A Transformer-Based Network for Dynamic Hand Gesture Recognition

Andrea D'Eusanio[1], Alessandro Simoni[1], Stefano Pini[1], Guido Borghi[2], Roberto Vezzani[1], Rita Cucchiara[1]
[1]Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia (Italy)
[2]Department of Computer Science and Engineering, University of Bologna (Italy)
{s.pini, name.surname}@unimore.it, guido.borghi@unibo.it

Abstract

Transformer-based neural networks represent a successful self-attention mechanism that achieves state-of-the-art results in language understanding and sequence modeling. However, their application to visual data and, in particular, to the dynamic hand gesture recognition task has not yet been deeply investigated. In this paper, we propose a transformer-based architecture for the dynamic hand gesture recognition task. We show that the employment of a single active depth sensor, specifically the usage of depth maps and the surface normals estimated from them, achieves state-of-the-art results, overcoming all the methods available in the literature on two automotive datasets, namely NVidia Dynamic Hand Gesture and Briareo. Moreover, we test the method with other data types available with common RGB-D devices, such as infrared and color data. We also assess the performance in terms of inference time and number of parameters, showing that the proposed framework is suitable for an online in-car infotainment system.

1. Introduction

The recent introduction of affordable RGB-D devices, which couple RGB cameras with active depth sensors, has attracted the interest of the research community in *Natural User Interfaces* (NUIs), in which the interaction is conveyed through the body of the user [39, 30] instead of traditional tools, like keyboards and mouse. In this context, the ability to recognize dynamic hand gestures, *i.e*. a combination of static hand poses and motion, without the use of contact-based sensors is an enabling and crucial task. The hand gesture recognition task is commonly tackled through the use of RNNs [22, 29], such as LSTMs [50, 7], architectures that are able to model the temporal and sequential nature of dynamic gestures. Alternatively, authors have proposed to classify temporal sequences using 3D CNNs [51, 31], standard CNNs [15, 14] or other machine learning methods, like HMMs [28, 6] or HOG and SVM [38, 18].

The recent spread of attentive models, which are characterized by the use of self-attention mechanisms, has come with the introduction of new approaches, such as the *Transformer* [43], which can replace traditional recurrent modules, such as RNNs and LSTMs. However, these approaches have not yet been deeply explored for the analysis of visual data and, in particular, for the dynamic hand gesture recognition task.

In this paper, we propose a method to classify dynamic hand gestures based on the Transformer architecture, which was originally developed for the machine translation and language modeling tasks. We propose the use of RGB-D or active depth devices and, in particular, we show that the use of depth maps and the surface normals estimated from them leads to state-of-the-art results. In addition, we investigate the adoption of the other data streams usually provided by RGB-D sensors, *i.e*. infrared amplitude and color images, and derived data, such as optical flow.
The employment of light-invariant data sources – depth and infrared images – guarantees the applicability of the proposed method for a *Human-Computer Interaction* (HCI) system able to work even in presence of dramatic and fast light changes, as often occurs in the automotive setting [37]. Indeed, the presence of tunnels and trees or bad weather conditions can strongly influence the quality of the acquired data in this scenario. Moreover, the use of inexpensive and compact cameras, which can be easily integrated in the car cockpit, is an optimal choice in order to avoid obstructions to the driver's movements or gaze. It is shown [46, 13] that the presence of a NUI-based system for the interaction with the infotainment system of a car can significantly reduce the driver's manual and visual distraction [4, 5] often responsible for fatal road crashes.

For these reasons, the choice of datasets to test the proposed system is automotive-driven: we exploit two publicly released datasets, namely *NVidia Dynamic Hand Gesture* [33] and *Briareo* [31]. They are both acquired in a realistic car simulator through several acquisition sensors placed in different position inside the car cockpit, as detailed in Section 4.1. When tested on these datasets, the

proposed transformer-based architecture achieves state-of-the-art results, overcoming existing literature competitors. Moreover, the proposed method is flexible, since it can be adapted to the available data types and is able to run in real-time on a dedicated graphics card.
The proposed architecture is implemented in PyTorch 1.5 and the code is available online [1].

2. Related Work

In the literature, the hand gesture recognition task has been approached using different strategies which enable the temporal observation of an action performed by a human. However, recent architectures [42, 33, 8], which exploit the potential of 3D Convolution in extracting temporal features from videos, become milestones as an action recognition system.

As many tasks in the computer vision field, the hand gesture recognition task can rely on different types and combination of input data. Therefore, from a general point of view, methods available in the literature can be grouped as unimodal and multimodal.

In the unimodal case a single input (*e.g.* RGB, infrared, depth) is used at a time. Köpüklü et al. [25] adapt state-of-the-art architectures, *i.e.* C3D [42] and ResNet [21], in a lightweight framework composed of a detector, that detects the beginning and the end of a gesture, and the gesture classifier. Since 3D CNNs needs more training data due to the larger number of parameters with respect to 2D CNNs, the networks are pre-trained on one of the largest public hand gesture dataset, namely *Jester* [32], and then fine-tuned on other datasets. In [10] authors exploit 3D hand joints to reconstruct the hand skeleton and then perform the gesture classification capturing the motion and the hand shape through a video sequence. Unfortunately, their method gets quite low results on datasets without high-quality hand skeleton annotations. Finally, with the recent success of *self-attention* [43] in emulating the human visual perception, an attention-based network has been introduced by Dhingra et al. [12]. They use a 3D CNN model in which 3 attention blocks are positioned between the residual modules in order to learn features at different scales. Since they train their network from scratch, they obtain good results only on datasets with a large amount of training data.

