \alpha_d \sum_{i=1}^{N} \langle f_p^d(x_i^d), f_s^d(x_i^d) \rangle^2 \\ + \alpha_l \sum_{i=1}^{N} \langle f_p^l(x_i^l), f_s^l(x_i^l) \rangle^2, \end{aligned} \tag{11}$$

where $\langle \cdot, \cdot \rangle$ is the dot product. α_c, α_d and α_l are the weights that control the orthogonality between each modality representations.

The shared loss L_{shared} encourages the shared representations of all modalities, $f_s^c(\mathbf{x}^c)$, $f_s^d(\mathbf{x}^d)$ and $f_s^l(\mathbf{x}^l)$, to be as similar as possible. Then, the shared loss is simply defined to minimize the pair-wise differences between the shared representations $f_s^c(\mathbf{x}^c)$, $f_s^d(\mathbf{x}^d)$ and $f_s^l(\mathbf{x}^l)$, such that:

$$\begin{aligned} L_{shared} = \beta_{cd} \sum_{i=1}^{N} \| f_s^c(x_i^c) - f_s^d(x_i^d) \|_2^2 + \beta_{cl} \sum_{i=1}^{N} \| f_s^c(x_i^c) - f_s^l(x_i^l) \|_2^2 \\ + \beta_{dl} \sum_{i=1}^{N} \| f_s^d(x_i^d) - f_s^l(x_i^l) \|_2^2, \end{aligned} \tag{12}$$

where $\|\cdot\|_2^2$ is the squared l_2-norm. β_{cd}, β_{cl} and β_{dl} are the weights of each pair-wise difference.

Finally, inference in an EENReg model is given by $\hat{y} = G(\mathbf{h}^m)$, where $\mathbf{h}^m$ represents a multimodal feature embedding given by merging (either by concatenation or sum) all private and shared feature representations, such that:

$$\mathbf{h}^m = f_p^c(\mathbf{x}^c) \| f_s^c(\mathbf{x}^c) \| f_p^d(\mathbf{x}^d) \| f_s^d(\mathbf{x}^d) \| f_p^l(\mathbf{x}^l) \| f_s^l(\mathbf{x}^l) \tag{13}$$

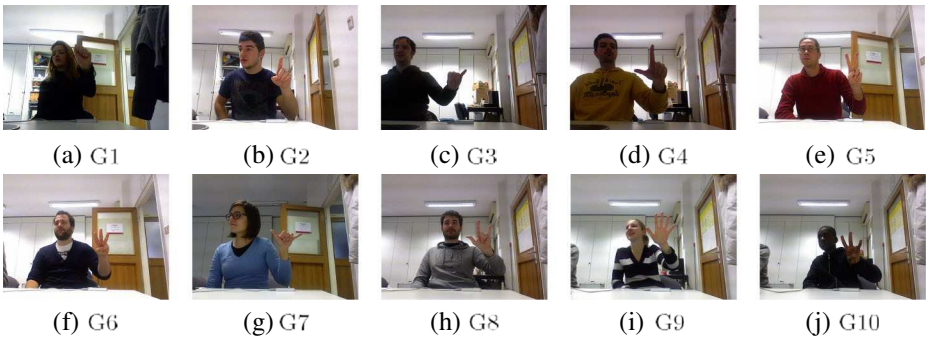


Fig. 10 Illustrative samples of 10 signs from the American Sign Language database [14, 15]

6 Experimental results

6.1 Dataset and evaluation protocol

The experimental evaluation of the proposed methodologies was performed in a public Microsoft Kinect and Leap Motion hand gesture recognition database [14, 15]. This is a balanced dataset of 10 classes, representing 10 static gestures from the American Sign Language (see Fig. 10). Each sign was performed by 14 different people, and repeated 10 times, which results in a total of 1400 gestures.

For each sign, data from both Leap Motion and Kinect were acquired together. The Kinect data include the colour images along with the corresponding depth maps.

To maximize the usage of the data in the evaluation process, the performance of the models was assessed using a k-fold cross validation scheme with signer independence, where $k = 5$. Therefore, all performance measures reported throughout this section are the average of their values computed in each split. This evaluation scheme, with $k = 5$, yields at each split a training set of 1100 images from 10 signers and test set of 300 images from the other 3 signers. The training set is further divided, also with signer independence, in 80% for training and 20% for validation.

