# Action Recognition from RGB-D Data: Comparison and fusion of spatio-temporal handcrafted features and deep strategies

Maryam Asadi-Aghbolaghi
Dept. of Computer Engineering
Sharif University of Tech., Iran
masadia@ce.sharif.edu

Hugo Bertiche
University of Barcelona
Barcelona, Spain
hugo_bertiche@hotmail.com

Vicent Roig
University of Barcelona
Barcelona, Spain
vicent.roig.ripoll@gmail.com

Shohreh Kasaei
Dept. of Computer Engineering
Sharif University of Tech., Iran
skasaei@sharif.edu

Sergio Escalera
University of Barcelona and
Computer Vision Center, Spain
sergio@maia.ub.es

## Abstract

*In this work, multimodal fusion of RGB-D data are analyzed for action recognition by using scene flow as early fusion and integrating the results of all modalities in a late fusion fashion. Recently, there is a migration from traditional handcrafting to deep learning. However, handcrafted features are still widely used owing to their high performance and low computational complexity. In this research,* Multimodal dense trajectories *(MMDT) is proposed to describe RGB-D videos. Dense trajectories are pruned based on scene flow data. Besides, 2DCNN is extended to multimodal (MM2DCNN) by adding one more stream (scene flow) as input and then fusing the output of all models. We evaluate and compare the results from each modality and their fusion on two action datasets. The experimental result shows that the new representation improves the accuracy. Furthermore, the fusion of handcrafted and learning-based features shows a boost in the final performance, achieving state of the art results.*

## 1. Introduction

In the last few years human action recognition has been an active research area in computer vision due to its potential applications, including health-care monitoring[1], interactive gaming [2], surveillance [3], and robotics [4], just to mention a few. In past decades, research on human action recognition has been extensively explored on RGB data. The recent advances in imaging devices, and in particular Microsoft Kinect, have facilitated the capturing of low-cost and high sample rate depth images in real-time alongside color images. Depth information complements the conventional RGB cameras by providing partial 3D information of the scene. Therefore, fusing these multimodal information into highly discriminative feature sets can lead methods to achieve higher levels of performance.

Many approaches [5, 6] have demonstrated that late fusion of both RGB and depth modalities is effective for action recognition. Moreover, motion-based representations on the basis of optical flow analysis have been provided the state of the art results for several years [7, 8]. Compared to optical flow, which is the projected motion onto the 2D image plane, scene flow [9] is the real 3D motion of objects that move completely or partially with respect to a camera. Scene flow can record motions in real 3D world while optical flow can only capture information in image plane. Therefore, whenever there is a significant motion perpendicular to the image plane, scene flow can be more discriminative than optical flow. Scene flow can be considered as a kind of early fusion which preserve 3D motion information from the spatial structure of both RGB and depth modalities.

Recent progress on human action recognition mainly relies on designing an efficient and robust video representation which can be broadly categorized into two classes: handcrafted representation and learning-based features. Recently, learning-based feature representations have received great attention from action recognition researchers. However, handcrafted approaches are still widely used owing to their high performance and low computational complexity.

Traditional handcrafted representation approaches can be decomposed into: 1) detectors which discover informative regions for action recognition and 2) descriptors which describe the visual pattern of the detected regions. Among various handcrafted feature schemes proposed for action recognition so far, *dense trajectory* (DT) [10] and *improved dense trajectory* (iDT) [7] have become very popular.

Unlike handcrafted approaches, deep-based methods automatically learn features from raw data by utilizing a trainable feature extractor followed by a trainable classifier. In [11, 12], deep architectures used for action recognition are categorized in four groups: 2D models, motion-based input features, 3D models, and temporal networks.

Here, we focus on comparing the performance of handcrafted and deep learning features for multimodal human action recognition. In this work, DT from handcrafted features and *2D convolutional neural network* (2DCNN) have been extended by using four modalities; i.e, RGB, depth, optical flow, and scene flow.

In more detail, we present MMDT by exploiting scene flow. In MMDT, as the detection part, dense trajectories are pruned by exploiting scene flow information. Moreover, we use *histogram of normal vector* (HON) by extracting normal vectors of depth images. We also evaluate the incorporation of scene flow information in deep learning action recognition systems. Each modality is trained separately by 2DCNN, and final classification is done by score averaging.

Furthermore, we evaluate the accuracy of the combination of both handcrafted and deep learning representations as the second level of fusion. Each of them has its own benefits. Handcrafted features are more powerful in describing motion information while deep learning-based representations are quiet good at describing appearance data. The experimental results show that fusing the information from different modalities can boost the accuracy rather than using just one modality. Furthermore, the second level of fusion also improves final recognition performance, achieving state of the art results on two public available datasets.