In the multimodal setting two or more input types are exploited for the recognition task. In [35] authors propose a novel architecture that, exploiting RGB and depth data together with their computed optical flow (4 different data types), analyses the motion using a spatial focus attention, which restricts the focus on specific body parts (*e.g.* global, right hand, left hand). Having a total number of 12 features channels, they face the problem of gesture classification weighting each channel with respect to its importance to a specific gesture. A different multimodal approach [26] has been introduced by the same authors of [25]: in this case, they apply a data level fusion between an RGB frame and several optical flow images computed on previous frames. This information is given as input to a deep network that extracts spatio-temporal features on which is performed the gesture classification task with a fully connected network.
An inspiring work by Abavisani et al. [1] proposes a method that explores the performance of multimodal training and also its effects on unimodal testing. They fine-tune a pre-trained 3D CNN network [8] on multiple source data (*e.g.* RGB, depth, optical flow). An interesting aspect of this work is the introduction of a loss, namely *spatio-temporal semantic alignment*, which encourages the network to learn a common understanding on different data types.

Authors of [20, 27, 9] propose transformer-based approaches similar to ours in order to tackle the action and the sign language recognition tasks.
In [20], a slightly-modified version of the transformer architecture is used as part of an action localization and recognition framework, resembling the structure of Faster R-CNN. In [27], a transformer-like architecture is used in combination of a feature extractor to real-time action recognition. It makes use of 1D convolutional layers between sequential decoder blocks, but it does not use any kind of positional encoding thus the temporal relationships are not explicitly modeled. On the other hand, in our approach the temporal information about the frame order is encoded through the positional encodings (PE). Moreover, the method proposed in [27] is not developed for the usage with depth sensors and it does not propose the usage of surface normals as a different depth map representation.

3. Proposed Method

In this section, we present the mathematical formulation and the transformer-based implementation of our method. The proposed model can process an input sequence of variable length and outputs the gesture classification. An overall view of the architecture is represented in Figure 1.

3.1. Formulation

The proposed gesture recognition architecture can be defined as a function

$$\Gamma : \mathbb{R}^{m \times w \times h \times c} \rightarrow \mathbb{R}^{n} \tag{1}$$

that predicts a probability distribution over n classes from a set $S_t \in \mathbb{R}^{m \times w \times h \times c}$ of m sequential frames I, with size $w \times h$ and c channels, acquired in a time range t. In other words, the function Γ takes a sequence clip and predicts a class distribution over the considered hand gestures. The

[1] https://aimagelab.ing.unimore.it/go/gesture-recognition-automotive

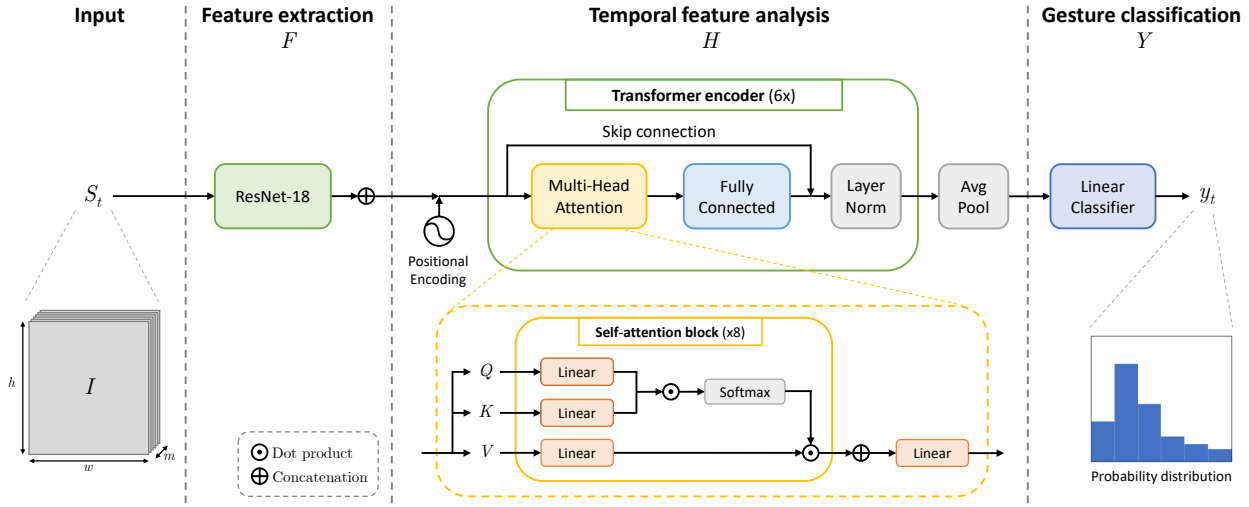

Figure 1. Overview of the proposed method. The temporal feature analysis, computed after the feature extraction performed by the ResNet-18 model, is highlighted showing the architecture of the transformer encoder and the self-attention block.

function can be decomposed in the following three components.

The first operation corresponds to a feature extraction function F applied at frame level:

$$\mathrm{f}_t = F(S_t) \quad \text{where} \quad F : \mathbb{R}^{m \times w \times h \times c} \rightarrow \mathbb{R}^{m \times k} \tag{2}$$

Here, the extracted features f_t consist of m independent visual features of size k. Therefore, the function F can be defined as the concatenation of the results of a frame-level feature extractor:

$$F(S_t) = \mathrm{f}_t^0 \oplus \mathrm{f}_t^1 \oplus \ldots \oplus \mathrm{f}_t^m \quad \text{where} \quad \mathrm{f}_t^j = G(S_t^j) \tag{3}$$

where $G : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^k$ is a function that extracts visual features from a single frame j of the sequence set S_t. $\oplus$ denotes the concatenation operator.