6.2 Implementation details

The parameters of the hand segmentation algorithm were empirically defined based on the available dataset and remained the same in all the experiments. That is, the number of Gaussian components of the skin and background colour models was set to 2 and 4, respectively. In addition, $T_{area} = 5$ and $T_{depth} = 5$.

All deep models were implemented in Theano [2] and trained with the Nesterov's Accelerated Gradient Descent with momentum using a batch size of 50 samples. We used a learning rate with step decay, in which the initial learning rate was multiplied by 0.99 at each training epoch. The hyperparameters that are common to all the implemented models (i.e., the learning rate and the l_2 coefficient) as well as the specific hyperparameters of the EENReg model (i.e., both $L_{private}$ and L_{shared} coefficients) were optimized by means of a

Table 1 Hyperparameters sets

Hyperparameters	Acronym	Set
Leaning rate	–	$\{1e^{-03}, 1e^{-04}\}$
l_2-norm coefficient	–	$\{1e^{-04}, 1e^{-05}\}$ $L_{private}$
coefficients	$\alpha_c, \alpha_d, \alpha_l$	$\{1e^{-03}, 5e^{-03}, 1e^{-04}\}$
L_{shared} coefficients	$\beta_{cd}, \beta_{cl}, \beta_{dl}$	$\{1e^{-03}, 5e^{-03}, 1e^{-04}\}$

grid search approach and cross-validation on the training set. The dropout rate was empirically set as 0.4 for all the experiments. The range of values of the adopted hyperparameters' grid search is presented in Table 1. For a fair comparison, it is important to note that the CNNs streams of all multimodal networks have the same architecture of the CNN model employed for single-modality classification.

Regarding the parameters of the data augmentation scheme, the rotation angle θ was randomly sampled from $\{-\pi/18, -\pi/36, 0, \pi/36, \pi/18\}$ The skew parameters, k_1 and k_2, were both randomly sampled from $\{-0.1, 0, 0.1\}$ The scale parameter s was randomly sampled from five different resize factors $\{0.9, 0.95, 1, 1.05, 1.1\}$. Finally, the translation parameters t_1 and t_2 are randomly sampled integers from the interval $[0, 5]$. Note that these sets of values were selected carefully, so that the meaning of the sign is not changed after the transformation.

The adopted SVM classifier consists in a multi-class SVM classifier based on the one-against-one approach, in which a nonlinear Gaussian Radial Basis Function (RBF) kernel is used. The parameters (C, γ) of the RBF kernel are estimated using a grid search and cross-validation on the training set.

6.3 The potential of multimodal learning

In order to assess the potential of multimodal learning in the SLR context, we computed the rate of test signs for which each single-modality method made a correct prediction while the others were wrong.

As presented in Table 2, these results clearly demonstrate that there is a relative big potential to tackle the SLR problem via multi-modality. In particular, there is a higher complementarity between each Kinect modality (i.e., colour or depth) with the Leap Motion rather than between both Kinect modalities. For instance, there are 4.88% and 5.00% of test instances for which Leap Motion made correct predictions while colour and depth made incorrect ones, respectively.

Table 2 The potential of multimodal learning, expressed by the rate of test instances for which modality B made correct predictions while modality A made incorrect ones

Modality A	Modality B	Multi-modality potential (%)
Colour	Depth	3.88
Colour	Leap motion	4.88
Depth	Colour	4.25
Depth	Leap motion	5.00
Leap motion	Colour	15.50
Leap motion	Depth	15.25

Table 3 Experimental results of the single-modality approaches with and without data augmentation and background suppression. The results are presented in terms of classification accuracy (%). Bold number indicates the best method with the highest value of Acc

Modality	Acc (%)		
	w/o background suppression	w/o augmentation	full
Colour	90.12	82.61	**93.17**
Depth	91.22	88.22	**92.61**
Leap motion	–	–	**82.83**

6.4 Discussion

The experimental results of the proposed single-modality and multimodal sign recognition methodologies are presented in Tables 3 and 4, respectively. The results are reported in terms of classification accuracy (Acc), which is given by the ratio between the number of correctly classified signs t and the total number of test signs n: $Acc\% = \frac{t}{n} \cdot 100$.