The contributions of this paper are summarized as follows: a) To reduce the effect of noise in depth images, a simple yet efficient depth image denoising and multimodal registration are first applied; b) A comprehensive study is provided for video representation from handcrafted features and learning-based one. A framework is introduced to integrate the output of handcrafted and learning-based features; c) We exploit scene flow as a good source of discriminative data to extend DT for multimodal data; d) Instead of random frame selection, and to get a complete coverage of video and providing high-level semantic information, keyframes are extracted as relevant visual information to discriminate actions; d) Different kinds of input data (RGB, scene flow, and optical flow) are analyzed for deep models; e) Finally, we study the effect of late fusion of the class membership probabilities of two methods for final classification, and achieve state of the art results on two public RGB-D action datasets.

The remainder of this paper is organized as follows. Section 2 reviews related work. Proposed methodology is introduced in section 3. Experimental results on two public available datasets are presented in section 4. Finally, section 5 concludes the paper.

## 2. Related Work

Literature of vision based human action recognition can be divided into two main groups: handcrafted based methods and deep learning based approaches. In this section, we review important literature related to handcrafted and deep methods for action recognition in image sequences.

### 2.1. Handcrafted Methods

Many video representations in action recognition are the spatio-temporal extension of classical descriptors in image recognition to the temporal dimension [13, 14, 15, 16]. [13] proposed 3D-SIFT which is the 3D extension of SIFT descriptor to include temporal dimension. This descriptor is invariant to orientation (temporally as well), which presumably can better generalize the underlying information to discriminate actions. Similarly, in [14], authors introduced the idea of HOG3D.

[15] proposed HON4D descriptor for depth data. In this work, *histogram of oriented normals* (HON) is extended to the temporal dimension based on the distribution of 4D normal vectors in some spatio-temporal cells around a region of interest while performing an action. Authors in [16] proposed *supervised spatio-temporal kernel descriptor* (SSTKDes) for recognizing human actions as the extended version of *supervised kernel descriptor* (SKDES) [17].

Motion features like optical flow are very successful on action recognition, since they have local temporal informationIn [18], authors introduced *histogram of oriented flow* (HOOF). In [19] it is proposed the *motion boundary histograms* (MBH) by using the second order of optical flow.

Using trajectories which consider longer temporal information is the main idea of some successful the approaches. In [10, 19, 7], authors propose the use of optical flow for trajectory construction, and descriptors are computed around representative trajectories. More details about this technique are reviewed in Section 3.2.1.

Similarly, scene flow [20, 21, 9] is introduced for RGB-depth data as the actual 3D motion field in real 3D world. Most of the existing methods for calculating scene flow are based on stereo or multiple view camera systems [20, 21]. These methods suffered from a high computational cost. Jaimez et al. in [9] proposed the first dense real-time scene flow algorithm for RGB-D cameras. It is an iterative solver which performs pixel-wise updates and can be efficiently implemented on modern GPUs.

As optical flow measures motion in pixels, the number and length of the trajectories are directly influenced by the distance to the camera. Thus, further objects from camera have smaller size in pixels. Using depth images it is possible to extract scene flow (3D motion field), which is measured in meters, and therefore, having trajectories invariant to camera distance. Besides, as scene flow has an additional dimension (z-axis, or depth direction), one can track motion

in this direction as well, dealing with the situation of having a dominant motion around this axis. In this paper, we use this method for computing scene flow, which is then used to prune the dense trajectories obtained by RGB. It is worth mentioning that an extended version of DT has been proposed as iDT [7] in which the camera motion is removed from trajectories. The camera of all used data in this research is fixed, hence DT is selected for this work due to its simpler computation.

### 2.2. Deep Learning Models

Dealing with the temporal dimension of the data is the most crucial challenge for deep learning-based human action recognition [22]. Based on the way it is dealt with, a survey [11] categorized deep models in four groups, i.e. 2D models, Motion-based input features, 3D models, and Temporal methods.

In the first group, [23, 24] use a pre-trained model on one or more frames which are sampled from the whole video. Then, the entire video is labeled by averaging the result of the sampled frames. To consider temporal information, in the second category, [25, 8] compute 2D motion features like optical flow. Afterwards, these features are exploited as different input channels of a 2D network. The third group introduces 3D filters in the convolutional and pooling layers to learn discriminative features along both spatial and temporal dimensions [26, 27]. The input data of these networks are a fixed length sequence of frames. Finally in the fourth category, temporal sequence modeling tools like recurrent neural network (RNN) [28] are utilized to process temporal information. RNN models suffered from short memory. To solve this problem, *Long Short-Term Memory* (LSTM) [29] is added as a hidden layer of RNN [30, 31].

Among previous methods, 2DCNN [8] and its extensions [32, 33] have achieved state of the art results on RGB datasets. Therefore, these methods have been selected in this paper to be extended and analyzed for RGB-D data. In the work of [8], authors present two-stream convolutional neural network (CNN) which incorporates both spatial (video frames) and temporal networks. Individual frames are utilized as the input data of the spatial network which is fine-tuned from a pre-trained network on ImageNet [34]. The temporal model is trained by using stacked frames of optical flow as input. For the test mode, 25 frames are randomly selected from the video and each of them is labeled by both networks. Then score averaging is used to classify the whole video.

In this paper, different modalities are used as the input data for deep learning-based method. In 2DCNN instead of random selection of frames from the input sequence, we evaluate the effect of extracting keyframes. Finally, as the classification part, the score averaging is used which improved the accuracy of the model.