The second operation is a temporal combination and analysis of the visual features extracted through F. This process can be defined as

$$\mathrm{h}_t = H(\mathrm{f}_t) = H(F(S_t)) \quad \text{where} \quad H : \mathbb{R}^{m \times k} \rightarrow \mathbb{R}^l \tag{4}$$

where H is a temporal function that processes m feature maps of size k and outputs an aggregated feature map of size l which encodes the temporal information of S_t.

Finally, the last operation is a mapping between the extracted temporal features h_t and the n gesture classes:

$$y_t = Y(\mathrm{h}_t) = Y(H(F(S_t))) \quad \text{where} \quad Y : \mathbb{R}^l \rightarrow \mathbb{R}^n \tag{5}$$

The resulting y_t, being a probability distribution over n classes, is a vector of size n so that $\sum_{i=1}^{n} y_{t,i} = 1$ and $y_{t,i} \in [0, 1]$.

3.2. Implementation

In our implementation, the function Γ is a combination of multiple neural networks, defined as following.

The function F is the concatenation of the frame-level features extracted by the function G, which is implemented as *ResNet-18* [21], taken from the first layer up to the last convolutional and average pooling layers. The network is designed for color images, but we adapt the first layer to work with inputs having a lower number of channels c as proposed in [33]. In practice, the convolutional kernels of the first layer are adapted to 1-channel images by summing their channels. In a similar way, they are adapted to 2-channel images by removing the third channel and rescaling the first two with a factor of 1.5.

The function H, which has to temporally combine the frames of the clip S_t, corresponds to a slightly-modified Transformer module [43] followed by an average pooling at frame level. The model can handle sequences of any length and can be defined formally as

$$H(x) = \mathrm{AvgPool}(\mathrm{Encoders}(x + PE)) \tag{6}$$

where $\mathrm{AvgPool}(\cdot)$ denotes the average pooling operation over the m frames, while $\mathrm{Encoders}(\cdot)$ corresponds to a sequence of 6 transformer encoders E, defined in the following.

As detailed in [43], we add positional encodings PE to the input data as a way of including temporal information about the order of the frames into the model, which does not contain any recurrent module. Among the several positional encodings [19], we employ the proposal of [43].

Each transformer encoder can be defined as

$$E(x) = \mathrm{Norm}(x + \mathrm{FC}(\mathrm{mhAtt}(x))) \tag{7}$$

where Norm$(\cdot)$ is a normalization layer, FC$(\cdot)$ is a sequence of two fully connected layers with 1024 units, followed by drop out (drop probability 0.1) and divided by a ReLU activation function. The multi-head attention block mhAtt is a self-attention layer that can be defined as

$$\text{mhAtt}(x) = (\,\text{Att}_1(x) \oplus \ldots \oplus \text{Att}_8(x)\,)\,W^O \tag{8}$$

where

$$\text{Att}_i(x) = \text{softmax}\left(\frac{Q_i\,K_i}{\sqrt{d_k}}\right) V_i \tag{9}$$

Here, $Q_i = xW_i^Q$, $K_i = xW_i^K$, $V_i = xW_i^V$ are independent linear projections of x into a 64-d feature space, $d_k = 64$ is a scaling factor corresponding to the feature size of K_i, $\oplus$ is the concatenation operator and W^O is a linear projection from and to a 512-d feature space.

Finally, the function Y is implemented as a fully connected layer with n hidden units followed by a softmax layer, resulting in a probability distribution over the n classes. The predicted gesture corresponds to the class with the highest probability.

We note that the proposed approach is supposed to receive a sequence of frames containing the whole gesture or can applied with a sliding-window approach. The temporal segmentation, *i.e*. the detection of the beginning and the end of each gesture, and the gesture detection, *i.e*. the distinction between gesture and no-gesture sequences, are out of the scope of this paper.

3.3. Data Representation

As mentioned above, we focus our investigation on the use of data produced by active depth sensors, *i.e*. depth data and infrared (amplitude) images. We include also RGB data since several depth devices available in the market consist of a combination of infrared and intensity sensors, like the *Microsoft Kinect* or *Intel RealSense* families.

In addition, we propose the use of surface normals, in which each pixel encodes the three components of the estimated surface normal in that point. From depth maps we obtain a representation containing an estimation of the surface normals, as introduced in [3]. Given a depth map D, we define $Z(x, y)$ as one of its pixel values. We compute the direction $\mathbf{d} = \langle d_x, d_y, d_z \rangle$ of a surface normal as:

$$\mathbf{d} = \left(-\frac{\partial Z(x,y)}{\partial x}, -\frac{\partial Z(x,y)}{\partial y}, 1\right) \tag{10}$$

where $\partial Z(x, y)/\partial x$, $\partial Z(x, y)/\partial y$ can be considered the depth gradients in the x and y directions [34], or rather:

$$\begin{aligned} \frac{\partial Z(x,y)}{\partial x} &\approx Z(x+1, y) - Z(x, y) \\ \frac{\partial Z(x,y)}{\partial y} &\approx Z(x, y+1) - Z(x, y) \end{aligned} \tag{11}$$

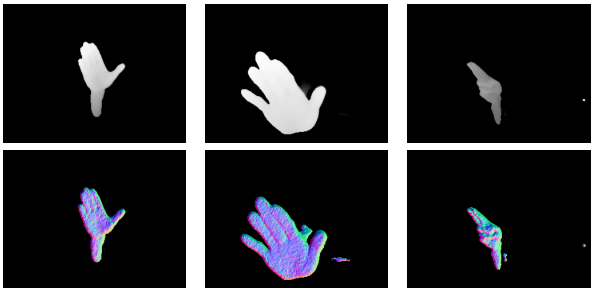

Figure 2. Sample depth (first row) and surface normals (last row) obtained from the *Nvidia Dynamic Hand Gesture* dataset. As shown, cameras are placed in a frontal position with respect to the driver and the noise level is low. Generally, in most of the frames, only the hand is visible.

