Master's Thesis

Multimodal 2DCNN action recognition from RGB-D Data with Video Summarization

Vicent Roig Ripoll

Master
in
Artificial Intelligence

Advisor: Sergio Escalera Guerrero
Co-advisor: Maryam Asadi-Aghbolaghi

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

October, 2017

Abstract

Human action recognition is nowadays within the most active computer vision research areas. The problem of action recognition is challenging due to the large intra-class variations, low video resolution and high dimension of video data, among others things. Recent development of affordable depth sensors like Microsoft Kinect leads to new opportunities in this field by providing both RGB and depth data. Multimodal fusion in this scenario can greatly help to boost performance of action recognition methods. Recently, although handcrafted features are still widely used owing to their high performance and low computational complexity, there has been a migration from traditional handcrafting towards deep learning. In this work, 2DCNN is extended to multimodal (MM2DCNN) by introducing scene flow fields as the new input for an additional stream. Then, model outputs are integrated in a late fusion fashion. Furthermore, this work also focuses on analyzing the impact of video summarization in action recognition models. To this end, four different summarization techniques have been applied and compared to uniform random selection. Video summarization algorithms aim to select the most discriminative frames of each video, providing keyframe sequences as a result. Each of these methods has been performed over the two different types of data available, extracting keyframe sequences from RGB and depth videos separately. On top of that, we also perform a novel hybrid-like summarization, namely RGB-D synopsis, by combining results from both sequences. Finally, we evaluate and compare the results of each modality in three state-of-the-art action datasets, integrating them with a late fusion for every summarization sequence modality along with uniform random selection. Experimental results show that our new representation improves the accuracy in comparison to 2DCNNs. Besides, the use of video summarization succeeds in boosting the final performance when compared to random frames.

Contents

1 Introduction **1**

2 Related Work **4**
2.1 Hand-crafted Features 4
2.2 Deep Learning 6
2.3 Convolutional Neural Networks 7
2.4 Two-stream Convolutional Neural Network 8
2.4.1 Architecture 8
2.4.2 Spatial stream ConvNet 8
2.4.3 Temporal stream ConvNet 9
2.4.4 Very deep two-stream ConvNets 10

3 Video Summarization **12**
3.1 Absolute Histogram Difference 13
3.2 Time Equidistant Algorithm 13
3.3 Sequential Distortion Minimization 14
3.3.1 Distortion formulation 15
3.3.2 Method Description 16
3.4 Content Equidistant Algorithm 17

4 Proposed Method **20**
4.1 RGB and Depth Registration 20
4.2 Denoising 21
4.2.1 Hybrid Median Filter 22
4.3 Multimodal 2DCNN 24
4.3.1 Late Fusion 25

5 Experimental results **26**
5.1 Datasets 27
5.1.1 MSR Daily Activity 3D 27
5.1.2 Montalbano V2 27

5.1.3 IsoGD 29
5.2 Experiments 32
5.2.1 MSR Daily Activity 3D 32
5.2.2 Montalbano V2 34
5.2.3 IsoGD 37
5.3 Comparison 39

6 Discussion **42**
6.1 Conclusion 42
6.2 Future work 43

A Confusion matrices **44**

Bibliography **56**

Chapter 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 healthcare monitoring [Lia+14], interactive gaming [Mar11], surveillance [AJM15], and robotics [Yu+13], to mention a few.

During past decades, research on human action recognition has been extensively explored on RGB data. The recent advances in imaging devices, in particular Microsoft Kinect, have facilitated 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. Compared with RGB data, depth images are insensitive to illumination changes and have discriminative information of the 3D geometrical data. Therefore, fusing these multimodal information into highly discriminative feature sets can lead methods to achieve higher levels of performance.

Many approaches [Ami+14; Che+17] 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 [WS13; SZ14a]. Compared to optical flow, which is the projected motion onto the 2D image plane, scene flow [Jai+15] 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 preserves 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, dense trajectories (DT) [Wan+11] and its extension, improved dense trajectories (iDT) [WS13], which also removes camera motion from trajectories, 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 [Asa+17c; Asa+17b], deep architectures used for action recognition are categorized in four groups: 2D models, motion-based input features, 3D models, and temporal networks. Generally, handcrafted features are more powerful in describing motion information while deep learning-based representations are quiet good at describing appearance data.

In a 2D convolutional neural network (2DCNN), the deep model is trained using individual frames as input data. Regarding the test mode, the model is used to classify different frame sequences, including randomly selected frames from the whole sequence to finally apply average scoring to get the final classification. Here, we focus on comparing the performance of deep learning features for multimodal human action recognition as well to introduce different frame selection approaches. For this, 2DCNN has been extended by using different modalities; i.e, RGB, optical flow, and scene flow. Also different kinds of video summarization are introduced in chapter 3.

In addition to the foregoing, we evaluate the incorporation of scene flow information in deep learning action recognition systems. Each modality is trained separately by a 2DCNN and final classification is done by score averaging. The experimental results show that scene flow is more discriminative than optical flow for recognizing actions. Also, results show that fusing information from different modalities improve the accuracy compared to just

using one modality.

Chapter 2

Related Work

2.1 Hand-crafted Features

In the literature of handcrafted methodologies, authors usually rely on three main approaches in order to cope with temporal information, i.e. 1) treating videos as spatio-temporal volumes, 2) flow-based features to explicitly deal with motion, and 3) trajectory-based approaches where motion is implicitly modeled. First group is considered as the spatio-temporal extension of classical descriptors in image recognition to the temporal dimension [SAS07; KMS08; OL13; AK17]. [SAS07] proposed 3D-SIFT which is the 3D extension of SIFT descriptor to include temporal dimension. In this case, apart from the computation of the image gradient on axis x and y, an additional gradient estimation is performed over the time dimension, resulting in a three-dimensional field vector, which allows for the construction of a 2D histogram (3D orientations represented with θ, ϕ and magnitude). The 3D neighborhood around a point of interest can be rotated so that the dominant orientation has $\theta = 0$ and $\phi = 0$, then, sub-histograms are built around the interest point in 3D cells of $4 \times 4 \times 4$ which encode spatio-temporal information, thus creating a descriptor invariant to orientations. This descriptor is invariant to orientation (temporally as well), which presumably can better generalize the underlying information to discriminate actions.

Similarly, in [KMS08], authors introduced the idea of HOG3D. To do so, 3D-XYT volume is considered for the computation of the descriptor. [OL13] 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 [AK17] proposed

supervised spatio-temporal kernel descriptor (SSTKDes) for recognizing human actions as the extended version of *supervised kernel descriptor* (SKDES) [Wan+13b] which utilized the kernel principal component analysis (KPCA) and *large margin nearest neighbour* (LMNN) to learn a compact descriptor for object recognition.

Motion features like optical flow are very successful on action recognition, since they have local temporal information, many authors have used them to construct descriptors which are included in the second category. In [Cha+09], authors introduced *histogram of oriented flow* (HOOF). In [Wan+13a] it is proposed the *motion boundary histograms* (MBH) by using the second order of optical flow. On the other hand, from the sequences of RGB frames, it is possible to extract motion features such as optical flow, from which is possible to compute handcrafted descriptors such as Histogram of Oriented Flow (HOOF) [Cha+09] and Motion Boundary Histograms (MBH) [Wan+13a]. This features encode motion information that is highly discriminative for action recognition.

The main idea of the approaches in the third group is to use trajectories which consider longer temporal information. In [Wan+11; Wan+13a; WS13], authors propose the use of optical flow for trajectory construction, and descriptors are computed around representative trajectories. The algorithm starts by densely sampling feature points on the first frame. These points are tracked using optical flow to form trajectories. Only high-motion regions of interest are kept and static tracks are removed. Surrounding these resulting trajectories, a spatio-temporal window is created and subdivided into sub-cells, where descriptors (HOG, HOF and MBH) are computed. The final feature descriptor per trajectory is the concatenation of these descriptors and for each video, a set of trajectories is obtained.

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.

As stated in the introduction, with the incorporation of the Kinect camera, which includes IR sensors allowing for depth data collection along with the

regular RGB imaging, new motion features emerge. Those captured depth maps, measured in millimeters represent the distance to the nearest object at that particular (x, y) coordinate in the depth sensor's field of view. Moreover, by knowing the field of view camera parameters, we can get a direct correspondence from pixels to the real world coordinates in relation to the camera position.

This allows to extended optical flow to its 3D homologous, scene flow [VSR13; ZK01; Jai+15], along with their corresponding 3D version from scene flow (HOSF and MBH_z). Scene flow 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 [VSR13; ZK01]. These methods suffered from a high computational cost. Jaimez et al. in [Jai+15] 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. Just as scene flow is the three-dimensional motion field of points in the real three-dimensional world and optical flow is the two-dimensional motion field of points in an image, we can understand optical flow as the projection of the scene flow onto the image plane of a camera.

2.2 Deep Learning

Articial intelligence is a prosperous and growing field with many practical applications and active research topics. We look to intelligent software to automate routine labor, understand speech or images. In the early days of articial intelligence, the field rapidly tackled and solved problems that were difficult for people but relatively simple for computers when properly described by a list of formal, well defined mathematical rules. The true AI potential unfolds when trying to solve problems that are easy for people to perform but hard for people to describe formally problems that we solve intuitively, that feel automatic, like understanding a speech or to recognize faces.

Similarly to humans, a solution is to allow computers to learn from experience so they can understand the domain by its hierarchy of concepts, with each concept defined by its relation to simpler ones. By collecting knowledge from experience, we avoid to describe and formalize all the knowledge and rules the computer would need. This hierarchy of concepts allows the computer to understand complex concepts by means of simpler ones. If

we were to draw a graph showing how these concepts are built on top of each other, the graph would be deep, with many layers. That is why this approach is known as deep learning [GBC16]. Deep Learning attempts to model high level abstractions by using a deep artificial neural network (ANN) with multiple processing layers, composed of multiple linear and non-linear transformations, such is the case of Convolutional Neural Networks.

Deep neural networks have shown successful results on image-based recognition tasks [KSH12; SZ14b; ZF14]. Also, there have been a number of studies presenting deep architectures for action recognition [SZ14a; Kar+14; WQT15; Ji+13; CLS15].

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a type of feed-forward networks based on the MLPs which connectivity pattern is inspired by Hubel and Wiesel's early work on the organization of the cat's visual cortex [HW68]. In 1962, they realized that the visual cortex contains a complex structure of cells, which are sensitive to small sub-regions of the visual field, called a receptive field. The sub-regions are tiled to cover the entire visual field. These cells act as local filters over the input space and are well-suited to exploit the strong spatially local correlation present in natural images. They also discovered complex cells, which also responded to edges of a specific orientation at a specific location, which is one of the main characteristics in how CNNs learn and extracts information.

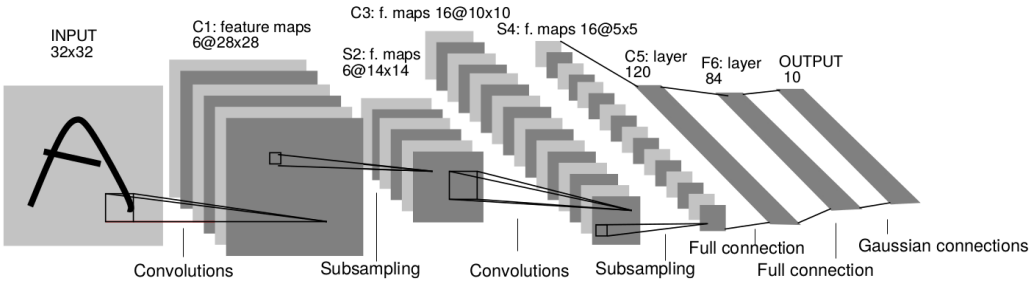


Figure 2.1: CNN architecture scheme.

As can be seen in Figure 2.1, CNNs are formed by convolutional layers and pooling layers. Convolutional layers apply convolutional filters to different parts of the image defined by a set of heights and pooling layers aggregate the output of the convolutional layers resulting in a sub-sampling

of the previous layer image.

CNNs makes use of a unsupervised pre-training step, based on training the input layer with the inputs and obtain the set of parameters for that layer. The output of the first layer is used as the input for the second and this step is repeated for every layer in the net.

2.4 Two-stream Convolutional Neural Network

Proposed by [SZ14a], it is an extension of the deep Convolutional Neural Networks, which nowadays is the state-of-the-art on still image representation, to the domain of automatic action and gesture recognition. This new architecture performs the recognition by processing two different streams (spatial and temporal) at the same time, and combining both by late fusion. The first stream captures spatial information from still frames of the video whereas the temporal stream uses as input the dense optical flow extracted from the same video. Both of the streams are implemented as a regular Convolutional Neural Network. This supposes an advantage as the spatial stream network can be pre-trained on larger datasets and fine tuned for the action or gesture recognition application.

The idea behind this proposed architecture is the two-stream hypothesis [GM92]. The same suggests that human visual cortex is indeed composed of two pathways: the ventral stream (which performs object recognition) and the dorsal stream (which recognizes motion). Nonetheless, authors did not further researched this connection.

2.4.1 Architecture

As mentioned above, the architecture of the two-stream convolutional neural network is actually a composition of two different deep ConvNets, which its soft-max scores are combined by late fusion. In figure 2.2, a detailed scheme of the model is shown.