## 3. RGB-D action recognition analysis

The main aim of this paper is to compare the performance of handcrafted and deep learning features for action recognition from multimodal data. To this end, we extend two state of the art methods, i.e. DT and 2DCNN,which were originally proposed for RGB data. In order to reduce the effect of noise in depth images, a denoising and registration step is first performed as a simple pre-processing of the multimodal data (section 3.1). Next, we extend DT to multimodal DT (MMDT) (section 3.2) by using scene flow as well as RGB and optical flow. Finally, we include scene flow as a new input stream to 2DCNN (section 3.3) and perform late fusion of all models.

### 3.1. Denoising and data alignment

Kinect depth images capture the distance to the objects as pixel values. However, due to the limitations of the IR sensor, depending on the captured material and distance from the objects to the camera, pixel values may result in reading errors. We recover missing data by interpolating zero value pixels from its surrounding data based on elliptic PDE. Inpainting reconstruction is then smoothed using a *hybrid median filter* (HMF) in order to reduce any pixel flickering between consecutive frames. Compared to the classical median filter, this method removes noise while improving corner preservation. This is achieved by considering a 3-step method consisting of computing different medians for different spatial directions; ranking horizontal/vertical and diagonal medians separately to finally compute the median of both of them along with the central pixel value.

Furthermore, some datasets are not distributed with an accurate RGB-D alignment. This is a common issue to address when working with images captured using a Kinect device, since their IR and optical cameras are separated from each other. In these cases, RGB-D registration is also required. In our case, we use the intrinsic (focal length and the distortion model) and extrinsic (translation and rotation) camera parameters to warp the color image to fit the depth one. Example results of the denoising and registration preprocessing procedures are shown in Figure 1.

### 3.2. Handcrafted Features

DT [10] were proposed for RGB data, obtaining current state of the art results for handcrafted features in action recognition. Here, DT is extended to be applied on RGB-D data by taking into account the depth information and motion features from RGB-D; i.e., by means of scene flow.

#### 3.2.1 Dense Trajectories

Dense Trajectories provide a video representation based on densely sampled trajectories and a set of descriptors: HOG for the spatial appearance, HOF for the first-order motion

Figure 1: 1st row: inpainting+HMF results of a depth sample from isoGD [35] dataset. 2nd row: superposition before registration. 3rd row: superposition after registration.

information, and MBH for the second-order motion information, respectively. Dense sampling leads the method to capture local information both from foreground and its surrounding objects. In other words, this method considers both motion appearance features.

DT algorithm is computed over multi-scale images. The first step consists of dense sampling of feature points over the first frame to ensure feature points cover all spatial positions and scales. Points over homogeneous areas are rejected because of tracking unfeasibility. Then, dense optical flow is computed for the current frame with respect to the next one. Then, three kinds of trajectories are removed: static trajectories which have no motion information, trajectories with sudden large displacements, and trajectories with higher XY variances.

In its standard version, sampled trajectories are tracked along $L = 15$ frames. As it can bee seen in Figure 2 the space-time volume around each trajectory is divided into $n_\sigma \times n_\sigma \times n_t$ cells, where $n_\sigma = 2$ and $n_t = 3$. For each sub-division of the spatio-temporal grid, descriptors are computed (HOG, HOF and MBH) from a neighborhood of $N \times N$ pixels, with $N = 32$ around each points in the trajectories. Then, the final descriptor is constructed by the concatenation of the histograms of each cell.

### 3.2.2 Multimodal Dense Trajectories

**Trajectories.** The original DT algorithm used optical flow for trajectory construction. However, this presents the drawback that trajectories are computed from pixels of the image plane. The correspondence along pixels and real world spatial coordinates directly depends on the distance to the camera. Hence, the same movement performed at different points of the space may produce different trajectory lengths in terms of pixels. Meter is utilized as the unit for scene flow

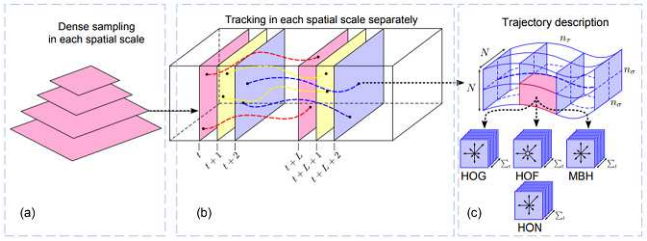


Figure 2: Dense trajectories algorithm. a) multi-scale dense sampling, b) pixel tracking for trajectory construction and c) descriptor extraction around trajectories.

(a) No prune (b) Pruned

Figure 3: Pruning trajectories using scene flow as criteria effectively changes the distribution of trajectories, focusing on regions which properly correspond to real motion.

instead of pixels which are used for optical flow. In real 3D world distance between two objects in meters does not depend on the relative position to the camera. Hence, scene flow is invariant to the distance of objects to the camera.