Then, the normal vector $\hat{\mathbf{v}} = \langle \hat{v}_x, \hat{v}_y, \hat{v}_z \rangle$ is obtained through a normalization operation [2]:

$$\hat{\mathbf{v}} = \frac{1}{B}(d_x, d_y, 1), \;\; B = \sqrt{d_x^2 + d_y^2 + 1} \tag{12}$$

Normals computed from depth maps are not frequently used in the literature, especially in the case of the hand gesture recognition task with neural architectures. Preliminary work investigated the use of surface normals for hand pose estimation [44] or human activity recognition [49, 36]. We show in the following that this representation is complementary to the common depth images and that greatly improves the overall accuracy when used in combination with the original depth data.

In order to compare our work with literature competitors, we also compute the optical flow from consecutive RGB frames following the implementation of Farnebäck *et al*. [17]. It is a well-known data representation that is often used to improve the performance of the proposed system, even in the hand recognition task [33, 1], thanks to its ability to provide an estimation of the magnitude and the direction of the object (the hands in our case) motion.

3.4. Multimodal Integration

Multimodal architectures are becoming increasingly common in the literature, for a variety of different tasks. Since several input types are available from RGB-D sensors, we adopt a neural network architecture that can be easily adapted to work with a single input type or a multimodal combination of them. Specifically, the proposed architecture is able to efficiently work in a unimodal way, *i.e*. with a single input modality (color, depth, infrared, normals or optical flow). Moreover, two or more unimodal networks can be used at the same time through a late fusion approach [41] in which the predicted probability distributions of the single models are merged into a final classification score. Late fusion strategies are reported to present comparable or even better results with respect to the state-of-the-art in many

computer vision tasks [48, 16]. In our case, we adopt a late fusion strategy based on the average of the intermediate scores to predict the final classification, as follows:

$$y_t = \frac{1}{N} \cdot \sum_i Y(H(F(S_{t,i}))) \tag{13}$$

where N is the total number of tested classifiers, $S_{t,i}$ is the set of sequential frames of the i-th input type and F, H are the functions defined in Section 3.1. Then, $Y(H(F(S_{t,i})))$ is the probability distribution of a classifier trained and tested on a specific input type.

4. Experimental Evaluation

In this section, we present the experimental setting and the results obtained on two public datasets. Then, we compare with literature methods and discuss the obtained results. Since surface normals can be considered as a different representation of depth maps, we include competitors relying on RGB-D data.
In addition to the core tests with depth images and estimated surface normals, we test on color and other modalities to compare with existing literature methods.

4.1. Datasets

Being interested in the usage of depth or RGB-D sensors and in the automotive environment, in which the light invariance is a key factor, we test our approach on two datasets, *NVGestures* [33] and *Briareo* [31], collected in a car simulator.

Nvidia Dynamic Hand Gesture. This dataset [33], also called NVGestures, is the largest dynamic hand gesture dataset in an automotive setting, in terms of number of gestures, subjects and sequences. Video sequences are acquired with two sensors: the *SoftKinetic DS325*, an active RGB-D sensor, and the *DUO 3D*, an infrared stereo camera. These acquisition devices lead to 3 modalities (RGB, depth, IR) and 5 streams (color, depth, color mapped on depth, IR left, IR right), available in the dataset. NVGestures is acquired in an indoor car simulator, the depth camera is placed next to the infotainment system, while the stereo camera is placed on top of the acquisition area. Authors did not release the infrared amplitude recorded by the depth sensor, but they provided infrared data from the dedicated *DUO 3D* camera, placed in a different position. The dataset contains 25 different gestures performed by 20 subjects with the right hand. Each gesture is repeated three times and acquired in 5-second video samples. Gestures range from swipes to rotations and from showing n fingers to showing the "OK" sign. For further details about this dataset, please refer to the original paper [33].
In our experiments, we employ the color (RGB), depth, and infrared (left IR) modalities. In addition, we compute an estimation of the surface normals from the depth data (see Section 3.3) and we report visual samples of these data in Figure 2. In order to compare with literature work, we compute the optical flow on color frames through [17], as done in previous work [33].

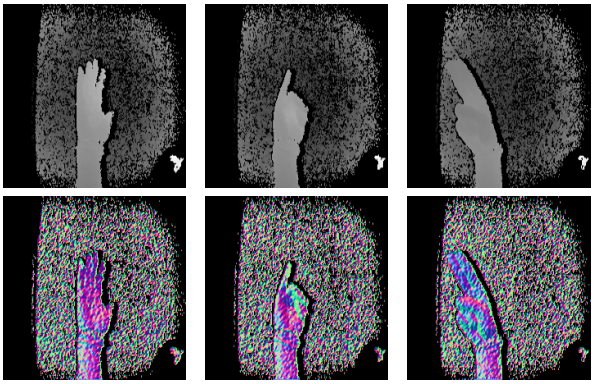

Figure 3. Sample depth (first row) and surface normals (last row) obtained from the *Briareo* dataset. Differently from the dataset from Nvidia, this dataset is acquired placing the camera looking upwards. Moreover, a strong noise signal is present in depth and, consequently, in surface normals.