2.4.2 Spatial stream ConvNet

The spatial recognition stream ConvNet receives video frames as its input, and then performs action recognition from still images. Static appearance of the video can be discriminative enough by itself, as some actions are strongly

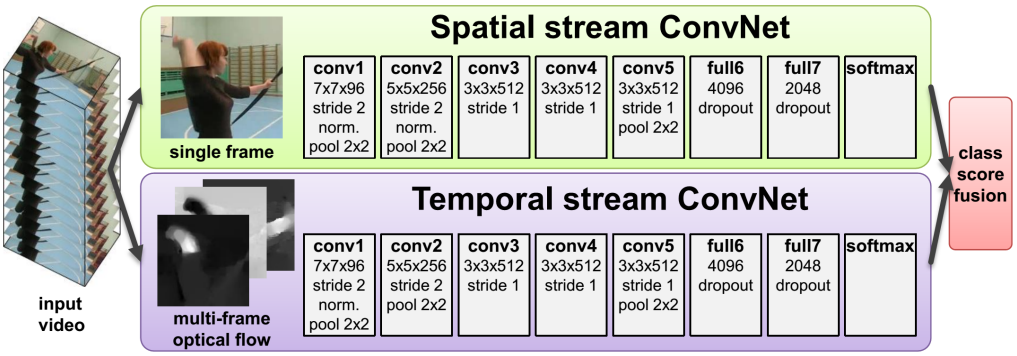


Figure 2.2: Two-stream architecture for video classification

associated to some objects (e.g. *'use a laptop'*, *'play a game'*, *'call from a cellphone'*, etc).

Additionally, as this architecture is basically an image classification model, it can benefit from the recent advances in large scale image recognition methods [KSH12] and pre-train the network on a large dataset, to later fine-tune its parameters for the specific domain of action or gesture recognition.

2.4.3 Temporal stream ConvNet

On the other hand, the temporal recognition stream ConvNet takes as input the stacks of optical flow displacement fields along several video frames. This kind of input explicitly defines the motion of the video, which makes the task easier, as the neural network does not have to learn to implicitly estimate motion. Authors of the method propose different possibilities on how to stack optical flow.

Optical flow stacking

A dense optical flow is a vector displacement field between a pair of frames. Each vector $d(x, y)$ corresponds to the displacement of the pixel in the position (x, y) to its position in the next frame. This configuration proposes to encode each component of the vector d, d_x and d_y as image channels. Therefore, being w and h the width and the height of the video, for an stack of L frames, the optical flow input volume dimension is $w \times h \times 2L$. Where each point $(x, y) \in \mathbb{R}^{w \times h}$ corresponds to a pixel position, odd channels store the horizontal components of optical flow, while even channels store the vertical one.

Trajectory stacking

Inspired by trajectory-based descriptors. Instead of sampling optical flow at the same location at each frame, flow is sampled along motion trajectories. The first element of the stack would be the same, but in the next one, the value for the position (x, y) is the flow sampled at the position $(x+d_x, y+d_y)$, where d is the optical flow of the first frame at location (x, y). Following this procedure through the L frames will result in an input volume of trajectories.

Bi-directional optical flow

Optical flow represents the displacements from frame t to frame $t+1$. It is possible to conceive then an extension to a bi-directional flow by computing an additional set of displacements in the opposite direction that connect a frame t with the frame $t-1$. This way, for L frames, $L=2$ forward optical flow is stacked for frames t to $t+L=2$, and $L=2$ backwards optical flow for frames t to $t/L=2$. This conception can be constructed using both of previous approaches, optical flow stacking and trajectory stacking.

Mean flow subtraction

For deep learning models, it is beneficial to use zero-centered input, as it allows the model to better exploit the rectification non-linearities. Optical flow can take both positive and negative values, and as movement to one direction is as probable as movement in the opposite direction, it can be naturally centered. Nevertheless, it is probable that between a pair of frames, movement in one direction is dominant. So, for each displacement vector d, the mean value is subtracted.

2.4.4 Very deep two-stream ConvNets

In [Wan+15], the author presented an extension to two-stream ConvNets called Very deep two-stream ConvNets by adapting them into the video domain, motivated by the lack of deep learning improvement in the action recognition field. Two-stream ConvNets seem to be too shallow for action recognition (5 convolutional layers and 3 fully-connected layers) when compared with those deeper models in image domain (e.g. VGGNet [SZ14b], GoogLeNet [Sze+15]), therefore their modeling capacity is constrained by their depth. Besides, datasets for action recognition tend to be really small, this contrasts with the huge number of training samples required for deep ConvNets to tune the network weights. To address these problems, a deeper architecture is proposed (up to 19 layers, based in [SZ14b]) and several good

practices are considered to reduce the effect of over-fitting, including smaller learning rates, data augmentation techniques and higher drop-out ratios.

As a result, Very deep two-stream ConvNets introduce high modeling capacity and are capable of handling the large complexity of action classes. Their design takes over two successful network architectures in object recognition, namely GoogLeNet and VGGNet, achieving a 91.4% accuracy on the UCF-101 [SZS12] dataset. The spatial stream network is built on a single frame image ($224 \times 224 \times 3$), therefore its architecture is the same as those for object recognition in image domain. The input of temporal net is 10-frame stacking of optical flow fields ($224 \times 224 \times 20$). Thus, the convolutional filters in the first layer are different from those of image classification models. It also relies on more data augmentation techniques, as multi-scale cropping for training and a new 4 corner cropping strategy. Besides, it sets high drop out ratios for the fully connected layers of both streams, temporal nets set drop out ratios to 0.9 and 0.8, while spatial nets use 0.9 and 0.9.

Chapter 3

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 the process. 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 maximize the information contained in the original video. A good summarization should not only consider frame relevance as the main selection criterion, but also try to keep frames from throughout the video in order to get a complete coverage. These keyframes are useful in multiple scenarios, providing high-level semantic information in video browsing and broadcast.

According to Panagiotakis [POM13], keyframe selection approaches can be generally classified into three categories, namely cluster-based methods, energy minimization-based methods and sequential methods [PDT09], [PDT07]. Cluster-based methods take all frames from every shot and classify by content similarity to take keyframes. The disadvantage of this approach is that the temporal information of a video sequence is omitted. Energy minimization methods extract keyframes by solving a rate-constrained problem. These methods are generally computational expensive by iterative techniques. Sequential methods consider a new keyframe when the content difference from the previous keyframe exceeds the predefined threshold. Moreover, some of the best solutions found in recent literature take advantage of dynamic programming, such as the MINMAX method [LSK05], at the expense of higher computational cost.

For this work, we apply some methods found in the literature with the intention of reducing the amount of data used towards deep models and evaluate each sequence in different modalities. Selected video summarization techniques are explained in detail below.

3.1 Absolute Histogram Difference

In the context of video analysis, Sheena and Narayanana [CN15] introduced a simple summarization technique based on the absolute difference of histograms of consecutive frames. This method is constituted of two parts. First, a threshold is computed given the mean and the standard deviation of the histogram of absolute difference of each pair of consecutive frames. Then, keyframes are extracted by comparing the threshold against these absolute differences. The algorithm starts by extracting video frames one by one, after pre-processing each frame, the histogram difference between two consecutive frames is computed. The mean and standard deviation of the absolute difference of histograms are used to fix the threshold point T using the following equation:

$$T = \mu_{adh} + \sigma_{adh} \tag{3.1}$$

Where μ_{adh} is the mean of the absolute difference and σ_{adh} is the standard deviation.

Despite the simplicity of this technique, it manages to discriminate quite well relevant frames coming from sudden changes just by measuring the *amount of variation* between consecutive frame intensities. Doing so, we get the benefit of being able to extract any desired number of keyframes from the histogram difference matrix, simply by selecting the best k candidates, those with highest difference, representing distinctive frames among the rest. However, this method might not be good enough to retain all required information from throughout the video to get a good overall coverage, as those k obtained keyframes are more likely to come out of segments with fastest behaviour (thus, resulting in higher histogram difference). In these cases, keeping a higher amount of keyframes is recommended. Figure 3.1 illustrates some $k = 5$ summarizations.

3.2 Time Equidistant Algorithm

The *time equidistant algorithm* (TEA) is based on equivalent frames in each shot of video by keeping keyframes in equal intervals in duration of shot.

Figure 3.1: Obtained hdiff $k = 5$ keyframes for different Montalbano RGB samples. First row shows an example for one sample belonging to 'vattene' gesture, second row for 'seipazzo', third row for 'combinato' and last row for 'ok'.

According to this method, the keyframe t_i, $i \in \{1, 2, \cdots, b_k\}$ in shot k is directly defined by the following equation:

$$t_i = \frac{i \times |SH_k|}{b_k} \tag{3.2}$$

where $|SH_k|$ denotes the number of frames of shot k and $\times$ denotes the nearest integer function. This is the simplest method for video synopsis creation, since it does not take into account visual changes. Figure 3.2 shows different $k = 5$ summarized sequences using this technique.

3.3 Sequential Distortion Minimization

Panagiotakis presents a new sequential technique for keyframe extraction called *Sequential Distortion Minimization (SeDiM)* [POM13] which considers temporal content variation, shot detection and minimizes content distortion with low computation cost, $O(N^2)$.

Figure 3.2: Obtained TEA $k = 5$ keyframes for mixed Montalbano RGB samples. First row shows an example for one sample belonging to 'vattene' gesture, second row for 'seipazzo', third row for 'combinato' and last row for 'ok'.

3.3.1 Distortion formulation

Panagiotakis defines distortion as follows; given a ratio between the temporal duration of the video synopsis and the initial video $\alpha \in [0, 1]$. Let N denote the number of frames of the original video. Then, resulting sequence will consist of $\alpha \cdot N$ representative keyframes. Let $C_i, i \in \{1, ..., N\}$ denote the visual descriptor of i-frame of original video. Let $S \subset \{1, ..., N\}$ denote the frames the final sequence. According to the problem definition, it holds that the number of keyframes $|S|$ is equal to $\alpha \cdot N$. Then, the distortion $D(\{1, ..., N\}, S)$ between the original video and the obtained keyframes is given by the following equation:

$$\begin{aligned} D(\{1, ..., N\}, S) = &\sum_{i=1}^{S(1)} d(i, S(1)) + \sum_{i=S(|S|)+1}^{N} d(i, S(|S|)) \\ &+ \sum_{i=S(1)+1}^{S(|S|)} min_{S(j) \leq i \leq S(j+1)}(d(i, S(j)), d(i, S(j+1))) \end{aligned} \tag{3.3}$$

where $d(i, S(j))$ is the distance between the visual descriptor of i-frame and $S(j)$-frame. $S(j)$ and $S(j+1)$ are two consecutive frames so that $S(j) \leq i \leq S(j+1)$, meaning that $S(j)$ is determined by the index i. First two sums relate to the special cases in which the i-th frame is located before the first keyframe $S(1)$ and after the last keyframe $S(|S|)$. The distortion defined by the summatory of visual distances between the frame of original video and the *closest* corresponding keyframe, can be considered as an extension of the definition of *Iso-Content Distortion Principle* [PDT09].

3.3.2 Method Description

The proposed method is divided into several steps. Given that descriptors based on image segmentation or camera motion estimation techniques are computationally expensive and provide results that are may not be accurate for all video content variation [PDT09] *Color Layout Descriptor* (CLD) is used instead. Then, in order to measure the content distance between two CLDs, a metric D is described:

$$\begin{aligned} D = &\sqrt{\sum_i (DY_i - DY'_i)^2} + \sqrt{\sum_i (DCb_i - DCb'_i)^2} \\ &+ \sqrt{\sum_i (DCr_i - DCr'_i)^2} \end{aligned} \tag{3.4}$$

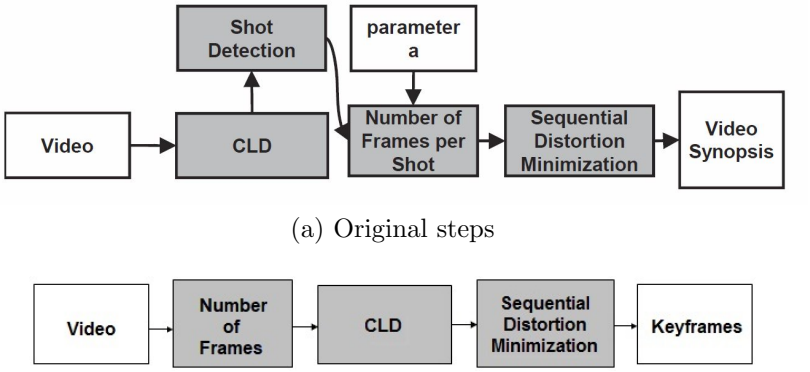
where (DY, DCb, DCr) are the i-th DCT coefficients of the respective color components [PDT09; PDT07].