The procedure for trajectory construction is the same as the original algorithm. In the first step, when a trajectory has achieved a length of $L = 15$, it is classified as valid trajectory, rejected otherwise. The original DT used thresholds on the length and the amount of motion of the trajectory measured in pixels provided by optical flow. In MMDT, we sample scene flow along the trajectories. We filter invalid trajectories by the information achieved by scene flow in meters. The effect of pruning dense trajectories by scene flow is depicted in Figure 3. We can observe two main potential advantages. The first one is that length and motion is measured in real 3D world units (meters), achieving invariance to the position of the subject relative to the camera. Secondly, scene flow has an additional dimension, which allows the measurement of motion through Z-axis. As a consequence, it is expected that the quality and distribution of trajectories improve. Another idea could be using scene flow instead of optical flow to form trajectories. However, although scene flow can be more discriminate, scene flow is sensitive to the noisy characteristic of depth maps, and thus, it may produce a less accurate description of motion.

**Descriptors.** Following the philosophy of descriptors computation from the spatio-temporal cells and concatenation of the resulting histograms, we propose to use HON descriptor by taking advantage of the new source of information; i.e., depth maps. As it can be seen in Figure 2, HON

descriptor is added at the last stage of DT algorithm.

For HON computation [36], each normal is represented by two angles $\theta$ and $\phi$. The surfaces seen by the camera will always have normals facing the camera as well, that is: $0 < \theta < \pi$ and $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$. To construct the histogram, for simplicity, a 2D histogram is used instead of a sphere tessellation histogram. Then, for each angle, 5 bins are considered, each of them separated $\pi/4$ radians, which leads to a total of 25 bins for each sub-histogram. The final descriptor is the concatenation of these sub-histograms along the 12 spatio-temporal cells, resulting in 300 dimensions.

### 3.3. Deep Learning Models

In this work we use 2DCNN [8] with scene flow alongside RGB and optical flow. Methods based on 2D models use to randomly sample video frames, and then obtain class score predictions per sampled frame. Score averaging is commonly considered to classify actions. Here, we first extract keyframes by video summarization and then apply 2D models on the selected frames.

#### 3.3.1 Video Summarization

Many deep methods mostly select a fixed number of frames with equal temporal spacing between them. Thus, some relevant information in unselected frames might be lost. In order to mitigate this problem we use video summarization. It allows to 1) select relevant visual information to discriminate actions while 2) keeping the size of the data small.

Video summarization allows for the extraction of few video frames (keyframes) so that they jointly try to maximaze the information contained in the original video. Keyframes can be useful in deep learning applications involving large amount of video data. In our case, we use *Sequential Distortion Minimization* (SeDiM) [37].

SeDiM selects frames so that the distortion between the original video and the video synopsis is minimized. Although SeDiM does not guarantee global minima of distortion, it provides a simple yet computationally feasible and discriminative way to extract keyframes. Summarization examples are shown in Figure 4.

#### 3.3.2 2DCNN

Simonyan et al. [8] presented a two-stream CNN which incorporates both spatial and temporal networks. Spatial network operates on individual video frames, effectively performing action recognition from still images. For the spatial network they used a pre-trained network on ImageNet [38]. Unlike spatial convnet, the input of the temporal model are volumes of stacking optical flow fields between several consecutive frames ($224 \times 224 \times 2F$, where $F$ is the number of stacking frames). Since the input of this model explicitly describes the motion, the network does not need to estimate

Figure 4: Obtained SeDiM $K = 5$ keyframes for different Montalbano RGB samples. First row shows an example for one sample belonging to 'vattene' gesture, second row for 'seipazzo' and third row for 'messidaccordo'.

motion implicitly. The original architecture consists of five convolutional layers, each of them followed by a pooling layer and three fully connected layers. Like [32], we use the same network for both spatial and temporal net except from the input layer, while the original two-stream ConvNets ignores the second local response normalized (LRN).

In this research we introduce *multimodal 2DCNN* (MM2DCNN) by using scene flow as the input data to 2DCNN along with RGB and optical flow. Scene flow for each pixel has three dimension of $(x, y, z)$ along three real world axis. We consider these three dimension as three input channels for 2DCNN. Therefore, we use the same architecture for scene flow as RGB data. For both RGB and optical flow streams, the network is finetuned from pre-trained models on UCF-101 dataset. Scene flow of each datasets is finetuned from the pre-tained model of its own RGB model.

### 3.4. Combining handcrafted and deep features

The fusion of both handcrafted features and deep learning-based ones has been studied by several researchers [32, 39]. Handcrafted spatio-temporal features contain discriminative motion information while deep models can accurately describe appearance. In this research, we also evaluate if the combination of outputs from different handcrafted and deep-based methods can improve final recognition performance. We extract outputs from the two main methods as a confidence matrix $CM(n, c) \in \mathbb{R}^{N \times C}$, where $N$ is the number of samples and $C$ is the number of classes. Element $CM(n, c)$ is the class membership probability of sample $n$ to belong to class $n$. For the combination of MMDT and MM2DCNN, we use a weighted score averaging strategy of the confidence matrices of two methods. In this case, the final $CM$ is calculated by a weighted some of two ones ($CM = \alpha \times CM_{MMDT} + (\beta) \times CM_{MM2DCNN}$). The weight $\alpha$ and $\beta$ are experimentally set for each dataset.