A first observation, regarding single-modality approaches, is that both colour and depth outperform Leap Motion, with classification accuracies of 93.17%, 92.61% and 82.83%, respectively. However, it should be noticed that Leap Motion sign recognition does not require any kind of preprocessing in order to segment the hand from the background for feature extraction.

To validate the impact of the proposed background suppression method and data augmentation scheme, both colour and depth CNN models were trained without them. As presented in Table 3, both colour and depth single-modality models performed consistently worse without background suppression and data augmentation, which clearly demonstrate their importance in the overall sign recognition performance.

Table 4 Experimental results of the multimodal fusion methodologies. C, D and L denote colour, depth and leap motion modalities, respectively. The results are presented in terms of classification accuracy (%). Bold number indicates the best method with the highest value of Acc

(a) Proposed multimodal fusion methods

Fusion Level	Method	Involved modalities	Acc (%)
Feature	*End-to-end*	C + D	92.80
		C + D + L	94.20
	Multi-step	C + D	96.78
		C +D + L	97.11
	EENReg	C + D	96.17
		C + D + L	**97.66**
Decision	*Average rule*	C + D	95.78
		C + D + L	97.33
	Confidence rule	C + D	95.78
		C + D + L	96.44
	Learned rule	C + D	95.83
		C + D + L	97.44

(b) State-of-the-art methodologies

Method	Acc (%)
Marin et al. 2014 [14]	91.28
Marin et al. 2016 [15]	96.50

Table 5 The effect of each loss term in the EENReg model. Bold number indicates the best method with the highest value of Acc

Method (modalities)	Acc (%)		
	w/o $L_{private}$	w/o L_{shared}	All loss terms
EENReg (C + D + L)	97.06	96.88	**97.66**

In the first column, the $L_{private}$ term was removed from the loss. In the second column, the L_{shared} term was removed from the loss. The third column is replicated from Table 4 as it includes all loss terms. The results are presented in terms of classification accuracy (%)

The most interesting observation is that multimodal fusion often promotes an overall improvement in the sign recognition accuracy - see Table 4. These results clearly demonstrate the complementarity between the three modalities. Typically, the classification accuracy increases as each modality is added to the recognition scheme. In particular, the novel end-to-end feature fusion model (EENReg), provides the best overall classification accuracy *(Acc* 97.66%*)*. The EENReg clearly outperforms the other two implemented feature-level approaches, especially if compared with the traditional end-to-end feature fusion model. These results demonstrate that explicitly modeling what is unique and shared between modalities can improve the model's ability to extract highly discriminative features for the sign classification.

In order to assess the impact of the loss terms in the EENReg model, both $L_{private}$ and L_{shared} constraints were removed from the loss, during the training, one at a time. These results are reported in Table 5 and, clearly, suggest that each loss term contributes to a better generalization of the model as its performance was consistently worse without them.

Figure 11 shows the confusion matrix obtained for the best methodology, which is the proposed EENReg model. The classification accuracy is larger than 97% for all signs, with

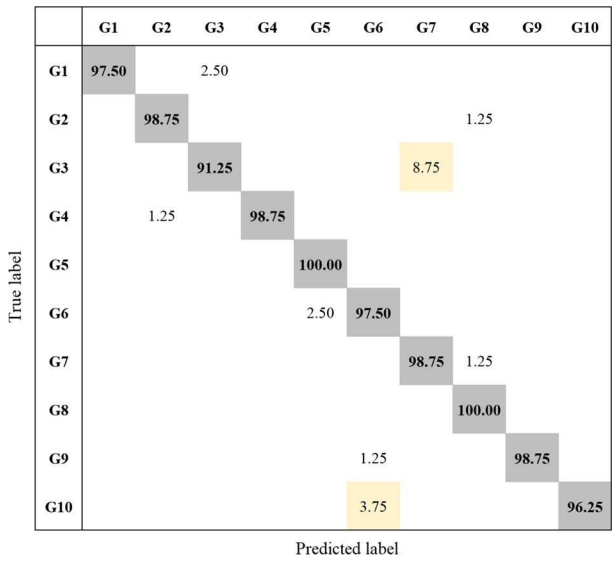


Fig. 11 Confusion matrix of the best implemented methodology, i.e., the EENReg model. Gray cells represent the true positives, while yellow cells correspond to the false positive rates greater than 2.5%

the exceptions of signs G3 and G10. While G3 is sometimes misclassified as G7, G10 is a few times misclassified as G6. This happens because these two pairs of signs have a very similar shape between each other. For instance, G10 and G6 just differ from each other in one finger position - see Fig. 10.