Initially, CLDs are computed for all frames, then a shot detection step is performed. Based on the number of shots and the α ratio, the number of keyframes per shot is estimated. Finally, the video distortion is sequentially minimized selecting b_k frames for the k shot, so that the distortion between the original video and the summarization is sequentially minimized. The order of each keyframe corresponds to its significance on content description. Let CAN_k denote the set of candidate frames of shot k for the summarization. Initially, $CAN_k = SH_k$. Let S_k be the set of keyframes of shot k. Initially, $S_k = \emptyset$. For each shot k, iteratively select the frame f from CAN_k so that if included in set S_k, video distortion of shot k is minimized according to 3.5, then f is removed from CAN_k and added to S_k.

$$\begin{aligned} f &= argmin_{u \in CAN_k} \sum_{i \in SH_k} D(SH_k, S_k \cup u) \\ CAN_k &= CAN_k - f, \quad S_k = S_k \cup f \end{aligned} \tag{3.5}$$

When the number of keyframes of shot k become b_k, $CAN_k = \emptyset$. This process repeats until the number of keyframes is equal to $\alpha \cdot N$.

In our particular scenario, given that all videos were captured by a static device and, if needed, split into single videos per action and subject, it is obvious we don't need to perform any shot detection and its consequent estimation of the number of frames per shot based on CLD. Thus, we have simplified the proposed SeDiM algorithm to allow direct specification of the k number of keyframes to keep, also we are not interested on creating a video synopsis from the original videos but in obtaining a sequence of the k keyframes instead. For this purpose, the compression ratio α is computed based on the desired k and single-shot videos are assumed. Figure 3.3 illustrates different steps for the original version and our adaptation.



(a) Original steps

(b) Modified version

Figure 3.3: Schemes for the original (a) and proposed version (b) system architectures.

3.4 Content Equidistant Algorithm

The *content equidistant algorithm* (CEA) is inspired by [PDT09], using the proposed *iso-content principle* to estimate keyframes that are equidistant in video content. According to this method, keyframes $t_1, t_2, ..., t_{b_k}$ for the k

Figure 3.4: 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', third row for 'combinato' and last row for 'ok'.

shot are defined as 3.6:

$$\begin{aligned} m &\approx \sum_{u=1}^{t_1-1} d(u,u+1) \approx \sum_{u=t_1}^{t_2-1} d(u,u+1) \approx \cdots \approx \sum_{u=t_{b_k}}^{b_k-1} d(u,u+1) \\ m &= \frac{1}{b_k-1} \cdot \sum_{u=1}^{b_k-1} d(u,u+1) \end{aligned} \tag{3.6}$$

So, based on the measurement m, first we compute keyframe t_1, next t_2 and so on. In figure 3.5, several resulting sequences are shown.

Figure 3.5: Obtained CEA $k = 5$ keyframes for mixed Montalbano RGB samples. First row shows an example for one sample belonging to 'vattene' gesture, second row for 'seipazzo', third row for 'combinato' and last row for 'ok'.

Chapter 4

Proposed Method

The main aim of this work is to compare the performance of different keyframe sequences and its impact when used as deep learning features for action recognition in a multimodal approach. To this end, we extend 2DCNN, originally proposed for RGB data. In order to reduce the effect of noise in depth images, a registration (section 4.2) and denoising (section 4.1) steps are first performed as a simple pre-processing for the multimodal data. Next, we add scene flow as a new input stream to 2DCNN along with RGB and optical flow to perform a late fusion over all modalities using several video summarization approaches.

4.1 RGB and Depth Registration

Some datasets are not distributed with an accurate RGB-D alignment (figure 4.1). 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. For this, 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 map. Example results of the denoising and registration pre-processing procedures are shown in figure 4.2.

RGB-D registration aims to align (undistort) RGB and IR images by mapping depth pixels with color pixels. In [Alm+11], the author shows a method to estimate the optimal transformation $[R, T]$ for this purpose. Once they are obtained, the 3D metric position $(X_{ir}, Y_{ir}, Z_{ir})^T$ of the pixel, with the respect to the IR camera can be computed from the depth d_m, using the

Figure 4.1: RGB and depth super-positions for three samples of isoGD, showing noise and misalignment.

following equation:

$$(X_{ir}, Y_{ir}, Z_{ir})^T = \left(\frac{(x_{ir} c_{xir}) \cdot d_m(x_{ir}, y_{ir})}{f_{xir}}, \frac{(y_{ir} c_{yir}) \cdot d_m(x_{ir}, y_{ir})}{f_{yir}}, d_m(x_{ir}, y_{ir}) \right)^T \tag{4.1}$$

where x_{ir}, y_{ir} are the coordinates of the depth pixel in image, f_{xir}, f_{yir} the IR camera focal length (pixel size units), c_{xir}, c_{yir} the coordinates of the image center of IR camera, and d_m is depth in meters. Although IR and RGB cameras are separated by a small baseline, it is possible to determine the 6 DOF transform between them. Knowing the rotation R and translation T between the RGB and IR camera, we can project each 3D point on the color image and get its color. The mapping between color and depth images can be expressed as follows:

$$(X_{rgb}, Y_{rgb}, Z_{rgb})^T = R(X_{ir}, Y_{ir}, Z_{ir})^T + T \tag{4.2}$$

$$\begin{aligned} x_{rgb} &= \frac{(X_{rgb} \cdot f_{xrgb})}{Z_{rgb}} + c_{xrgb} \\ y_{rgb} &= \frac{(Y_{rgb} \cdot f_{yrgb})}{Z_{rgb}} + c_{yrgb} \end{aligned} \tag{4.3}$$

where x_{rgb}, y_{rgb} are the coordinates of the RGB pixel in the image, f_{xrgb}, f_{yrgb} the RGB camera focal length, c_{xrgb}, c_{yrgb} the image center, and d_m is depth in meters.

4.2 Denoising

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

Figure 4.2: 1st row: inpainting+HMF results of a depth sample from isoGD [Wan+16a] dataset. 2nd row: superposition before registration. 3rd row: superposition after registration.

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. In-painting reconstruction is then smoothed using a *hybrid median filter* (see 4.2.1) to reduce any pixel flickering between consecutive frames. 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.

4.2.1 Hybrid Median Filter

Although median filters preserve edges from digital images, they are also known to remove fine image detail such as lines. For example, 3×3 median filters remove lines 1 pixel wide, while in 5×5 filters, they remove lines 2 pixel wide [Dav04]. Depending on the application, this loss of information might become a major issue. In order to overcome this problem, in 1987 Nieminen et al. [NHN87] presented the "bi-directional" linear-median hybrid

filter (termed 2LH+), also known as hybrid median filter (HMF). HMF is a modification of the traditional median filter, consisting of a three step ranking operation to improve the detail preserving property. The basic idea behind the filter is for any given pixel of the image, apply a median operation several times, varying the window shape to rank different orientations. These window shapes are illustrated in figure 4.3.

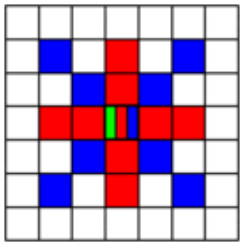


Figure 4.3: 5x5 hybrid median filter different shapes. Blue: cross-shape for horizontal and vertical pixels. Red: x-shape for the diagonals. Green: center pixel.

In a 5x5 pixel neighbourhood, the median values of the 45° neighbours forming a diagonal and the 90° neighbours forming a cross are compared with the central pixel. Then, the median value of that set is then saved as the new pixel value (figure 4.4).

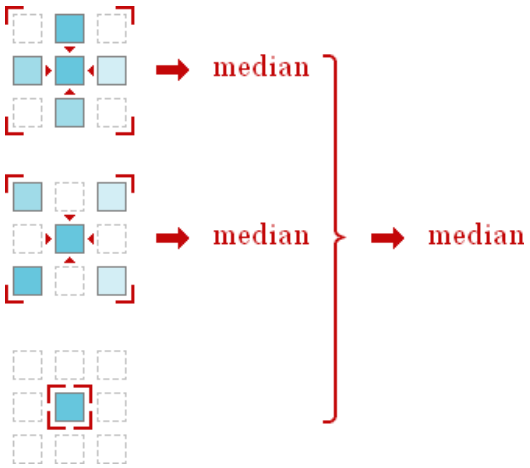


Figure 4.4: Hybrid median filter workflow.

The three step ranking operation does not impose a serious computational penalty as is the case of median filter. In fact, it allows for a more computationally efficient compared to the conventional median or K-nearest

neighbor averaging filters. Each of the ranking operations is for a much smaller number of values than used in a square region of the same size. Following the 5 pixel-wide neighbourhood example, it contains either 25 (in the square neighbourhood) which must be ranked in the traditional method. On the contrary, in the hybrid method each of the two groups contains only 9 pixels, and the final comparison involves only three values. Even with the additional logic and manipulation of values, the hybrid method is faster than the conventional median. This median filter overcomes the tendency of median and truncated median filters to erase lines which are narrower than the half width of the neighbourhood and to round corners.

4.3 Multimodal 2DCNN

Simonyan et al. [SZ14a] 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. This spatial classification is like an image classification architecture which is trained on single frame images ($224 \times 224 \times 3$). For the spatial network they used a pre-trained network on ImageNet [Den+09b]. Unlike the 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 motion implicitly. The original architecture consists of five convolutional layers, each of them followed by a pooling layer and three fully connected layers. Like [WQT15], 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). Different streams of the network are fed with different data 1) RGB: three channel frames; 2) optical flow: two channels using 10 stacked frames and 3) scene flow: three channel frames (same as the spatial stream).

For this, we introduce the **Multimodal 2DCNN** (MM2DCNN) by adding scene flow fields as a new input data, 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 fine-tuned from pre-trained models on UCF-101 dataset. Scene flow of each datasets is fine-tuned from the pre-trained model of its own RGB model.

4.3.1 Late Fusion

In order to perform the modality fusion we will take a simple approach, a weighted summatory of the class scores per each modality. That is, given M modalities, each sample has N feature arrays of size K classes, then, the final scores are: $S_f = \sum_i^N w_i S_i$, where weights w_i are to be optimized. Doing so, we are able to prioritize one score modality (RGB, optical flow or scene flow) over the others, depending on which one performed better on each dataset.

Chapter 5

Experimental results

In this chapter, we evaluate the proposed MM2DCNN using all different summarized sequences using different modalities on three public benchmark datasets: MSR Daily Activity [Wan+12], Montalbano V2 [Esc+13; Esc+14; EAG17] and IsoGD [Esc+14].

Every 2DCNN has been fine-tuned from spatial and temporal UCF-101 caffe models (available from here), using RGB, optical flow and scene flow frames. These models are trained using the strategy described in [Wan+15]. Model and training configurations are set according to the original report. Model parameters are initialized with the public available VGG-16 model and trained on the UCF-101 dataset.

Tables 5.1 and 5.5 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 (RGB-D synopsis). Finally, randomized-frame selection accuracy is also included for the sake of comparison.

For each dataset, we fine-tune RGB and optical flow model from trained models on UCF-101 (see 2.4.4), which was in its turn fine-tuned from a VGG-16 model composed of 13 convolutional layers and 3 fully-connected layers, trained on the ImageNet dataset [Den+09a]. However, it turned out training from scratch was not good enough for scene flow data, probably because we do not have enough input data to learn from. Thus, we fine-tune scene flow networks from our pre-trained models for RGB data of the same dataset. Final results of the fusions are compared with the state-of-the-art methods of action recognition using RGB-D data. In order to provide further details

about our classification results confusion matrices for fusion classifications are included in the appendix A.

5.1 Datasets

5.1.1 MSR Daily Activity 3D

This dataset consists of 16 actions captured with Microsoft Kinect [Wan+12]. Each of these samples is composed of RGB video and a sequence of depth images compressed into binary files. Contained actions include *'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. Such a low amount of samples may lead the models to overfit if not treated carefully. Moreover, subjects appear at different distances to the camera, and most of the actions involve object interactions. All these facts make this dataset very challenging. Some samples of this dataset are shown in figure 5.1. It is worth noting that there is a large inter-class similarity in this dataset, there are some actions which are very similar to each other (e.g. *use laptop* and *write on paper*), also the amount of motion found in these action samples is very small.

As for the experiments, half of the subjects shall be used to train the model while the other half for testing. In this particular case, we will use odd-numbered subjects (1, 3, 5, 7 and 9) for training, so the even-numbered ones (2, 4, 6, 8 and 10) will be kept for testing. Distribution of samples per class for train, validation and test are shown in figure 5.2.