## 4. Experimental results

In this section, we evaluate the proposed MMDT and MM2DCNN and combination of descriptors from different

modalities on two public benchmark datasets: MSR Daily Activity [40] and Montalbano II [41, 22, 42]. Final fused results are also compared with the state-of-the-art methods of action recognition from RGB-D data.

## 4.1. Datasets

### 4.1.1 MSR Daily Activity 3D

This dataset consists of sixteen actions captured with Microsoft Kinect [40]. Each sample is composed of RGB video and a sequence of depth images. It consists of: *drink, eat, read book, write on paper, use laptop, play game, call cellphone, use vacuum cleaner, cheer up, sit still, walking, sit down, toss paper, lay down on sofa, stand up and play guitar*. Each action is performed twice by 10 different subjects, leading to 20 samples per action and a total of 320 samples. For experimental result, we use subject with numbers 1, 3, 5, 7, 9 for training and the rest for testing. One sample of this dataset is shown in the top row of Figure 5.

Figure 5: Samples from MSR Daily 3D Activity (top row) and Montalbano II (bottom row).

### 4.1.2 Montalbano II

This dataset is composed of 940 sample videos of subjects performing 20 different Italian gestures [41, 22]. Videos had been recorded with Kinect device, therefore, both RGB and depth data are available. Each video consist of several gestures, as this dataset is used for gesture detection as well. Nevertheless, all the videos had been previously split into single gesture samples. This leads to a total of $12,575$ samples. This dataset is already divided in three subsets, train, validation and test. One sample of this dataset can be seen in the middle row of Figure 5.

## 4.2. Results

Here, we compare DT and proposed MMDT considering the different descriptors for the different modalities. We also compare performance of different modalities over keyframes and fusion results. For final classification we utilize a weighted sum of the class scores per each modality.

### 4.2.1 MMDT

After extracting each descriptor from each modality, PCA is applied. The output size of PCA is 32. A codebook of size 32 is constructed for each descriptor separately. Then, each descriptor is assigned to a vocabulary word using *Fisher Vector* (FV) encoding. Finally we use SVM with RGB kernel to classify actions. The accuracy achieved by two methods with different modalities on MSR Daily dataset is shown in Table 1. The best accuracy for DT is obtained by the combination of HOG, HOF and MBH. On the other hand, best accuracy of MMDT is obtained by the fusion of HON and MBH. On the other hand, proposed MMDT trajectories achieve better performance results than DT.

Table 1: DT and MMDT accuracy on MSRDaily Act. 3D.

| Descriptors | DT | MMDT |
|---|---|---|
| HOG (RGB) | 43.125 | 45.625 |
| HON (Depth) | - | 72.5 |
| HOF + MBH (Opt. flow) | 62.5 | 70 |
| Best | **63.125** | **78.13** |

Table 2 shows the result of MMDT on Montalbano dataset. The combination of HOG, HOF, and MBH obtains the best accuracy for DT and MMDT. Although pruning DT with scene flow does not improve the results in this case, using HON instead of HOG results in better accuracy.

Table 2: DT and MMDT accuracy on Montalbano II.

| Descriptors | DT | MMDT |
|---|---|---|
| HOG (RGB) | 67.3 | 67.3 |
| HON (Depth) | - | 77.67 |
| HOF + MBH (Opt. flow) | 82.0 | 82.0 |
| Best | **83.5** | **85.66** |

### 4.2.2 MM2DCNN

In order to test how video summarization can affect the classification when considering different modalities without trajectories but at frame level, we have extracted $k = 14$ keyframe sequences from both RGB and depth videos for each dataset. Each experiment consists of testing each of the summarization sequences on a different model. Doing so, we intent to spot weather using keyframes can improve results compared to randomly selected frames or not.

Besides, we are also interested in analyzing if depth-based video summarization is able to hold similar results to RGB. In this regard, we include a hybrid-like summarization we refer to as RGB-D synopsis. RGB-D synopsis is an ordered concatenation of RGB $k = 7$ and depth $k = 7$ keyframe sequences to test how depth and RGB can contribute when combined.

Every 2DCNN has been fine-tunned from spatial/temporal UCF101 caffe models, using RGB, optical

flow and scene flow frames. Tables 3 and 4 include final accuracies for every CNN model and summarization modality. RGB and Depth columns refer to $k = 14$ summarization sequences for RGB and Depth videos separately, while the RGB-D column specify results for the hybrid combination. Finally, randomized-frame selection accuracy is also included for the sake of comparison.

Table 3 shows the evaluation of MM2DCNN on MSR Daily dataset. In general the result is not good for MSR Daily dataset. The most important problem related to MSR Daily for a deep model is the low number of samples. Among the different modalities, Scene flow has the best accuracy. We can also see that the keyframe selection strategy on depth data gets better result. Background in MSR Daily Compared to other dataset, background in MSR Daily is more clutter in RGB frames. Since the distance of the clutter background is more than the Kinect range of view, this dataset has cleaner depth images with less noise than its RGB ones. Thus the scene flow frames are more accurate. We can also see that the accuracy of scene flow is better than optical flow thanks to considering real 3D information.