Briareo. This is a recently-released automotive dataset [31] for the dynamic hand gesture recognition task. Video sequences are acquired using three synchronized devices: an active depth sensor (*Pico Flexx*), an infrared stereo sensor (*Leap Motion*) and a standard RGB camera. Therefore, several image types are available: depth and infrared amplitude, left and right IR, color. In addition, the SDK of the Leap Motion device has been used to estimate and record the hand joint positions. Recording devices are placed in the central tunnel console of a car simulator between the driver and the passenger seat, looking upwards. In this case, authors released the infrared amplitude recorded by the depth sensor, along with the infrared data acquired by the Leap Motion sensor. The dataset contains 12 different gestures performed by 40 different subjects (33 males and 7 females) with the right hand. As in NVGestures, each gesture is repeated three times and captures the entire gesture motion. Gestures are designed for the interaction between the driver and the car infotainment system. Some examples are the swipes in the four directions and the "thumb up" and "phone" signs. For further details about this dataset, please refer to the original paper [31].
In our experiments, we employ most of the available modalities, *i.e.* color (RGB), depth, and infrared amplitude. In addition, we estimate the surface normals from the depth data, as explained in Section 3.3 and depicted in Figure 3.

4.2. Model Training

We train and test the model with fixed-length clips of 40 frames extracted from the dataset sequences around the

center of the gesture. We empirically set this input size, but the proposed model can potentially analyze sequences of any length thanks to its flexible design. For the NVGestures dataset, we extract the 80 central frames around the gesture and sample them to obtain 40 equidistant frames. For the Briareo dataset, which has a lower frame rate, we select the 40 frames containing the gesture movement.

Each input data is normalized individually to obtain zero mean and unit variance input, with the exception of the surface normals that are normalized to have unit-magnitude and are contained in the range $[-1, 1]$. Then, frames are cropped to 224×224 pixels as required by the chosen frame-level feature extractor (*i.e.* ResNet-18). We apply random rescale (with rescale factor in the range $[0.8, 1.2]$), random crop and random rotation between -15 and 15 degrees as data augmentation, in order to avoid overfitting.

The ResNet-18 architecture is initialized with weights pre-trained on ImageNet [11] while the remaining of the architecture is trained from scratch. The architecture is then trained end-to-end using the Adam optimizer [24] to minimize the categorical cross entropy loss. We use a mini-batch size of 8 video samples, learning rate $1e^{-4}$, weight decay $1e^{-4}$ and random dropout. We apply the early stopping based on the accuracy on the validation set, following the official dataset splits.

A different model is trained for each modality and multiple modalities are combined at prediction level with the late fusion approach presented in Section 3.4. Empirically, we find that other types of fusion, *e.g.* mid and early fusion, results in overfitting on the training set, in line with what found in [33].

4.3. Results using NVGestures dataset

We analyze here the performance on the NVGestures dataset.

Table 1 compares our method to the literature in the unimodal case, *i.e.* when a single input is fed into the model. Focusing on depth data, the proposed approach achieves state-of-the-art results when depth maps are the only used input. A similar high accuracy is also achieved using surface normals as input, revealing that normals are a discriminative representation for the hand gesture recognition task, even though no competitors are currently available. Also the infrared modality overcomes the competitor, even if the final accuracy is lower. On the other remaining modalities, *i.e.* color and optical flow, our method achieves comparable accuracy to the I3D method [8, 1]. However, we note this method is pre-trained on ImageNet [11] (as our feature extractor) and on Kinetics [23], which is a large dataset of action recognition in videos. We hypothesize that the slight gap between this and our method can be due to this pre-training step, which was not available for the other types of the exploited data.

Method	Modality	Accuracy
color	Spat. st. CNN [40]	54.6%
	iDT-HOG [45]	59.1%
	Res3ATN [12]	62.7%
	C3D [42]	69.3%
	R3D-CNN [33]	74.1%
	Ours	**76.5%**
	I3D [8]†	78.4%
depth	SNV [47]	70.7%
	C3D [42]	78.8%
	R3D-CNN [33]	80.3%
	I3D [8]†	82.3%
	Ours	**83.0%**
infrared	R3D-CNN [33]	63.5%
	Ours	**64.7%**
flow	iDT-HOF [45]	61.8%
	Temp. st. CNN [40]	68.0%
	Ours	72.0%
	iDT-MBH [45]	76.8%
	R3D-CNN [33]	**77.8%**
	I3D [8]†	83.4%
normals	**Ours**	**82.4%**
color	*Human [33]*	*88.4%*

Table 1. Unimodal results on NVGestures [33]. Previous results are taken from the respective papers and from [33, 1]. † indicates models pre-trained on Kinetics [23], in addition to ImageNet [11].

Moving from the unimodal to the multimodal case, we show in Table 2 a thorough analysis of the possible multimodal combinations, following the late-fusion approach reported in Section 3.4. The results are grouped by number of employed modalities and ordered by accuracy. It can be seen that, in general, the proposed approach benefits from the multimodal integration. Moreover, the best performing methods in each group are those using a combination of depth and surface normals as input data, confirming that the partial 3D data obtained by the depth sensors contains discriminative information for the gesture recognition task. We highlight that the combination of depth images and surface normals leads to a remarkable accuracy of 87.3%. This result confirms that these two modalities are complementary and their combination greatly improves the overall accuracy compared to the usage of a single modality (which scores 83.0 for the depth and 82.4 for the surface normals). Combining additional modalities (color and infrared) the accuracy is slightly incremented, reaching 87.6%.