5.1.2 Montalbano V2

Montalbano dataset is composed of 940 video samples, showing subjects performing 20 different italian gestures [Esc+13; Esc+14]. Labels include: *vattene, vieniqui', 'perfetto, 'furbo', 'cheduepalle', 'chevuoi', 'daccordo', 'seipazzo', 'combinato', 'freganiente', 'ok', 'cosatifarei', 'basta', 'prendere', 'noncenepiu', 'fame', 'tantotempo', 'buonissimo', 'messidaccordo' and 'sonostufo'*. Just as in 5.1.1, videos had been captured using a Kinect v1 device, therefore RGB and depth data are available. However, in this case each sample show a subject performing multiple gestures, as this dataset is also used for gesture detection. The dataset is distributed along with files providing gesture

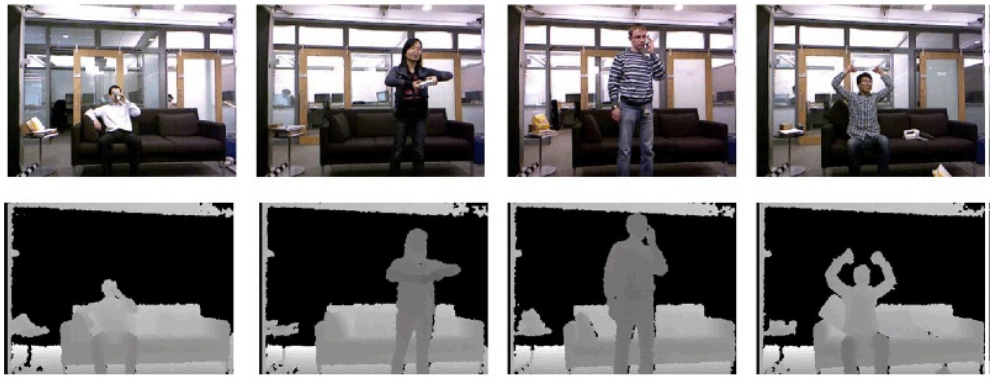
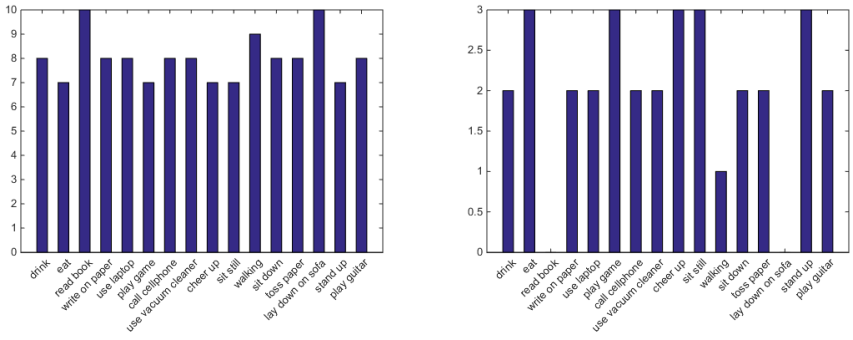
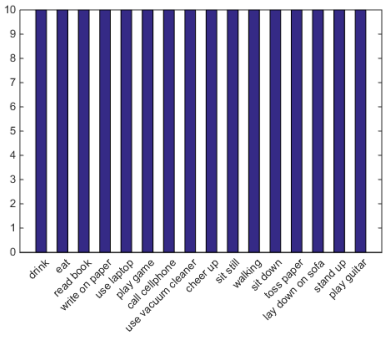


Figure 5.1: MSRDailyAct3D RGB and depth samples. col 1: *'drink'*; col 2: *'play game'*; col 3: *'call cellphone'*; col 4: *'cheer up'*.



(a) Train (b) Validation



(c) Test

Figure 5.2: Number of samples per class in MSR DailyAct3D.

indexation per video, specifying starting and ending frames per label. Therefore we have previously split those original video samples into single gesture

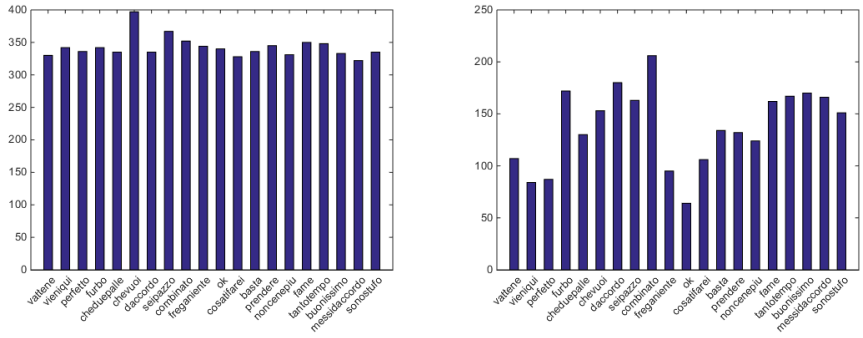
videos, leading to a total of 12575 samples. Figure 5.3 shows some samples of this dataset.

The dataset is already divided into three subsets: train, validation and test. Their sample distributions per class is illustrated in figure 5.4. Some samples for this dataset are shown in figure 5.3, as can be observed, the background is not always the same. Furthermore, subject's distance to the camera changes from sample to sample, some of the subjects appear completely while in other cases subjects fall partially outside the field of view. As this dataset has more classes, we could considered it become a harder challenge, however considering the large amount of samples, deep models should be able to better generalize each action.

Figure 5.3: Montalbano V2 RGB and depth samples (train sub-set).

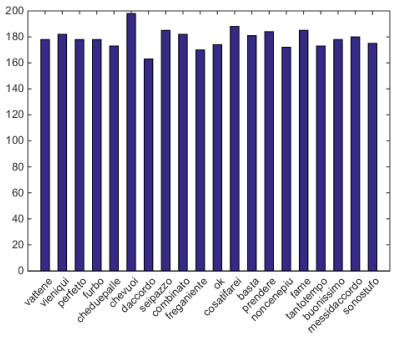
5.1.3 IsoGD

The ChaLearn LAP Isolated Gesture Dataset (IsoGD) [Esc+14] is the largest datasets evaluated in this work. This database includes 47.933 RGB-D gesture videos, in the format of RGB and depth separated videos captured using a Kinect device. Each RGB-D for all samples video represents one gesture only, and there are 249 gestures labels performed by 21 different individuals. The most crucial challenges of this dataset is the large number of classes, i.e., 249, compared to other action and gesture recognition datasets. The database has been divided to three sub-sets, these being mutually exclusive. Training dataset contains 35.878 samples, while validation has 5.784 samples and testing has 6.271 samples. Figure 5.5 shows some samples of this dataset. Figure 5.6 shows the amount of samples per class in each of these sub-sets.



(a) Train

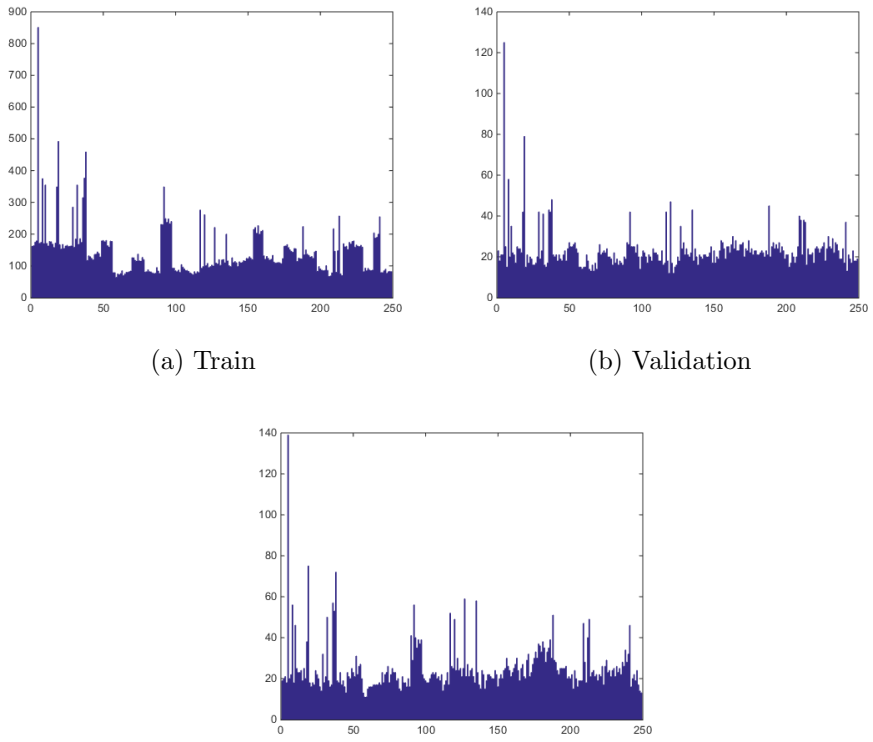
(b) Validation



(c) Test

Figure 5.4: Number of samples per class in Montalbano.

Figure 5.5: IsoGD RGB and depth samples (test partition). col 1: *'00001'*; col 2: *'00022'*; col 3: *'00053'*; col 4: *'00194'*.



(a) Train

(b) Validation

(c) Test

Figure 5.6: Number of samples per class in isoGD.

5.2 Experiments

In order to test how video summarization can affect the classification over different modalities at frame level, we have extracted $k = 14$ keyframe sequences from both RGB and depth videos per dataset. Performed experiments consist of testing each of the summarization sequences on every different deep model, trained using different modalities. Doing so, we intent to spot weather using keyframes can improve results compared to randomly selected frames, and if so, how much video summarization is able to increase the final accuracy. Also, we want to see if depth-based video summarization is able to hold similar results to RGB.

Besides, we include a hybrid-like summarization which we call as **RGB-D synopsis**. RGB-D synopsis is basically an ordered-frame concatenation of the separated summarizations of $k = 7$ RGB and $k = 7$ depth, given both keyframe sequences they are ordered by frame index and blend together. This hybrid summarization is due to test how much depth and RGB can contribute each other when combined.

5.2.1 MSR Daily Activity 3D

Results for every summarization technique (i.e. SeDiM, histogram difference, TEA and CEA) are presented in tables 5.1, 5.2, 5.3 and 5.4 accordingly. We see that in general, the result is not good. The most important problem related to this dataset for a deep model to learn, is the low number of samples. As explained in 5.1.1, MSR Daily only contains 320 samples, of which half of them are used for training. It is obvious that 160 actions are not enough for fine-tuning a network. In fact, in [Wan+15], the author points out this particular problem regarding the number of samples found in most of the action recognition datasets. Therefore, the accuracy achieved for MSR Daily is already expected to be not as good as it could be if more samples were used for training. Also, regarding the inter-class similarity, in confusion matrices (figures A.1, A.3, etc) it can be seen that most of the miss-classification comes from similar actions. This explains why *'write on paper'* is the most miss-classified action, being the one sharing more similarities with the others. The RGB model overfit on the texture of background and the human cloths and the motion model does not have enough information.

Among all different modalities, scene flow provides the best accuracy. We can see how keyframe selection for our hybrid strategy on RGB and depth data tends to get a better result. For this dataset, RGB background is more

cluttered than the others which could lead to overfitting. In contrast, as the background is found to be out of the IR range, most of this region is filled with non-determined values (zeroes) instead (see figure 5.1), meaning that depth images contain mainly foreground data, which enables for a cleaner scene flow extraction. We also see how optical flow outperforms RGB and the accuracy of scene flow is overall much better than optical flow, thanks to be considering real 3D information.

For this dataset, best accuracy is obtained by using the Time Equidistant (TEA) summarization technique 5.3. As for the late fusion, for this dataset we have determined experimentally that best results are obtained with weights $W = [0.2, 0.3, 0.5]$ for RGB, optical flow and scene flow scores, respectively, giving half of the relevance to scene flow scores, as it has proved to be the best modality among the rest of summarized sequences. Rest is distributed between optical flow and RGB modalities, giving a little more relevance to optical flow due to its slightly higher accuracy.

Despite of random frame selection yields same or even better accuracies over other keyframes sequences, we can see how through the late fusion we manage to outperform it using some summarized keyframe sequences techniques (e.g.in table 5.3).

Observe that in TEA we got two different keyframe sequences giving the best accuracy, these being depth and RGB-D synopsis keyframe sequences, even though RGB-D synopsis yields better results in both optical flow and RGB nets (55% as opposed to $\approx 53\%$), depth keyframes work better in the scene flow modality. Nevertheless, once the fusion is performed, it seems modalities somehow compensate each other in both cases, leading to the same result. Moreover, for this dataset histogram difference proved to be unable to outperform regular random frame selection in any keyframe category (table 5.2), this might be due to its inherent lack for full video coverage. As explained in section 3.1, if most of the activity of the video concentrates in consecutive frames, this summarization would only take into consideration those, resulting in a bad coverage and hence, a naive summarization. Also, it is important to note that for this dataset no in-painting step was performed given the amount of missing depth values. This can be considered a positive, in the sense that depth data includes less irrelevant data from background, but it also means that depth will keep much of its usual noise, even after being filtered with HMF.

Regarding the other video summarization methods, results of table 5.4 show

that CEA is able to outperform random frame selection when using the hybrid combination of depth and RGB keyframes, whereas in the rest of summarizations, it provides the same accuracy once fusion is applied. On the other hand, SeDiM gets the second best result (67.72%) when using keyframes from RGB (table 5.1). However, the rest of sequences (i.e. depth and RGB-D) provide worse accuracies compared to random selection.

Table 5.1: Accuracy for **SeDiM** on MSR Daily Activity 3D.

Model	RGB	Depth	RGB-D	Random
RGB	53.75	54.37	53.75	52.50
Opt. flow	55.63	52.50	55.00	53.75
Scene flow	62.50	59.84	61.42	62.20
Late Fusion	**67.72**	64.57	63.78	66.14

Table 5.2: Accuracy for **Histogram Difference** on MSR Daily Activity 3D.

Model	RGB	Depth	RGB-D	Random
RGB	51.88	50.62	51.88	52.50
Opt. flow	51.25	48.75	51.88	53.75
Scene flow	58.27	55.12	58.27	62.20
Late Fusion	62.20	63.78	65.35	**66.14**

Table 5.3: Accuracy for **TEA** on MSR Daily Activity 3D.