Table 3: Accuracy for SeDiM on MSR Daily Activity 3D.

| Model | RGB | Depth | RGB-D | Random |
|---|---|---|---|---|
| RGB | 53.91 | 53.12 | 53.91 | 53.12 |
| Opt. flow | 55.47 | 57.81 | 55.47 | 55.70 |
| Scene flow | 67.19 | 68.75 | 66.41 | 64.84 |
| Late Fusion | 70.08 | **71.65** | 70.08 | 69.29 |

Figure 6 shows three examples from MSR Daily dataset in which each sample is classified correctly only by one modality and by the late fusion. The action in top row is "eat". While it is classified as "read a book" by optical flow and scene flow given the similar motion among these two classes, RGB can take benefit of appearance to discriminate the action. Middle row shows one sample that is classified correctly only by scene flow as "stand up". In this action the subject starts walking and approaching to the camera after standing up. Scene flow properly discriminates the action thanks to its invariance to camera distance in contrast to RGB and optical flow, which classify the action as "walk". Action in bottom row is "read a book". The movements of human body in this action are very slow. Optical flow shows to be more discriminative for slow motion, properly discriminating this action while the rest of modalities classify this action as "write on a paper".

Table 4 shows the performance of 2DCNN on Montalbano dataset. The accuracy of RGB data is the best achieved one. Compared to MSR Daily, background of samples in Montalbano dataset is simpler. Moreover, this dataset is larger than MSR Daily, and there are more samples to fine-tune the network. Therefore, weights can be better learnt. For this dataset, RGB modalities work better than others.

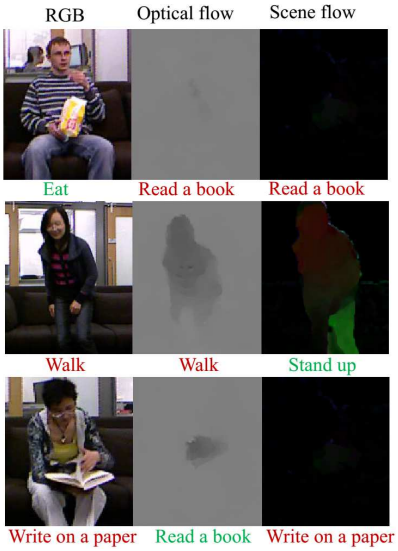


Figure 6: Examples from MSR Daily. Each column shows one modality. Each rows shows the classification result of each modality. Red: Wrong classification, Green: Correct classification.

Depth data of Montalbano are noisier than MSR Daily. Hence, its scene flow is also noisy. However, it is worth mentioning that scene flow gives better accuracy than optical flow (like MSR Daily). Different strategies of keyframe selection result in almost the same accuracy.

Table 4: Accuracy for SeDiM on Montalbano II.

| Model | RGB | Depth | RGB-D | Random |
|---|---|---|---|---|
| RGB | 96.03 | 97.06 | 95.72 | 97.06 |
| Opt. flow | 61.06 | 59.74 | 60.67 | 64.24 |
| Scene flow | 69.90 | 69.68 | 69.02 | 70.93 |
| Late Fusion | 96.28 | 96.25 | 96.16 | **97.06** |

For both datasets, different tested strategies of selecting keyframes (i.e., from different modalities) does not significantly affect the result. The reason of it might be related to the fact that the general human body shape and its changes during time are almost the same in both RGB and depth data. As for the late fusion, we use different weights for different modalities of $[1, 0.2, 0.3]$ for RGB, optical flow and scene flow, respectively. It can be seen that late fusion can improve the results for most of keyframe strategies.

#### 4.2.3 Combination of MMDT and MM2DCNN

We also combined MMDT and MM2DCNN by applying a second late fusion strategy for both datasets. Table 5 shows the result of the second late fusion. It can be seen that the accuracy is improved for both datasets. Deep learning methods can encode appearance features than handcrafted one,

while handcrafted features can achieve good result on motion features. By combining these two kinds of features we can benefit from both methods. By this combination we achieve the state of the art result on Montalbano II and a comparable result with the state of the art method on MSR Daily dataset. For Montalbano II dataset, the per gesture accuracy for all gestures are more than 0.90. Figure 7 shows the confusion matrix of MSR Daily. Some actions with the same appearance and motion features are mistaken, such as "read a book" and "write on a paper". In these actions the position of human body and its motion are the same.