We also compare our method in the multimodal setting with state-of-the-art approaches reported in Table 3. Among other methods that exploit several data types, our approach

#	Modality	Accuracy
1	infrared (ir)	64.7%
	color	76.5%
	normals	82.4%
	depth	83.0%
2	color + ir	79.0%
	depth + ir	81.7%
	normals + ir	82.8%
	color + depth	84.6%
	color + normals	84.6%
	depth + normals	**87.3%**
3	color + ir + depth	85.3%
	color + ir + normals	85.3%
	color + depth + normals	86.1%
	depth + normals + ir	**87.1%**
4	**color + depth + normals + ir**	**87.6%**

Table 2. Multimodal results on NVGestures [33] using several combinations of modalities. # refers to the number of used modalities.

obtains state-of-the-art accuracy (87.3%) using only depth data and surface normals, which derive from a single depth sensor. Therefore, the whole system can depend from a single depth or RGB-D device and can run in real time, as will be shown in Section 4.5. In addition, our method, combining a broader set of modalities (*i.e.* color, depth, surface normals, infrared), slightly improves the overall accuracy, reaching a 87.6 recognition rate.

A wide set of other methods make use of the optical flow, but still perform worse than our method. However, we note that the computation of the optical flow on the whole sequence of frames heavily affects speed performance, hindering the achievement of real time computation.

Finally, we show the confusion matrix for the best performing multimodal combination (*i.e.* color + depth + normals + ir) in Figure 4. Most of the gestures are correctly classified, but some errors caused by confusion between pairs of gestures are also visible. As expected, the model sometimes swaps similar – in terms of hand poses or motion – gestures, such as "move hand/fingers left/right", "opening" and "shaking" hand and "push hand down/towards the camera" .

4.4. Results using Briareo dataset

Table 4 presents the results of the unimodal and the multimodal setting for the Briareo dataset. The results are grouped by number of employed modalities and ordered by accuracy.

Considering the unimodal case, the surface normals obtains the highest accuracy, reaching 95.8%, outperforming the results using other modalities. This confirms that surface normals estimated from depth are an informative and discriminative representation for the hand gesture recognition task. Also the infrared source achieves a high accuracy, probably

Method	Modality	Accuracy
Two-st. CNNs [40]	color + flow	65.6%
iDT [45]	color + flow	73.4%
R3D-CNN [33]	color + flow	79.3%
R3D-CNN [33]	color + depth + flow	81.5%
R3D-CNN [33]	color + depth + ir	82.0%
R3D-CNN [33]	depth + flow	82.4%
R3D-CNN [33]	all	83.8%
8-MFFs-3f1c [26]*	color + flow	84.7%
I3D [8]†	color + depth	83.8%
I3D [8]†	color + flow	84.4%
I3D [8]†	color + depth + flow	85.7%
$\text{MTUT}_{\text{RGB-D}}$ [1]†	color + depth	85.5%
$\text{MTUT}_{\text{RGB-D+flow}}$ [1]†	color + depth	86.1%
$\text{MTUT}_{\text{RGB-D+flow}}$ [1]†	color + depth + flow	86.9%
Ours	**depth + normals**	**87.3%**
Ours	**color+depth+normals+ir**	**87.6%**
Human [33]	*color*	*88.4%*

Table 3. Multimodal results on NVGestures [33], comparison with competitors. Previous results are taken from the respective papers and from [33, 1]. † indicates models pre-trained on Kinetics [23], in addition to ImageNet [11], while * shows models pre-trained on the Jester gesture dataset [32].

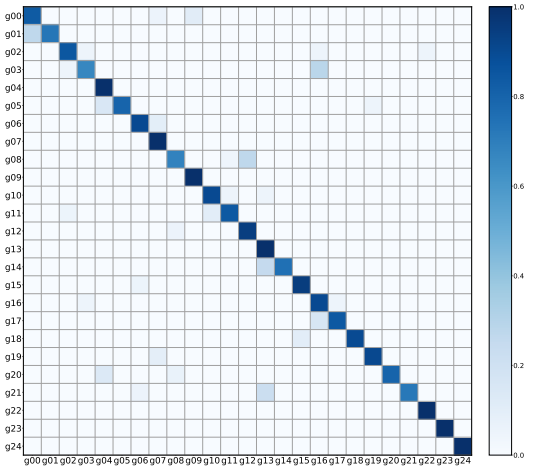

Figure 4. Confusion matrix for the best performing multimodal combination (fusion of color, depth, normals, ir) on NVGestures. Best viewed in color.

#	Modality	Accuracy
1	color	90.6%
	depth	92.4%
	ir	95.1%
	normals	**95.8%**
2	color + depth	94.1%
	depth + ir	95.1%
	color + ir	95.5%
	depth + normals	96.2%
	color + normals	96.5%
	ir + normals	**97.2%**
3	color + depth + ir	95.1%
	color + depth + normals	95.8%
	color + ir + normals	96.9%
	depth + ir + normals	**97.2%**
4	**color + depth + ir + normals**	**96.2%**

Table 4. Unimodal and multimodal results obtained on Briareo. # refers to the number of used modalities.

due to the position of the infrared sensor, close to the hand. The combination of multiple modalities, with the late fusion approach presented in Section 3.4, slightly improves the overall results. The fusion of infrared and normals results in an overall accuracy of 97.2% which is the highest result. While the combination of surface normals with infrared and depth increases the combined accuracy, the usage of color data does not provide significant gains.