Finally, it is important to stress that the best implemented multimodal fusion approach (i.e., EENReg) outperformed both state-of-art methods [14] and [15], with an Acc of 97.66% against 91.28% and 96.50%, respectively.

7 Conclusions

This paper addresses the topic of static SLR, by exploring multimodal learning techniques, using of data from 3 distinct modalities: (i) colour; (ii) depth, both from Kinect; and (iii) Leap Motion data. In this regard, single-modality approaches as well as different multimodal methods, to fuse them at different levels, are proposed. Multimodal techniques include feature-level and decision-level fusion techniques.

Experimental results suggest that both Kinect modalities are more discriminative than the Leap Motion data. However, the most interesting observation is that, in general, multimodal learning techniques outperform single-modality methods.

Our main contribution is a novel end-to-end feature-level deep neural network that explicitly models private representations that are specific to each modality and shared feature representations that are similar between them. By imposing such constraints in the learning process, the model is able to jointly learn both modality-specific and modality-shared features and outperform the state-of-the-art multimodal approaches. As future work, it is expected to extend the proposed methodologies for dynamic signs (i.e., for video).

References

1. Adithya V, Vinod PR, Gopalakrishnan U (2013) Artificial neural network based method for indian sign language recognition. In: 2013 IEEE conference on information communication technologies (ICT), pp 1080–1085. https://doi.org/10.1109/CICT.2013.6558259
2. Bastien F, Lamblin P, Pascanu R, Bergstra J, Goodfellow IJ, Bergeron A, Bouchard N, Bengio Y (2012) Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop
3. Bousmalis K, Trigeorgis G, Silberman N, Krishnan D, Erhan D (2016) Domain separation networks. In: lee DD, Sugiyama M, Luxburg UV, Guyon I, Garnett R (eds) Advances in neural information processing systems 29, pp 343–351
4. Cooper H, Bowden R (2007) Large lexicon detection of sign language. Springer, Berlin, pp 88–97
5. den Bergh MV, Gool LV (2011) Combining rgb and tof cameras for real-time 3d hand gesture interaction. In: 2011 IEEE workshop on applications of computer vision (WACV), pp 66–72