Table 5: Second late fusion of MMDT and MM2DCNN.

| Dataset | Accuracy |
|---|---|
| MSR Daily | 82.50 |
| Montalbano II | 97.44 |

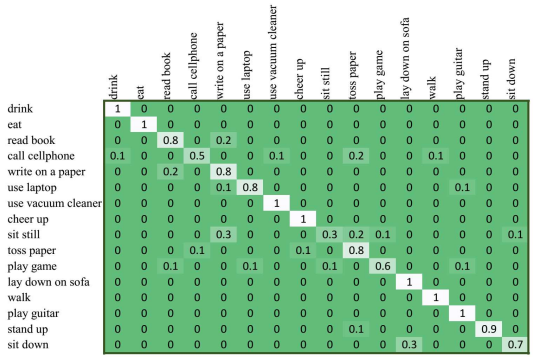


Figure 7: Confusion matrix of MSR Daily Activity 3D.

Table 6 lists the accuracy of the existing methods on MSR Daily Dataset. MMDT achieves better accuracy than MM2DCNN. It can be seen that the accuracy of MMDT is comparable with the best state of the art accuracy [40].

Table 6: Performance comparison on MSRDaily Act. 3D.

| Method | Accuracy |
|---|---|
| EigenJoints[43] | 58.10 |
| MovingPose[44] | 73.80 |
| HON4D [15] | 80.00 |
| SSTKDes [16] | 85.00 |
| ActionLet [40] | **85.75** |
| MMDT | **82.50** |
| MM2DCNN | 71.65 |

Although Montalbano II dataset was designed for segmentation, it has been recently re-used by some researchers just for classification. Table 2 lists a number of methods on this dataset. Other papers published precision for this dataset. The precision achieved by our method is 97.52, defining current state of the art results.

Table 7: Performance comparison on Montalbano II.

| Method | Accuracy/Precision |
|---|---|
| Fernando et al. [45] | 75.3 |
| Pigou et al. [46] | 94.49 |
| MMDT | 85.66 |
| MM2DCNN | **97.44 (97.52 Precision)** |

In Table 6, it can be seen that MMDT works better than MM2DCNN for MSR Daily, while the result of MM2DCNN is better than MMDT for Montalbano II dataset in Table 7. It is due to the fact that, Montalbano II is large dataset, and deep models with works better with larger data. On the other hand, handcrafted features can model motion information for smaller dataset.

## 5. Discussion

In this work we proposed two methods, i.e., MMDT and MM2DCNN for multimodal RGB-D action recognition. By taking into account depth data and scene flow, we showed performance improvements in comparison to only considering RGB data, achieving state of the art results on two public RGB-D action datasets.

For MMDT, we showed that considering scene flow to prune DT trajectories can result in performance improvements with respect trajectories computed from RGB. For MM2DCNN we considered different multimodal descriptors to be trained within a 2DCNN. Among these two methods, MMDT (like other handcrafted methods) was able to produce better results for the smaller dataset (MSR Daily) since deep models need large amounts of data for a better generalization. Among the features used in this research, 2DCNN on RGB data of Montalbano II resulted in better accuracy. This model was fine-tuned from a pre-trained network on UCF-101 dataset [8], previously fine-tuned on ImageNet [38]. In addition, gestures in Montalbano II datasets have simple (static and near homogeneous) background.

We have also tested fine-tuning one modality from other modalities. It is worth mentioning that it works better than training from scratch. For instance, by training scene flow from scratch the per frame accuracy of one epoch was around 20% while the per frame accuracy of fine-tuning the same network from pre-trained RGB model was around 60 %.

*Acknowledgements*

This work has been partially supported by the Spanish project TIN2016-74946-P (MINECO/FEDER, UE) and CERCA Programme / Generalitat de Catalunya. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

## References


[1] Znaonui Liang, Gang Zhang, Jimmy Xiangji Huang, and Qmming Vivian Hu. Deep learning for healthcare decision making with emrs. In *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*, pages 556–559. IEEE, 2014.

[2] Richard Marks. System and method for providing a real-time three-dimensional interactive environment, December 6 2011. US Patent 8,072,470.

[3] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3908–3916, 2015.

[4] Jincheng Yu, Kaijian Weng, Guoyuan Liang, and Guanghan Xie. A vision-based robotic grasping system using deep learning for 3d object recognition and pose estimation. In *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pages 1175–1180. IEEE, 2013.

[5] S Mohsen Amiri, Mahsa T Pourazad, Panos Nasiopoulos, and Victor CM Leung. Human action recognition using meta learning for rgb and depth information. In *Computing, Networking and Communications (ICNC), 2014 International Conference on*, pages 363–367. IEEE, 2014.

[6] Chen Chen, Baochang Zhang, Zhenjie Hou, Junjun Jiang, Mengyuan Liu, and Yun Yang. Action recognition from depth sequences using weighted fusion of 2d and 3d auto-correlation of gradients features. *Multimedia Tools and Applications*, 76(3):4651–4669, 2017.

[7] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013.

[8] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576. 2014.

[9] Mariano Jaimez, Mohamed Souiai, Javier Gonzalez-Jimenez, and Daniel Cremers. A primal-dual framework for real-time dense rgb-d scene flow. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 98–104. IEEE, 2015.

[10] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3169–3176. IEEE, 2011.