In Table 5 we compare our method in the multimodal setting with state-of-the-art approaches. The proposed approach obtains state-of-the-art accuracy 97.2% using only infrared data and surface normals, which derive from a single active depth sensor. Even with the usage of a single modality, *e.g.* surface normals, our method outperforms the literature competitors by a clear margin. Indeed, it performs better than methods based on recurrent networks (LSTMs) and 3D joint features (computed by the Leap Motion SDK), which require additional computation. Also in this case, the whole system requires a single active depth device and can run in real time, as shown in the next section.

4.5. Performance Analysis

We assess the computational requirements of our and other architectures in terms of number of parameters, inference time on a single GPU, and required VRAM on the graphics card. We test them on a workstation with an *Intel Core* i7-7700K and a *Nvidia GeForce GTX* 1080 Ti. As shown in Table 6, our method has fewer parameters, faster inference speed and comparable memory usage when used with a single modality. When applied on multiple modalities, running in parallel on the same hardware, the proposed approach still maintains real time speed and acceptable memory usage, both in case of 2 modalities and in case of 4 modalities.

Method	Modality	Accuracy
C3D-HG [31]	color	72.2%
C3D-HG [31]	depth	76.0%
C3D-HG [31]	ir	87.5%
LSTM-HG [31]	3D joint features	94.4%
Ours	normals	95.8%
Ours	depth + normals	96.2%
Ours	**ir + normals**	**97.2%**

Table 5. Comparison with the state-of-the-art methods tested on Briareo.

Model	Parameters (M)	Inference (ms)	VRAM (GB)
R3D-CNN [33]	38.0	30	1.3
C3D-HG [31]	26.7	55	1.0
Ours (1 modality)	24.3	26.7	1.8
Ours (2 modalities)	48.6	61.7	3.0
Ours (4 modalities)	97.2	108.3	5.3

Table 6. Performance analysis of the proposed method. Specifically, we report the number of parameters, the inference time and the amount of video RAM (VRAM) needed to run the system.

5. Conclusions

In this paper, we propose a transformer-based architecture for the dynamic hand gesture recognition task. Through an extensive evaluation we show how the frame-level feature extraction and the temporal aggregation computed by the transformer, starting from depth and surface normals combined through a late fusion approach, achieves state-of-the-art results. Moreover, we investigate the use of other data types usually provided by RGB-D sensors, such as color and infrared images. Experimental results obtained on two automotive datasets, namely NVidia Dynamic Hand Gesture and Briareo, confirm the feasibility of the proposed method for the automotive setting, in which the light invariance is an enabling element. Even though the temporal flow is explicitly encoded into the transformer-based architecture, there are several "symmetric" gestures that are occasionally confused. In fact, the main challenges of the problem are still related to the temporal progression of the gesture, which will be addressed in future work. The performance analysis shows that the framework is able to run with real time performance and it requires a limited amount of video memory, making it suitable for an online infotainment system.

References

[1] M. Abavisani, H. R. V. Joze, and V. M. Patel. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1165–1174, 2019. 2, 4, 6, 7

[2] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2014. 4

[3] P. J. Besl and R. C. Jain. Invariant surface characteristics for 3d object recognition in range images. *Computer vision, graphics, and image processing*, 33(1):33–80, 1986. 4

[4] G. Borghi, R. Gasparini, R. Vezzani, and R. Cucchiara. Embedded recurrent network for head pose estimation in car. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1503–1508. IEEE, 2017. 1

[5] G. Borghi, S. Pini, R. Vezzani, and R. Cucchiara. Mercury: a vision-based framework for driver monitoring. In *International Conference on Intelligent Human Systems Integration*, pages 104–110. Springer, 2020. 1

[6] G. Borghi, R. Vezzani, and R. Cucchiara. Fast gesture recognition with multiple stream discrete hmms on 3d skeletons. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 997–1002. IEEE, 2016. 1

[7] F. M. Caputo, S. Burato, G. Pavan, T. Voillemin, H. Wannous, J.-P. Vandeborre, M. Maghoumi, E. Taranta, A. Razmjoo, J. LaViola Jr, et al. Online gesture recognition. In *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association, 2019. 1

[8] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6299–6308, 2017. 2, 6, 7

[9] M. De Coster, M. Van Herreweghe, and J. Dambre. Sign language recognition with transformer networks. In *12th Int. Conf. on Language Resources and Evaluation*, 2020. 2

[10] Q. De Smedt, H. Wannous, and J.-P. Vandeborre. Heterogeneous hand gesture recognition using 3d dynamic skeletal data. *Computer Vision and Image Understanding*, 181:60–72, 2019. 2

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6, 7

[12] N. Dhingra and A. Kunz. Res3ATN - deep 3d residual attention network for hand gesture recognition in videos. In *2019 International Conference on 3D Vision (3DV)*, pages 491–501. IEEE, 2019. 2, 6

[13] Y. Dong, Z. Hu, K. Uchimura, and N. Murayama. Driver inattention monitoring system for intelligent vehicles: A review. *IEEE transactions on intelligent transportation systems*, 12(2):596–614, 2011. 1

[14] A. D'Eusanio, A. Simoni, S. Pini, G. Borghi, R. Vezzani, and R. Cucchiara. Multimodal hand gesture classification for the human–car interaction. In *Informatics*, volume 7, page 31. Multidisciplinary Digital Publishing Institute, 2020. 1

[15] A. Elboushaki, R. Hannane, K. Afdel, and L. Koutti. Multid-cnn: A multi-dimensional feature learning approach based on deep convolutional networks for gesture recognition in rgb-d image sequences. *Expert Systems with Applications*, 139:112829, 2020. 1

[16] H. Ergun, Y. C. Akyuz, M. Sert, and J. Liu. Early and late level fusion of deep convolutional neural networks for visual concept recognition. *International Journal of Semantic Computing*, 10(03):379–397, 2016. 5