6. Dominio F, Donadeo M, Zanuttigh P (2014) Combining multiple depth-based descriptors for hand gesture recognition. Pattern Recogn Lett 50:101–111
7. Ferreira PM, Cardoso JS, Rebelo A (2017) Multimodal learning for sign language recognition. In: Iberian conference on pattern recognition and image analysis, pp 313–321. Springer
8. Geng Y, Zhang G, Li W, Gu Y, Liang RZ, Liang G, Wang J, Wu Y, Patil N, Wang JY (2017) A novel image tag completion method based on convolutional neural transformation. In: Lintas A, Rovetta S, Verschure PF, Villa AE (eds) Artificial neural networks and machine learning – ICANN 2017. Springer International Publishing, Cham, pp 539–546
9. Hamid ATZ, Wirza RR, Iqbal SM, Suhaiza SP (2014) Skin segmentation using yuv and rgb color spaces. J Inf Process Syst 10(2):283
10. Huang C, Loy CC, Tang X (2016) Local similarity-aware deep feature embedding. In: Lee DD, Sugiyama M, Luxburg UV, Guyon I, Garnett R (eds) Advances in neural information processing systems 29, pp 1262–1270
11. Kurakin A, Zhang Z, Liu Z (2012) A real time system for dynamic hand gesture recognition with a depth sensor. In: 2012 Proceedings of the 20th European signal processing conference (EUSIPCO), pp 1975–1979
12. Lenz I, Lee H, Saxena A (2015) Deep learning for detecting robotic grasps. Int J Robot Res 34(4-5):705–724. https://doi.org/10.1177/0278364914549607
13. Liang R, Liang G, Li W, Li Q, Wang JJ (2016) Learning convolutional neural network to maximize pos@top performance measure. arXiv:1609.08417
14. Marin G, Dominio F, Zanuttigh P (2014) Hand gesture recognition with leap motion and kinect devices. In: 2014 IEEE International conference on image processing (ICIP), pp 1565–1569
15. Marin G, Dominio F, Zanuttigh P (2016) Hand gesture recognition with jointly calibrated leap motion and depth sensor. Multimedia Tools and Applications 75(22):14,991–15,015. https://doi.org/10.1007/s11042-015-2451-6
16. Ngiam J, Khosla A, Kim M, Nam J, Lee H, Ng AY (2011) Multimodal deep learning. In: International conference on machine learning (ICML), vol 6
17. Potter LE, Araullo J, Carter L (2013) The leap motion controller: a view on sign language. In: Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration, OzCHI '13. ACM, New York, pp 175–178
18. Ramachandram D, Taylor GW (2017) Deep multimodal learning: a survey on recent advances and trends. IEEE Signal Proc Mag 34(6):96–108. https://doi.org/10.1109/MSP.2017.2738401
19. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: The IEEE conference on computer vision and pattern recognition (CVPR)
20. Sohn K, Shang W, Lee H (2014) Improved multimodal deep learning with variation of information. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ (eds) Advances in neural information processing systems 27, pp 2141–2149. Curran Associates, Inc. http://papers.nips.cc/paper/5279-improved-multimodal-deep-learning-with-variation-of-information.pdf
21. Srinivas S, Sarvadevabhatla RK, Mopuri KR, Prabhu N, Kruthiventi S, Radhakrishnan VB (2016) A taxonomy of deep convolutional neural nets for computer vision. Frontiers in Robotics and AI 2(36):1–13
22. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958. http://jmlr.org/papers/v15/srivastava14a.html
23. Su F, Wang J (2018) Domain transfer convolutional attribute embedding. arXiv:1803.09733
24. Wang A, Cai J, Lu J, Cham TJ (2015) Mmss: Multi-modal sharable and specific feature learning for rgb-d object recognition. In: 2015 IEEE International conference on computer vision (ICCV), pp 1125–1133
25. Wang A, Lu J, Cai J, Cham TJ, Wang G (2015) Large-margin multi-modal deep learning for rgb-d object recognition. IEEE Trans Multimedia 17(11):1887–1898. https://doi.org/10.1109/TMM.2015.2476655
26. Wang J, Shi L, Wang H, Meng J, Wang JJ, Sun Q, Gu Y (2016) Optimizing top precision performance measure of content-based image retrieval by learning similarity function. arXiv:1604.06620
27. Wang JJY, Wang Y, Zhao S, Gao X (2015) Maximum mutual information regularized classification. Eng Appl Artif Intell 37:1–8. https://doi.org/10.1016/j.engappai.2014.08.009. http://www.sciencedirect.com/science/article/pii/S0952197614002085
28. Wu Z, Jiang YG, Wang J, Pu J, Xue X (2014) Exploring inter-feature and inter-class relationships with deep neural networks for video classification. In: Proceedings of the 22Nd ACM International conference on multimedia, MM '14. ACM, New York, pp 167–176. https://doi.org/10.1145/2647868.2654931. http://doi.acm.org/10.1145/2647868.2654931
29. Yang H (2015) Sign language recognition with the kinect sensor based on conditional random fields. Sensors 15(1):135–147. https://doi.org/10.3390/s150100135
30. Zhang G, Liang G, Li W, Fang J, Wang J, Geng Y, Wang JY (2017) Learning convolutional ranking-score function by query preference regularization. In: Yin H, Gao Y, Chen S, Wen Y, Cai G, Gu T, Du

J, Tallón-Ballesteros AJ, Zhang M (eds) Intelligent data engineering and automated learning – IDEAL 2017. Springer International Publishing, Cham, pp 1–8
31. Zhang S, Wang H, Huang W (2017) Two-stage plant species recognition by local mean clustering and weighted sparse representation classification. Clust Comput 20(2):1517–1525. https://doi.org/10.1007/s10586-017-0859-7