Model	RGB	Depth	RGB-D	Random
RGB	51.25	51.88	52.50	52.50
Opt. flow	53.12	53.12	55.00	53.75
Scene flow	59.84	59.84	56.69	62.20
Late Fusion	67.72	**68.50**	**68.50**	66.14

5.2.2 Montalbano V2

For this dataset, selected weights for the late fusion are $W = [0.65, 0.15, 0.2]$ for RGB, optical flow and scene flow scores. Just as for every other dataset, these weights have been tweaked experimentally until acquiring the best possible result. In this case, given that we already reach state-of-the-art accuracies just by using the RGB network itself, it is clear we must put most of the

Table 5.4: Accuracy for **CEA** on MSR Daily Activity 3D.

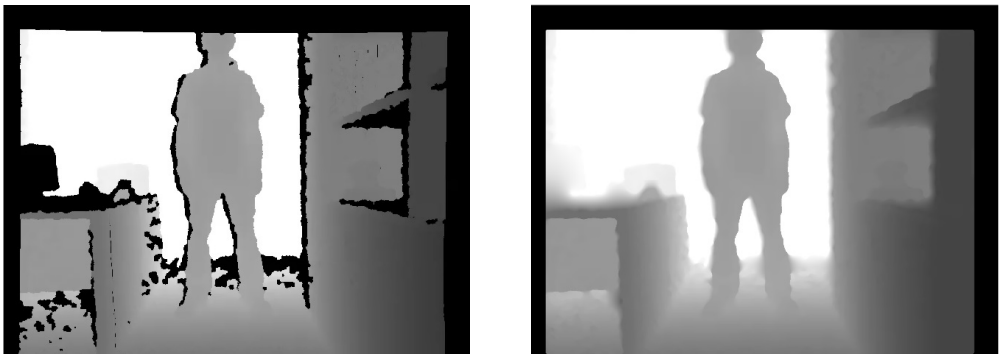
Model	RGB	Depth	RGB-D	Random
RGB	53.12	53.75	54.37	52.50
Opt. flow	56.87	53.12	54.37	53.75
Scene flow	59.84	57.48	59.06	62.20
Late Fusion	66.14	66.14	**66.93**	66.14

relevance to these scores. As can be observed from the reported accuracies found in tables 5.5, 5.6, 5.7 and 5.8, results for any of our different keyframe sequences in RGB modality stay around 96% accuracy and up. Therefore, we can not expect to improve it much by merging other modalities. Even with that, most of the fusions succeed in achieving slightly better results.

Regardless of having different backgrounds depending on the subject (see figure 5.3), they are more uniform compared to those found in MSR Daily. As explained in 5, RGB and optical flow networks are fine-tuned from a pre-trained network on UCF-101 and that pre-trained RGB network was first fine tuned from ImageNet dataset. Therefore, weights are more reliable for a dataset with simple background. Thus, for this dataset RGB model can be perfectly fine-tuned. Also, in this case we used more samples to train the network, leading to a much better learning in the RGB modality.

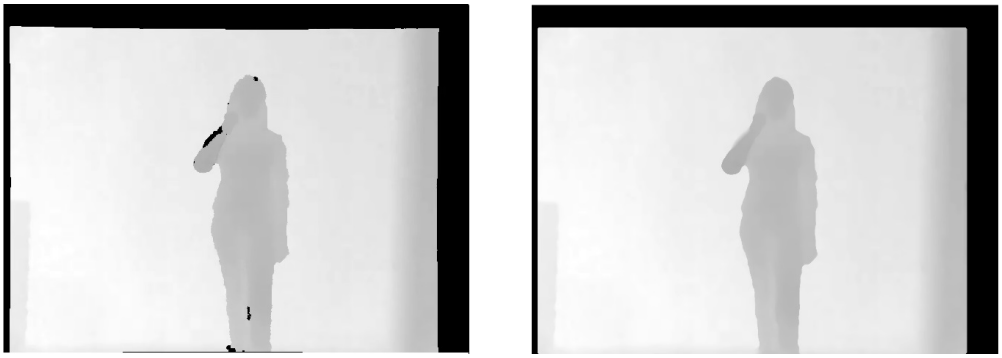
Regarding depth, while depth frames background are more regular than the previous dataset in some samples, in some scenes we still have cluttered backgrounds including different objects closer to the subjects. Also, in general depth samples are noisy. We tried to address this issue by means of in-painting and HMF, which managed to reduce the noise significantly (see figures 5.7, 5.8). Note that in this dataset, the IR receptor captures some black borders around depth frames, this may happen in Kinect devices with wrong calibration settings. In order to avoid this to affect the in-painting interpolation, we have extracted the region of interest (ROI) by removing those borders to then perform all denoising steps on this region only. Once the procedure is finished, the resulting depth frame is re-arranged in order to keep the original aspect, which is important in order to maintain a proper alignment between depth and RGB pairs.

Even with that, optical flow and scene flow can not compete with the results obtained by the RGB modality. Scene flow provides better results over optical flow throughout all summarized sequences. As for the best result, it



(a) Original (b) Denoised

Figure 5.7: Montalbano depth map reconstruction, including inpainting and HMF filtering.



(a) Original (b) Denoised

Figure 5.8: Montalbano depth map reconstruction, including inpainting and HMF filtering.

is acquired by using, again, the time equidistant technique along with depth data, yielding a total of 97.74% accuracy (table 5.7), which improves from the single RGB network 97.37% accuracy thanks to our multimodal fusion contribution. This result is followed by SeDiM with a total of 97.74% accuracy (table 5.5), taking into account that random selection already achieves 97.25%, it is not a significant difference given the additional cost for the summarization to this purpose.

As in MSR Daily, histogram difference approach does not manage to catch up with uniform random selection (table 5.6), just as CEA which this time (table 5.8) only manages to perform faintly better compared to the more

naive histogram difference.

Table 5.5: Accuracy for **SeDiM** on Montalbano.

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	66.79	66.31	65.15	64.72
Late Fusion	97.28	**97.56**	97.20	97.25

Table 5.6: Accuracy for the **Histogram Difference** on Montalbano.

Model	RGB	Depth	RGB-D	Random
RGB	96.10	96.47	96.10	97.06
Opt. flow	60.24	62.31	59.79	64.24
Scene flow	66.03	67.21	65.63	64.72
Late Fusion	96.46	96.89	96.27	**97.25**

Table 5.7: Accuracy for **TEA** on Montalbano.

Model	RGB	Depth	RGB-D	Random
RGB	97.51	97.48	97.37	97.06
Opt. flow	63.09	62.69	63.63	64.24
Scene flow	70.41	70.33	68.60	64.72
Late Fusion	97.68	97.70	**97.74**	97.25

Table 5.8: Accuracy for **CEA** on Montalbano.

Model	RGB	Depth	RGB-D	Random
RGB	96.50	96.77	96.55	97.06
Opt. flow	62.15	63.34	62.73	64.24
Scene flow	66.82	67.13	66.96	64.72
Late Fusion	96.94	97.19	97.17	**97.25**

5.2.3 IsoGD

The approach we followed to perform the experiments over this dataset varies with respect to the previous. In this case we have not been able to extend

MM2DCNN with scene flow as we have been not able to successfully train a network from this modality of data, probably because of the amount of noise contained in the scene flow frames obtained. However, in order to include the results for the rest of modalities in relation to video summarization, we decided to perform a late fusion for RGB and optical flow only. To this end, experimental weights showed that the best combination is achieved by using $W = [0.20, 0.80]$. Following the same table distribution, results are presented in tables 5.9, 5.10, 5.11 and 5.12.

Being this dataset the most challenging of this work, counting with 249 gestures to classify, we already expected to get worse results than in the previous. Even so, we manage to reach an acceptable accuracy thanks to the optical flow modality and its fusion with RGB. We can see how in this case video summarization works better than random selection. This may due to the overall longer videos, meaning more potential data loss when choosing non-discriminative enough in random selection, thus not holding a meaningful representation. Here, all different summarizations allow for a better classification than random selection in the RGB modality. In contrast, for the optical flow modality, random selection slightly outperforms all of our summarization approaches.

This dataset is sort of biased, in the sense that it provides far more samples for some of its classes. For that reason, the network tends to predict most of the samples as these classes. In general we tend to see that the model is able to predict fairly well those actions that we have more samples in training. That might be an issue, as if we recall from figure 5.6, there exist a huge difference between the amount of samples in some videos (peaks in the histogram, non-uniform data samples). Those cases correspond to actions for which our deep network seems to predict really well. This can also be seen in the IsoGD confusion matrices included on the appendix (figures A.9, A.10, A.12 and A.11), some of those examples are action 5, 18, etc. So, in this case, we suspect that having more video samples for the rest of actions would translate into a significantly better prediction rate.

Again, best summarization technique so far proved to be time equidistant algorithm (TEA), yielding $46,63\%$ accuracy when using depth keyframes and followed closely by its RGB counterpart. Surprisingly, this time RGB-D synopsis is outperformed by both of them, most probably due to lack of enough different frames from RGB and depth $k = 7$ sequences, in case of sharing similar frames, chances are that in longer videos, a higher amount of keyframes provide a better summarization.

Unlike the previous experiments, here absolute histogram difference manages to outperform not only random selection but also SeDiM (table 5.10).

Table 5.9: Accuracy for **SeDiM** on IsoGD.

Model	RGB	Depth	RGB-D	Random
RGB	26.37	27.32	27.47	21.50
Opt. flow	39.82	39.58	39.88	42.74
Late Fusion	42.95	43.43	**43.69**	43.66

Table 5.10: Accuracy for **Histogram Difference** on IsoGD.

Model	RGB	Depth	RGB-D	Random
RGB	27.11	29.36	27.48	21.50
Opt. flow	40.11	39.79	39.35	42.74
Late Fusion	42.93	**44.56**	43.09	43.66

Table 5.11: Accuracy for **TEA** on IsoGD.

Model	RGB	Depth	RGB-D	Random
RGB	30.93	30.38	30.24	21.50
Opt. flow	41.67	41.34	41.60	42.74
Late Fusion	**46.62**	**46.63**	46.35	43.66

Table 5.12: Accuracy for **CEA** on IsoGD.

Model	RGB	Depth	RGB-D	Random
RGB	26.68	29.29	27.98	21.50
Opt. flow	41.10	42.28	41.95	42.74
Late Fusion	43.52	**45.91**	45.08	43.66

5.3 Comparison

For all three datasets, our different tested strategies for keyframes selection (i.e., from different modalities) does not significantly affect the result. However, different summarization techniques have been proved to definitely have some impact in the final classification. We have managed to achieve

state-of-the-art accuracies for Montalbano V2 (table 5.14) and also, to get a good enough results in IsoGD, considering its complexity, even outperforming most of the reported methods of table 5.15. However, we are still far from reaching the state-of-the-art accuracy, which utilizes a more complex solution. While we are extending very deep two stream CNN with an additional stream, the state of the art technique goes for combination of two different networks; 1) convolutional two-stream consensus voting network (2SCVN) for both short-term and long-term RGB videos combined with 2) a 3D depth saliency ConvNet stream (3DDSN) to filter out distractions from the background.

As for MSR Daily Activity 3D, our results can not compete with those achieved by the state-of-the-art, of which we only outperformed EigenJoints [YZT12]. As we have explained, this might be caused due to lack of samples for train, which in some cases is not sufficient for such a deep model to learn. In these cases, taking a different approach by using an algorithm focused on hand-crafted local features, such as improved trajectories, along with Fisher vector representations may provide better solution [WS13].

Table 5.13: Performance comparison with state-of-the-art methods on MSR Daily Activity 3D.

Method	Accuracy
EigenJoints[YZT12]	58.10
MovingPose[ZLS13]	73.80
HON4D [OL13]	80.00
SSTKDes [AK17]	85.00
ActionLet [Wan+12]	**85.75**
MMDT [Asa+17a]	78.13
MM2DCNN	68.50

Table 5.14: Performance comparison with state-of-the-art methods on ChaLearn Montalbano.

Method	Accuracy
Fernando et al. [Fer+17]	75.30
Pigou et al. [Pig+15]	94.49
MMDT [Asa+17a]	85.66
MM2DCNN	**97.74**

Table 5.15: Performance comparison with state-of-the-art methods on ChaLearn IsoGD.

Method	Accuracy
NTUST	20.33
MFSK [WGL16]	24.19
MFSK+DeepID [Wan+16a]	23.67
XJTUfx	43.92
XDETVP-TRIMPS	50.93
TARDIS	40.15
ICT NHCI [Cha+16]	46.80
AMRL [Wan+16b]	55.57
2SCVN-3DDSN [Dua+16]	**67.19**
MM2DCNN	46.63

Chapter 6

Discussion

6.1 Conclusion

Our proposed method, multimodal 2DCNN (MM2DCNN) proves to succeed in improving the final classification accuracy by adding an additional stream to take advantage of the scene flow fields, which provide real 3D world motion information achieving state-of-the-art for Montalbano V2. On top of that, video summarization techniques ensure meaningful image selection from video sequences, increasing its variability and allowing for a general improvement during classification. We have seen how different versions of keyframe sequences work better depending on the dataset, the network modality and the summarization method applied. Given this we cannot claim any of them to be better than the rest. As for the summarization methods themselves, the simple TEA technique proved to be the best alternative in all the three datasets.

By fine-tuning the scene flow network from the RGB model, we find the learning process to be way faster compared to training it from scratch, furthermore it really improves our result. We find out that in the case of human action recognition, unlike in object recognition, fine-tuning one modality from another one improves the result. This kind of fine-tuning can be considered as a sort of regularization of the model.