[11] Maryam Asadi-Aghbolaghi, Albert Clapés, Marco Bellantonio, Hugo Jair Escalante, Víctor Ponce-López, Xavier Baró, Isabelle Guyon, Shohreh Kasaei, and Sergio Escalera. Deep learning for action and gesture recognition in image sequences: A survey. In *Gesture Recognition*, pages 539–578. Springer, 2017.

[12] Maryam Asadi-Aghbolaghi, Albert Clapés, Marco Bellantonio, Hugo Jair Escalante, Víctor Ponce-López, Xavier Baró, Isabelle Guyon, Shohreh Kasaei, and Sergio Escalera. A survey on deep learning based approaches for action and gesture recognition in image sequences. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, pages 476–483. IEEE, 2017.

[13] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 357–360. ACM, 2007.

[14] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008.

[15] Omar Oreifej and Zicheng Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2013.

[16] Maryam Asadi-Aghbolaghi and Shohreh Kasaei. Supervised spatio-temporal kernel descriptor for human action recognition from rgb-depth videos. *Multimedia Tools and Applications*, pages 1–21, 2017.

[17] Peng Wang, Jingdong Wang, Gang Zeng, Weiwei Xu, Hongbin Zha, and Shipeng Li. Supervised kernel descriptors for visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2858–2865, 2013.

[18] Rizwan Chaudhry, Avinash Ravichandran, Gregory Hager, and René Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1932–1939. IEEE, 2009.

[19] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60, 2013.

[20] Christoph Vogel, Konrad Schindler, and Stefan Roth. Piecewise rigid scene flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1377–1384, 2013.

[21] Ye Zhang and Chandra Kambhamettu. On 3d scene flow and structure estimation. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2001.

[22] Sergio Escalera, Xavier Baró, Jordi Gonzalez, Miguel Ángel Bautista, Meysam Madadi, Miguel Reyes, Víctor Ponce-López, Hugo Jair Escalante, Jamie Shotton, and Isabelle Guyon. Chalearn looking at people challenge 2014: Dataset and results. In *ECCV Workshops (1)*, pages 459–473, 2014.

[23] Lin Sun, Kui Jia, Dit-Yan Yeung, and Bertram E Shi. Human action recognition using factorized spatio-temporal convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4597–4605, 2015.

[24] Xiaolong Wang, Ali Farhadi, and Abhinav Gupta. Actions˜ transformations. In *Proceedings of the IEEE Conference*

*on Computer Vision and Pattern Recognition*, pages 2658–2667, 2016.

[25] Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Learning to track for spatio-temporal action localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3164–3172, 2015.

[26] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE TPAMI*, 35(1):221–231, 2013.

[27] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *arXiv preprint arXiv:1604.04494*, 2016.

[28] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[29] Felix A Gers, Nicol N Schraudolph, and Jürgen Schmidhuber. Learning precise timing with lstm recurrent networks. *JMLR*, 3(Aug):115–143, 2002.

[30] Vivek Veeriah, Naifan Zhuang, and Guo-Jun Qi. Differential recurrent neural networks for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4041–4049, 2015.

[31] Yong Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118, June 2015.

[32] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4305–4314, 2015.

[33] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1933–1941, 2016.

[34] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[35] Jun Wan, Yibing Zhao, Shuai Zhou, Isabelle Guyon, Sergio Escalera, and Stan Z Li. Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 56–64, 2016.

[36] Shuai Tang, Xiaoyu Wang, Xutao Lv, Tony X Han, James Keller, Zhihai He, Marjorie Skubic, and Shihong Lao. Histogram of oriented normal vectors for object recognition with a depth sensor. In *Asian conference on computer vision*, pages 525–538. Springer, 2012.

[37] Costas Panagiotakis, Nelly Ovsepian, and Elena Michael. Video synopsis based on a sequential distortion minimization method. In *International Conference on Computer Analysis of Images and Patterns*, pages 94–101. Springer, 2013.

[38] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[39] Zhe Wang, Limin Wang, Wenbin Du, and Yu Qiao. Exploring fisher vector and deep networks for action spotting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 10–14, 2015.

[40] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1290–1297. IEEE, 2012.

[41] Sergio Escalera, Jordi Gonzàlez, Xavier Baró, Miguel Reyes, Oscar Lopes, Isabelle Guyon, Vassilis Athitsos, and Hugo Escalante. Multi-modal gesture recognition challenge 2013: Dataset and results. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 445–452. ACM, 2013.

[42] Sergio Escalera, Vassilis Athitsos, and Isabelle Guyon. Challenges in multi-modal gesture recognition. In *Gesture Recognition*, pages 1–60. Springer, 2017.

[43] Xiaodong Yang, Chenyang Zhang, and YingLi Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 1057–1060. ACM, 2012.

[44] Mihai Zanfir, Marius Leordeanu, and Cristian Sminchisescu. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2752–2759, 2013.

[45] Basura Fernando, Efstratios Gavves, José Oramas, Amir Ghodrati, and Tinne Tuytelaars. Rank pooling for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):773–787, 2017.

[46] Lionel Pigou, Aäron van den Oord, Sander Dieleman, Mieke Van Herreweghe, and Joni Dambre. Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. *International Journal of Computer Vision*, pages 1–10, 2015.