[17] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003. 4, 5

[18] K.-p. Feng and F. Yuan. Static hand gesture recognition based on hog characters and support vector machines. In *2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)*, pages 936–938. IEEE, 2013. 1

[19] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 1243–1252, 2017. 3

[20] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman. Video action transformer network. In *CVPR*, 2019. 2

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 3

[22] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng. A novel attention-based hybrid cnn-rnn architecture for semg-based gesture recognition. *PloS one*, 13(10):e0206049, 2018. 1

[23] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6, 7

[24] D. P. Kingma and J. A. Ba. A method for stochastic optimization. arxiv 2014. *arXiv preprint arXiv:1412.6980*, 434, 2019. 6

[25] O. Köpüklü, A. Gunduz, N. Kose, and G. Rigoll. Real-time hand gesture detection and classification using convolutional neural networks. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8. IEEE, 2019. 2

[26] O. Kopuklu, N. Kose, and G. Rigoll. Motion fused frames: Data level fusion strategy for hand gesture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2103–2111, 2018. 2, 7

[27] A. Kozlov, V. Andronov, and Y. Gritsenko. Lightweight network architecture for real-time action recognition. In *35th Annual ACM Symposium on Applied Computing*, 2020. 2

[28] H.-K. Lee and J.-H. Kim. An hmm-based threshold model approach for gesture recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 21(10):961–973, 1999. 1

[29] G. Lefebvre, S. Berlemont, F. Mamalet, and C. Garcia. Blstm-rnn based 3d gesture classification. In *International*

conference on artificial neural networks, pages 381–388. Springer, 2013. 1

[30] W. Liu. Natural user interface-next mainstream product user interface. In *2010 IEEE 11th International Conference on Computer-Aided Industrial Design & Conceptual Design 1*, volume 1, pages 203–205. IEEE, 2010. 1

[31] F. Manganaro, S. Pini, G. Borghi, R. Vezzani, and R. Cucchiara. Hand gestures for the human-car interaction: the briareo dataset. In *International Conference on Image Analysis and Processing (ICIAP)*, pages 560–571. Springer, 2019. 1, 5, 8

[32] J. Materzynska, G. Berger, I. Bax, and R. Memisevic. The jester dataset: A large-scale video dataset of human gestures. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2, 7

[33] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz. Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4207–4215, 2016. 1, 2, 3, 4, 5, 6, 7, 8

[34] Y. Nakagawa, H. Uchiyama, H. Nagahara, and R.-I. Taniguchi. Estimating surface normals with depth image gradients for fast and accurate registration. In *2015 International Conference on 3D Vision (3DV)*, pages 640–647. IEEE, 2015. 4

[35] P. Narayana, R. Beveridge, and B. A. Draper. Gesture recognition: Focus on the hands. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5235–5244, 2018. 2

[36] X. S. Nguyen, T. P. Nguyen, and F. Charpillet. Effective surface normals based action recognition in depth images. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 817–822. IEEE, 2016. 4

[37] S. Pini, A. D'Eusanio, G. Borghi, R. Vezzani, and R. Cucchiara. Baracca: a multimodal dataset for anthropometric measurements in automotive. In *International Joint Conference on Biometrics (IJCB)*, 2020. 1

[38] Y. Ren and C. Gu. Hand gesture recognition based on hog characters and svm. *Bulletin of Science and Technology*, 2:011, 2011. 1

[39] E. N. Saba, E. C. Larson, and S. N. Patel. Dante vision: In-air and touch gesture sensing for natural surface interaction with combined depth and thermal cameras. In *2012 IEEE International Conference on Emerging Signal Processing Applications*, pages 167–170. IEEE, 2012. 1

[40] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems (NIPS)*, pages 568–576, 2014. 6, 7

[41] C. G. Snoek, M. Worring, and A. W. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 399–402, 2005. 4

[42] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 4489–4497, 2015. 2, 6

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems (NIPS)*, pages 5998–6008, 2017. 1, 2, 3

[44] C. Wan, A. Yao, and L. Van Gool. Hand pose estimation from local surface normals. In *European conference on computer vision*, pages 554–569. Springer, 2016. 4

[45] H. Wang, D. Oneata, J. Verbeek, and C. Schmid. A robust and efficient video representation for action recognition. *International journal of computer vision (IJCV)*, 119(3):219–238, 2016. 6, 7

[46] F. A. Wilson and J. P. Stimpson. Trends in fatalities from distracted driving in the united states, 1999 to 2008. *American journal of public health*, 100(11):2213–2219, 2010. 1

[47] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 804–811, 2014. 6

[48] Y. Yang, J. Song, Z. Huang, Z. Ma, N. Sebe, and A. G. Hauptmann. Multi-feature fusion via hierarchical regression for multimedia analysis. *IEEE Transactions on Multimedia*, 15(3):572–581, 2012. 5

[49] C. Zhang, X. Yang, and Y. Tian. Histogram of 3d facets: A characteristic descriptor for hand gesture recognition. In *2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, pages 1–8. IEEE, 2013. 4

[50] L. Zhang, G. Zhu, L. Mei, P. Shen, S. A. A. Shah, and M. Bennamoun. Attention in convolutional lstm for gesture recognition. In *Advances in Neural Information Processing Systems*, pages 1953–1962, 2018. 1

[51] G. Zhu, L. Zhang, P. Shen, and J. Song. Multimodal gesture recognition using 3-d convolution and convolutional lstm. *Ieee Access*, 5:4517–4524, 2017. 1