The present work is part of a 2017 ICCV workshop publication, entitled *"Action Recognition from RGB-D Data: Comparison and fusion of spatio-temporal handcrafted features and deep strategies"* [Asa+17a].

6.2 Future work

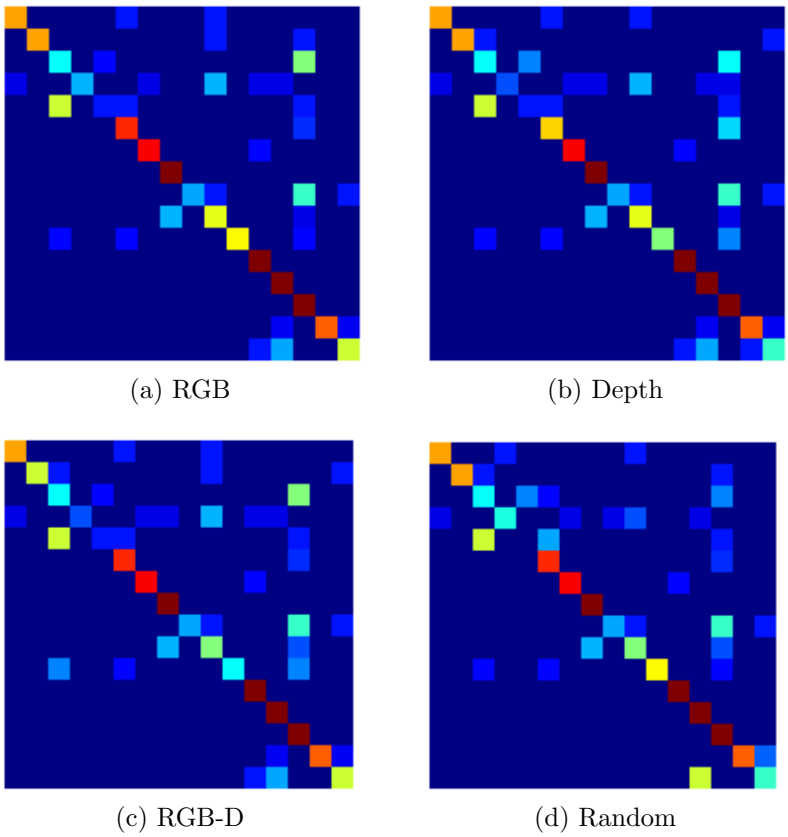
About video summarization, first thing to consider should be to vary k number of keyframes to keep depending on the mean video-length of each dataset. It is evident that retaining a higher amount of keyframes might be required for a dataset with longer video samples in order to get an improvement. In this work we have established $k = 14$ considering the length of all of three datasets overall, nevertheless different k per dataset might lead to better results. It is also important to mention that any of the video summarization methods used in this work is optimal, therefore other summarization algorithms might improve even more the reported results.

Regarding the fusion, weighted sum of scores may be substituted by learning a new model. This is, given M modalities, each sample of which has N feature arrays of size K, in this case, scores would be concatenated to create a new feature array of size $N \times K$. With this, two different models might be applied, random forests and multi-class SVM. Before training the algorithms, a PCA should be performed on the data to avoid overfitting. Middle fusion could be also good idea for the extension of this method, i.e. combining two networks from middle layers instead of training them separately.

As for our model contribution, combining hand-crafted features with deep learning can compensate for those cases in which MM2DCNN cannot generalize well [Asa+17a]. Beyond that, using 3DCNN instead of 2DCNN might improve the result. Using a large dataset such as NTU [Sha+16], can also be helpful. Training all the multimodal networks with this large dataset would allow us to use them later to fine-tune new models for smaller datasets with potential improvement.

Appendix A

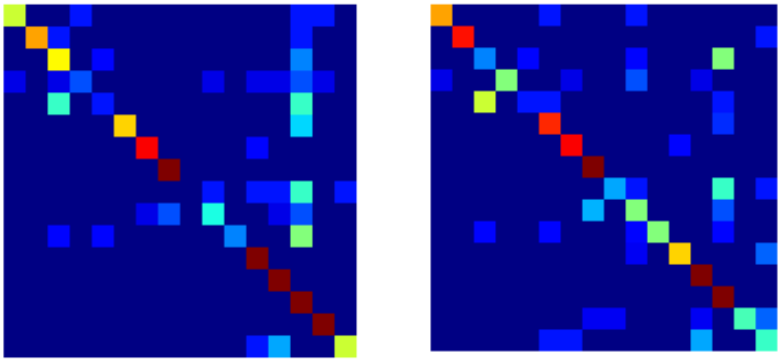
Confusion matrices



(a) RGB (b) Depth

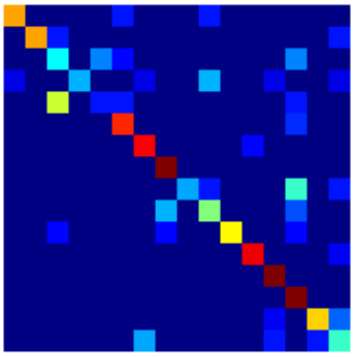
(c) RGB-D (d) Random

Figure A.1: Visual representation of confusion matrices for all MSR Daily Activity 3D **SeDiM** summarization fusions

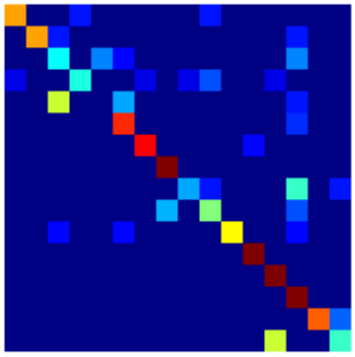


(a) RGB

(b) Depth

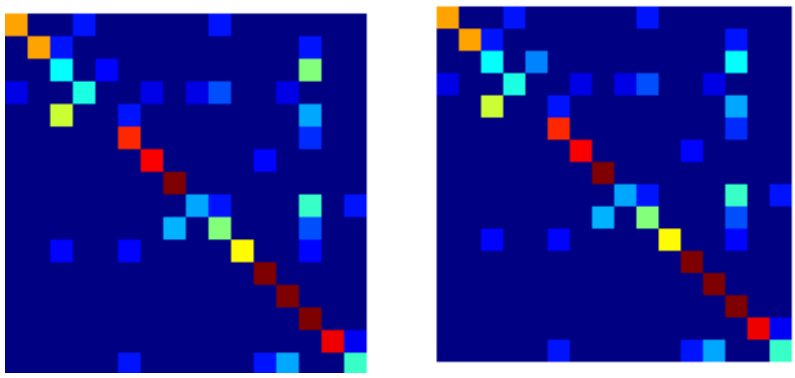


(c) RGB-D



(d) Random

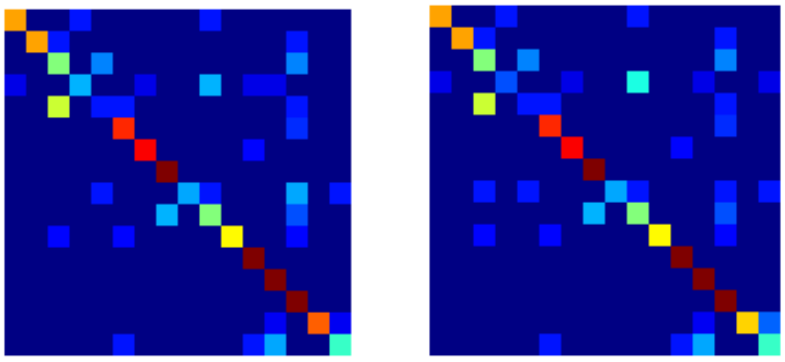
Figure A.2: Visual representation of confusion matrices for all MSR DailyActivity 3D **Histogram Difference** summarization fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

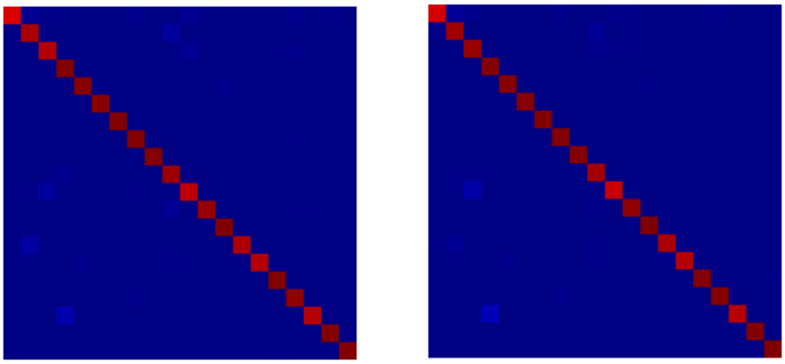
Figure A.3: Visual representation of confusion matrices for all MSR DailyActivity 3D **TEA** summarization fusions



(a) RGB (b) Depth

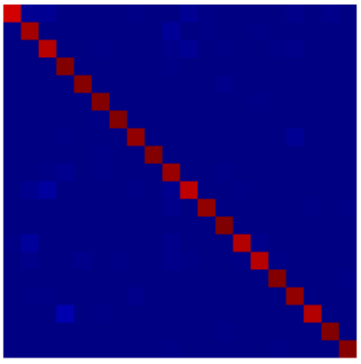
(c) RGB-D (d) Random

Figure A.4: Visual representation of confusion matrices for all MSR Daily Activity 3D **CEA** summarization fusions

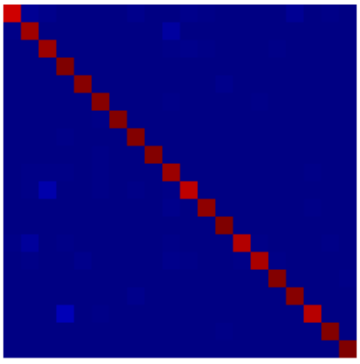


(a) RGB

(b) Depth

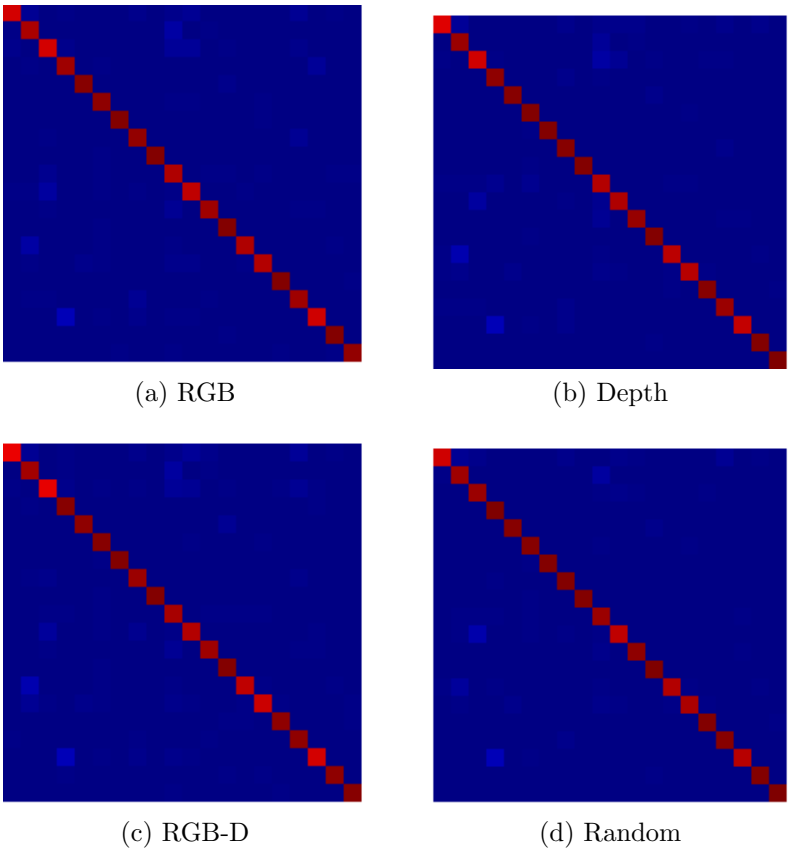


(c) RGB-D



(d) Random

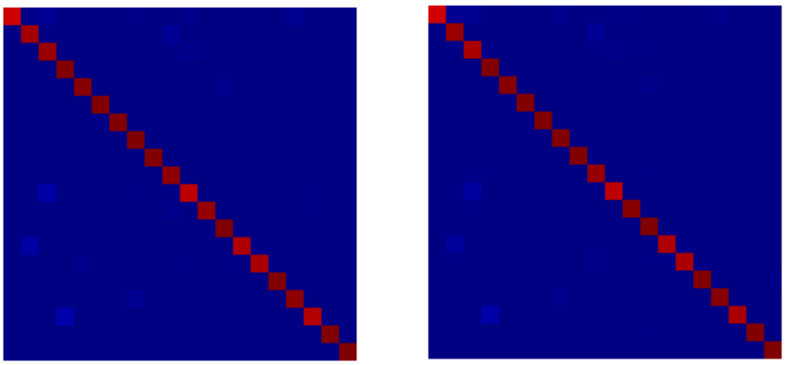
Figure A.5: Visual representation of confusion matrices for all Montalbano **SeDiM** summarization fusions



(a) RGB (b) Depth

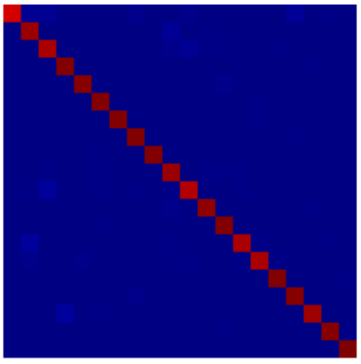
(c) RGB-D (d) Random

Figure A.6: Visual representation of confusion matrices for all Montalbano **Histogram Difference** summarization fusions

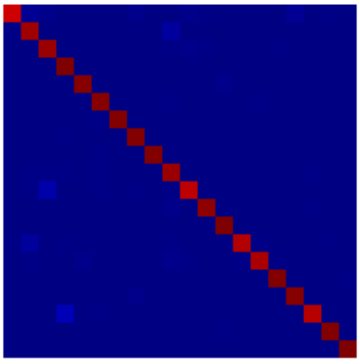


(a) RGB

(b) Depth

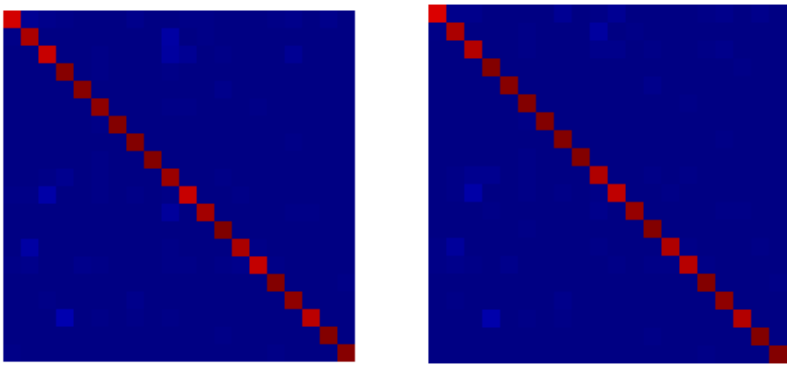


(c) RGB-D



(d) Random

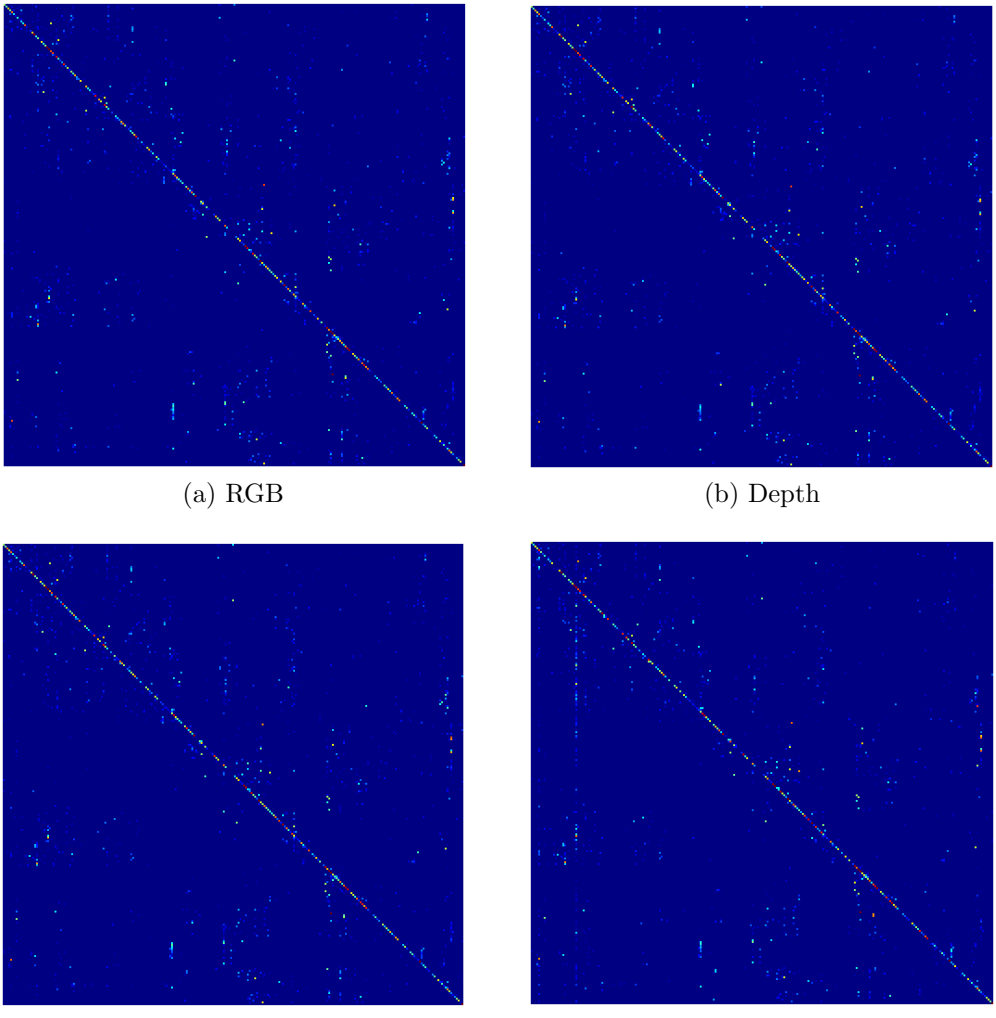
Figure A.7: Visual representation of confusion matrices for all Montalbano **TEA** summarization fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

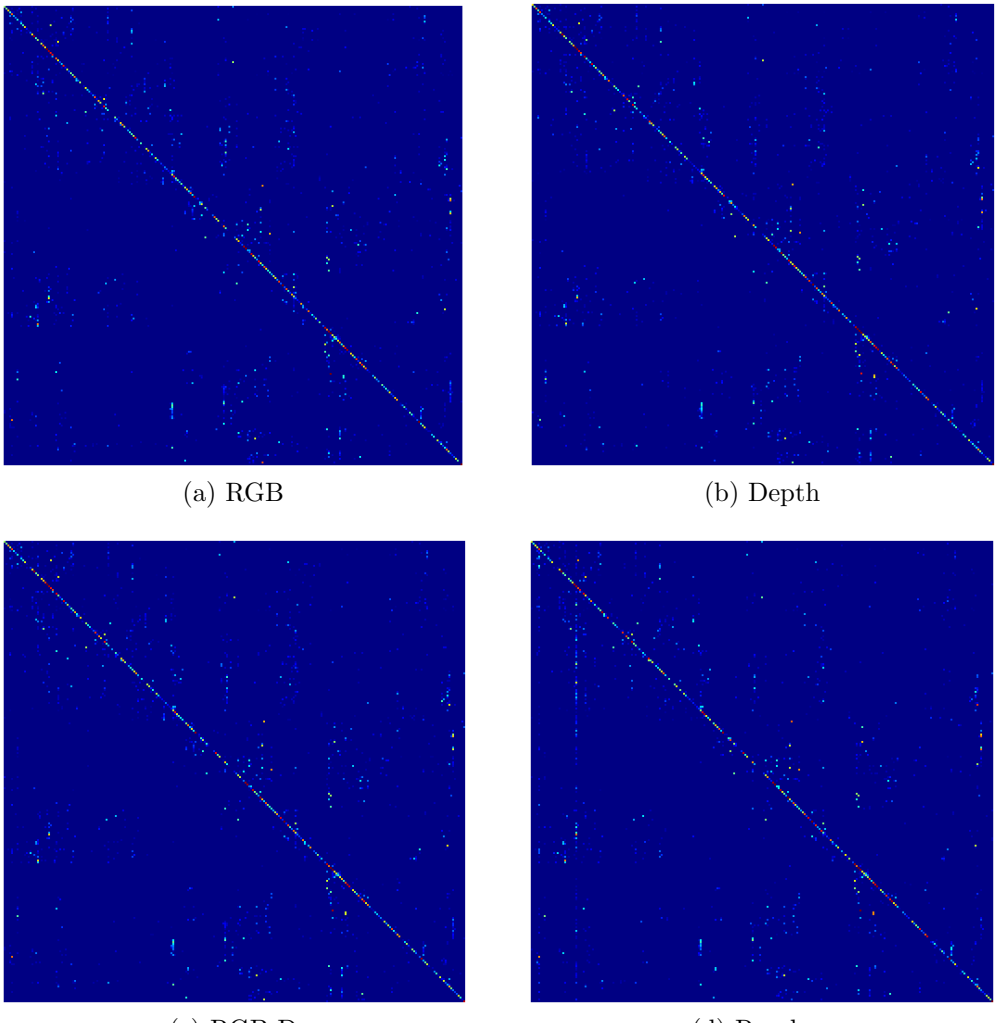
Figure A.8: Visual representation of confusion matrices for all Montalbano **CEA** summarization fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

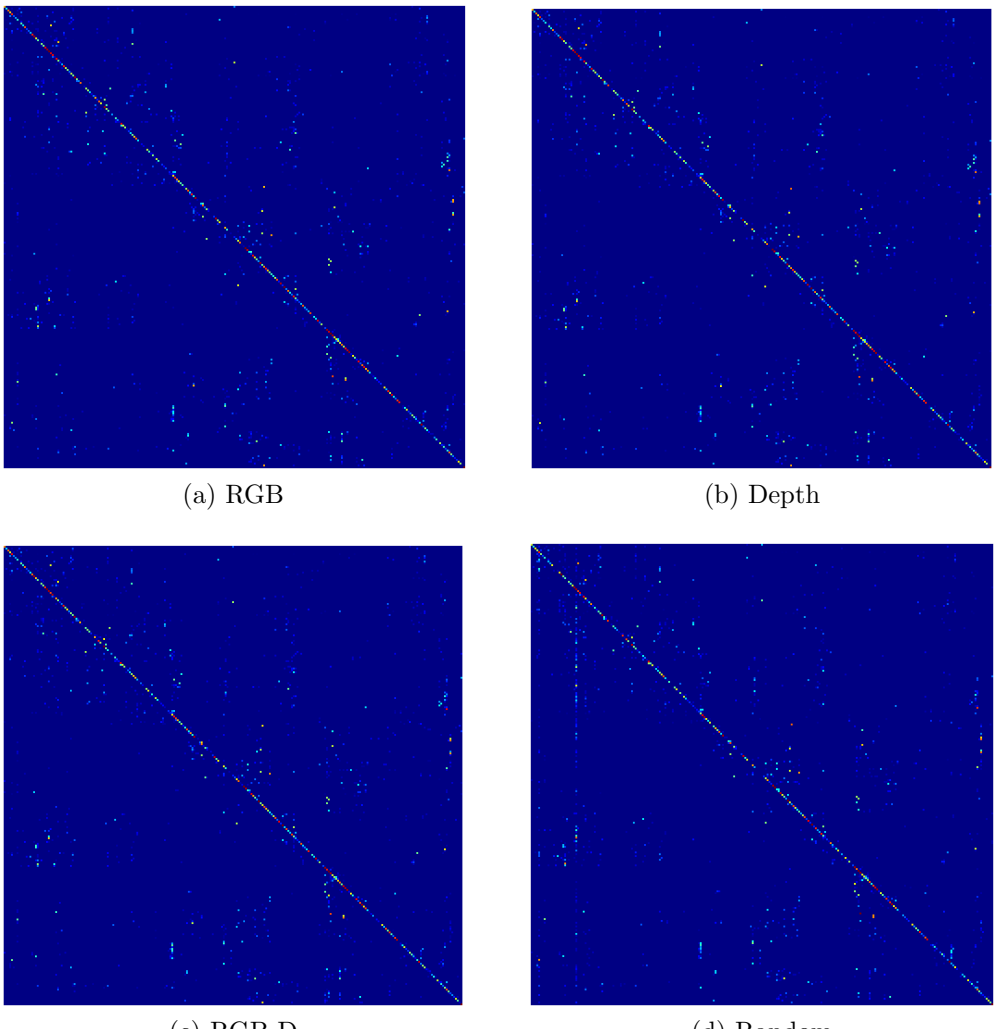
Figure A.9: Visual representation of confusion matrices for IsoGD **Histogram Difference** fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

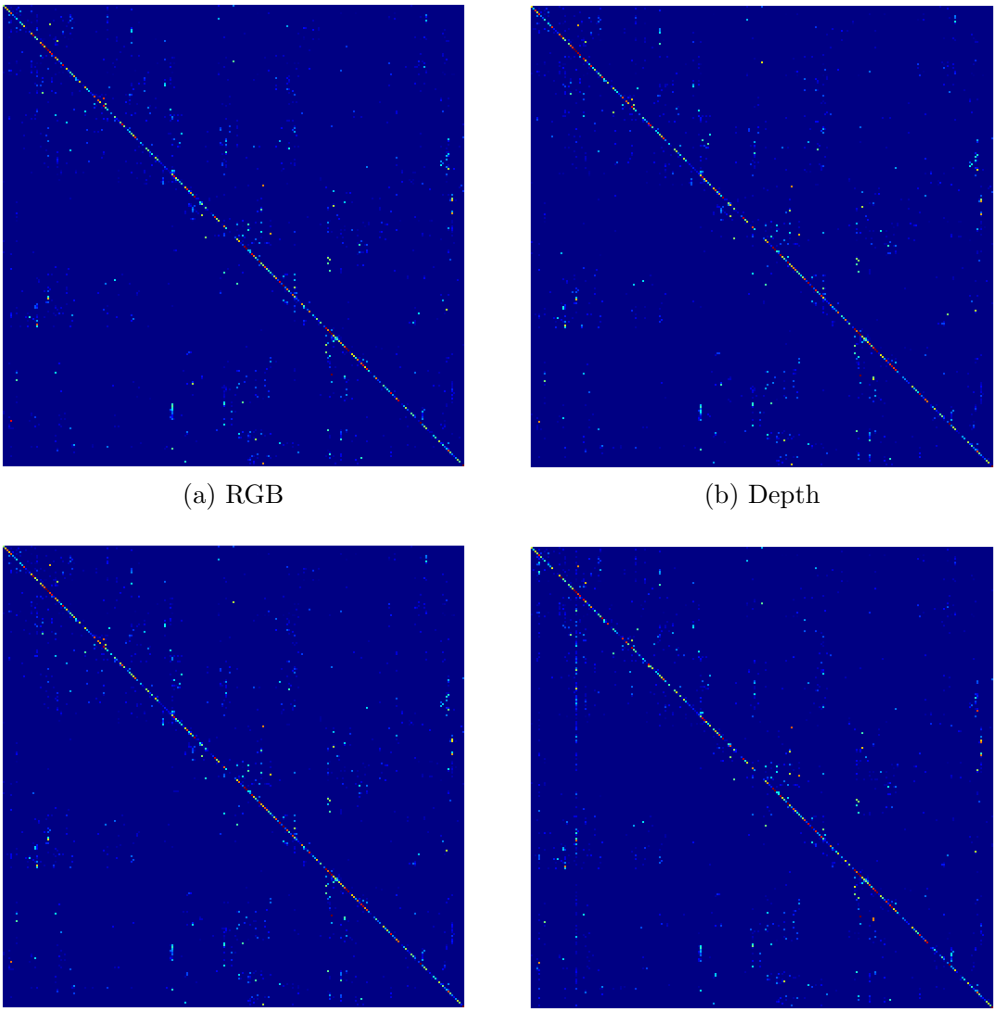
Figure A.10: Visual representation of confusion matrices for IsoGD **Histogram Difference** fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

Figure A.11: Visual representation of confusion matrices for IsoGD **TEA** fusions



(a) RGB (b) Depth

(c) RGB-D (d) Random

Figure A.12: Visual representation of confusion matrices for IsoGD **CEA** fusions

Bibliography

[HW68] David H Hubel and Torsten N Wiesel. "Receptive fields and functional architecture of monkey striate cortex". In: *The Journal of physiology* 195.1 (1968), pp. 215–243.

[NHN87] A. Nieminen, P. Heinonen, and Y. Neuvo. "A New Class of Detail-Preserving Filters for Image Processing". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9.1 (Jan. 1987), pp. 74–90. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1987.4767873.

[GM92] Melvyn A. Goodale and A. David. Milner. "Separate visual pathways for perception and action". In: *Trends in Neurosciences* 15.1 (1992), pp. 20–25.

[ZK01] 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*. Vol. 2. IEEE. 2001, pp. II–II.

[Dav04] E.R. Davies. *Machine Vision: Theory, Algorithms, Practicalities*. Signal Processing and its Applications. Elsevier Science, 2004. ISBN: 9780080473246. URL: https://books.google.es/books?id=uY-Z3vORugwC.

[LSK05] Zhu Li, G. M. Schuster, and A. K. Katsaggelos. "MINMAX optimal video summarization". In: *IEEE Transactions on Circuits and Systems for Video Technology* 15.10 (Oct. 2005), pp. 1245–1256. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2005.854230.

[PDT07] C. Panagiotakis, A. Doulamis, and G. Tziritas. "Equivalent Key Frames Selection Based on Iso-Content Distance and Iso-Distortion Principles". In: *Image Analysis for Multimedia Interactive Services, 2007. WIAMIS '07. Eighth International Workshop on*. June 2007, pp. 29–29. DOI: 10.1109/WIAMIS.2007.41.

[SAS07] 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*. ACM. 2007, pp. 357–360.

[KMS08] Alexander Klaser, Marcin Marszalek, and Cordelia Schmid. “A spatio-temporal descriptor based on 3d-gradients”. In: *BMVC 2008-19th British Machine Vision Conference*. British Machine Vision Association. 2008, pp. 275–1.

[Cha+09] Rizwan Chaudhry, Avinash Ravichandran, Gregory Hager, and Rene 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*. IEEE. 2009, pp. 1932–1939.

[Den+09a] J. Deng, W. Dong, R. Socher, L. J. Li, Kai Li, and Li Fei-Fei. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. June 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[Den+09b] 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*. IEEE. 2009, pp. 248–255.

[PDT09] C. Panagiotakis, A. Doulamis, and G. Tziritas. “Equivalent Key Frames Selection Based on Iso-Content Principles”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 19.3 (Mar. 2009), pp. 447–451. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2009.2013517.

[Alm+11] Luis Almeida, Francisco Vasconcelos, Joao P Barreto, Paulo Menezes, and Jorge Dias. “On-line incremental 3D human body reconstruction for HMI or AR applications”. In: (Sept. 2011).

[Mar11] Richard Marks. *System and method for providing a real-time three-dimensional interactive environment*. US Patent 8,072,470. June 2011.

[Wan+11] Heng Wang, Alexander Klaser, Cordelia Schmid, and Cheng-Lin Liu. “Action recognition by dense trajectories”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 3169–3176.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems 25*. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., 2012, pp. 1097–1105. URL: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[SZS12] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. "UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild". In: *CoRR* abs/1212.0402 (2012). URL: http://arxiv.org/abs/1212.0402.

[Wan+12] 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*. IEEE. 2012, pp. 1290–1297.

[YZT12] 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*. ACM. 2012, pp. 1057–1060.

[Esc+13] Sergio Escalera, Jordi Gonzalez, Xavier Baro, 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*. ACM. 2013, pp. 445–452.

[Ji+13] S. Ji, W. Xu, M. Yang, and K. Yu. "3D Convolutional Neural Networks for Human Action Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.1 (Jan. 2013), pp. 221–231. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2012.59.

[OL13] 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*. 2013, pp. 716–723.

[POM13] Costas Panagiotakis, Nelly Ovsepian, and Elena Michael. "Video Synopsis Based on a Sequential Distortion Minimization Method". In: *Computer Analysis of Images and Patterns: 15th International Conference, CAIP 2013, York, UK, August 27-29, 2013, Proceedings, Part I*. Ed. by Richard Wilson, Edwin Hancock,

Adrian Bors, and William Smith. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 94–101. ISBN: 978-3-642-40261-6. DOI: 10.1007/978-3-642-40261-6_11. URL: https://doi.org/10.1007/978-3-642-40261-6_11.

[VSR13] Christoph Vogel, Konrad Schindler, and Stefan Roth. "Piecewise rigid scene flow". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013, pp. 1377–1384.

[Wan+13a] Heng Wang, Alexander Klaser, Cordelia Schmid, and Cheng-Lin Liu. "Dense trajectories and motion boundary descriptors for action recognition". In: *International journal of computer vision* 103.1 (2013), p. 60.

[WS13] Heng Wang and Cordelia Schmid. "Action recognition with improved trajectories". In: *Proceedings of the IEEE international conference on computer vision*. 2013, pp. 3551–3558.

[Wan+13b] 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*. 2013, pp. 2858–2865.

[Yu+13] 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*. IEEE. 2013, pp. 1175–1180.

[ZLS13] 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*. 2013, pp. 2752–2759.

[Ami+14] 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*. IEEE. 2014, pp. 363–367.

[Esc+14] Sergio Escalera, Xavier Baro, Jordi Gonzalez, Miguel Angel Bautista, Meysam Madadi, Miguel Reyes, Victor Ponce-Lopez, Hugo Jair Escalante, Jamie Shotton, and Isabelle Guyon. "ChaLearn Looking at People Challenge 2014: Dataset and Results." In: *ECCV Workshops (1)*. 2014, pp. 459–473.

[Kar+14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. "Large-scale Video Classification with Convolutional Neural Networks". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2014.

[Lia+14] 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*. IEEE. 2014, pp. 556–559.

[SZ14a] Karen Simonyan and Andrew Zisserman. "Two-Stream Convolutional Networks for Action Recognition in Videos". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger. Curran Associates, Inc., 2014, pp. 568–576. URL: http://papers.nips.cc/paper/5353-two-stream-convolutional-networks-for-action-recognition-in-videos.pdf.

[SZ14b] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *CoRR* abs/1409.1556 (2014). URL: http://arxiv.org/abs/1409.1556.

[ZF14] Matthew D. Zeiler and Rob Fergus. "Visualizing and Understanding Convolutional Networks". In: *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*. Ed. by David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars. Cham: Springer International Publishing, 2014, pp. 818–833. ISBN: 978-3-319-10590-1. DOI: 10.1007/978-3-319-10590-1_53. URL: https://doi.org/10.1007/978-3-319-10590-1_53.

[AJM15] 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*. 2015, pp. 3908–3916.

[CN15] Sheena C V and N.K. Narayanan. "Key-frame Extraction by Analysis of Histograms of Video Frames Using Statistical Methods". English. In: *Procedia Computer Science* 70.Complete (2015), pp. 36–40. DOI: 10.1016/j.procs.2015.10.021.

[CLS15] Guilhem Cheron, Ivan Laptev, and Cordelia Schmid. "P-CNN: Pose-Based CNN Features for Action Recognition". In: *The IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[Jai+15] 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*. IEEE. 2015, pp. 98–104.

[Pig+15] Lionel Pigou, Aaron van den Oord, Sander Dieleman, Mieke Van Herreweghe, and Joni Dambre. "Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video". In: *International Journal of Computer Vision* (2015), pp. 1–10.

[Sze+15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going Deeper With Convolutions". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015.

[WQT15] 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*. 2015, pp. 4305–4314.

[Wan+15] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. "Towards good practices for very deep two-stream ConvNets". In: *CoRR* (2015).

[Cha+16] X. Chai, Z. Liu, F. Yin, Z. Liu, and X. Chen. "Two streams Recurrent Neural Networks for Large-Scale Continuous Gesture Recognition". In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. Dec. 2016, pp. 31–36. DOI: 10.1109/ICPR.2016.7899603.

[Dua+16] Jiali Duan, Shuai Zhou, Jun Wan, Xiaoyuan Guo, and Stan Z. Li. "Multi-Modality Fusion based on Consensus-Voting and 3D Convolution for Isolated Gesture Recognition". In: *CoRR* abs/1611.06689 (2016). URL: http://arxiv.org/abs/1611.06689.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[Sha+16] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis". In: *CoRR* abs/1604.02808 (2016). URL: http://arxiv.org/abs/1604.02808.

[WGL16] J. Wan, G. Guo, and S. Z. Li. "Explore Efficient Local Features from RGB-D Data for One-Shot Learning Gesture Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.8 (Aug. 2016), pp. 1626–1639. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2015.2513479.

[Wan+16a] 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: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2016.

[Wan+16b] P. Wang, W. Li, S. Liu, Z. Gao, C. Tang, and P. Ogunbona. "Large-scale Isolated Gesture Recognition using Convolutional Neural Networks". In: *2016 23rd International Conference on Pattern Recognition (ICPR)*. Dec. 2016, pp. 7–12. DOI: 10.1109/ICPR.2016.7899599.

[Asa+17a] Maryam Asadi-Aghbolaghi, Hugo Bertiche, Vicent Roig, Shohreh Kasaei, and Sergio Escalera. "Action Recognition from RGB-D Data: Comparison and fusion of spatio-temporal handcrafted features and deep strategies". In: *Chalearn Workshop on Action, Gesture, and Emotion Recognition: Large Scale Multimodal Gesture Recognition and Real versus Fake expressed emotions (ICCV17)*. Oct. 2017.

[Asa+17b] Maryam Asadi-Aghbolaghi, Albert Clapes, Marco Bellantonio, Hugo Jair Escalante, Victor Ponce-Lopez, Xavier Baro, 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*. IEEE. 2017, pp. 476–483.

[Asa+17c] Maryam Asadi-Aghbolaghi, Albert Clapes, Marco Bellantonio, Hugo Jair Escalante, Victor Ponce-Lopez, Xavier Baro, Isabelle Guyon, Shohreh Kasaei, and Sergio Escalera. "Deep Learning for Action and Gesture Recognition in Image Sequences: A Survey". In: *Gesture Recognition*. Springer, 2017, pp. 539–578.

[AK17] Maryam Asadi-Aghbolaghi and Shohreh Kasaei. "Supervised spatio-temporal kernel descriptor for human action recognition from RGB-depth videos". In: *Multimedia Tools and Applications* (2017), pp. 1–21.

[Che+17] 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". In: *Multimedia Tools and Applications* 76.3 (2017), pp. 4651–4669.

[EAG17] Sergio Escalera, Vassilis Athitsos, and Isabelle Guyon. "Challenges in Multi-modal Gesture Recognition". In: *Gesture Recognition*. Springer, 2017, pp. 1–60.

[Fer+17] Basura Fernando, Efstratios Gavves, Jose Oramas, Amir Ghodrati, and Tinne Tuytelaars. "Rank pooling for action recognition". In: *IEEE transactions on pattern analysis and machine intelligence* 39.4 (2017), pp. 773–787.