Multi-modal Gesture Recognition Challenge 2013: Dataset and Results

Sergio Escalera
Dept. Applied Mathematics, Universitat de Barcelona
Computer Vision Center, UAB
sergio@maia.ub.es

Jordi Gonzàlez
Dept. Computer Science, UAB, Barcelona
Computer Vision Center, UAB
poal@cvc.uab.es

Xavier Baró
EIMT at the Open University of Catalonia, Barcelona
Computer Vision Center, UAB
xbaro@uoc.edu

Miguel Reyes
Dept. Applied Mathematics, Universitat de Barcelona
Computer Vision Center, UAB
mreyese@gmail.com

Oscar Lopes
Computer Vision Center, Campus UAB, Barcelona
oscar.pino.lopes@gmail.com

Isabelle Guyon
ChaLearn, Berkeley, California
guyon@chalearn.org

Vassilis Athitsos
University of Texas
athitsos@uta.edu

Hugo J. Escalante
INAOE, Puebla, Mexico
hugojair@inaoep.mx

ABSTRACT

The recognition of continuous natural gestures is a complex and challenging problem due to the multi-modal nature of involved visual cues (e.g. fingers and lips movements, subtle facial expressions, body pose, etc.), as well as technical limitations such as spatial and temporal resolution and unreliable depth cues. In order to promote the research advance on this field, we organized a challenge on multi-modal gesture recognition. We made available a large video database of 13,858 gestures from a lexicon of 20 Italian gesture categories recorded with a Kinect^{TM} camera, providing the audio, skeletal model, user mask, RGB and depth images. The focus of the challenge was on *user independent multiple gesture learning*. There are no resting positions and the gestures are performed in continuous sequences lasting 1-2 minutes, containing between 8 and 20 gesture instances in each sequence. As a result, the dataset contains around 1.720.800 frames. In addition to the 20 main gesture categories, 'distracter' gestures are included, meaning that additional audio and gestures out of the vocabulary are included. The final evaluation of the challenge was defined in terms of the Levenshtein edit distance, where the goal was to indicate the real order of gestures within the sequence. 54 international teams participated in the challenge, and outstanding results were obtained by the first ranked participants.

ICMI'13, December 9–13, 2013, Sydney, Australia

http://dx.doi.org/10.1145/2522848.2532595.

1. INTRODUCTION

Predictive modelling competitions or *Challenges* have been fostering progress in Computer Vision in recent years. One of the most important challenges so far has been undoubtedly the PASCAL Visual Object Classes Challenges[1], organized by Everingham et al. [1], which have contributed to push the state-of-the-art on image classification, detection and segmentation.

These initial efforts concentrating on image understanding are now moving towards the analysis of video data, in which recognizing human activities in visual data has received much attention. Improving automatic recognition of human actions in visual data will allow the development of novel applications useful in surveillance and security, new generation of eHealth applications such as assisted living, the development of natural interfaces for Human Computer Interaction, and improved control in leisure scenarios.

As a result, several contests have been organized in activity recognition with applications in video surveillance, for example the VIRAT Action Recognition Challenge[2], the 3D human reconstruction and action recognition Grand Challenge [3], the Human Activities Recognition and Localization Competition [4] and the Contest on Semantic Description of Human Activities[5].

Following this trend, we organized a challenge called 'Multi-modal Gesture Recognition Challenge'[6] focusing on recognizing multiple gestures from a novel data set of videos recorded with a Microsoft Kinect™ camera. Kinect™ has revolutionized computer vision in recent years by providing an affordable 3D camera. The applications, initially driven by the game industry [5], have been rapidly diversifying and

[1] http://pascallin.ecs.soton.ac.uk/challenges/VOC
[2] http://www.umiacs.umd.edu/conferences/cvpr2011/ARC
[3] http://mmv.eecs.qmul.ac.uk/mmgc2013/
[4] http://liris.cnrs.fr/harl2012/
[5] http://cvrc.ece.utexas.edu/SDHA2010/
[6] http://gesture.chalearn.org

include video surveillance, computer interfaces, robot vision and control, and education [3].

Our previous one-shot learning challenge [2] was devoted to learning a gesture category from a single example of gesture coming from a limited vocabulary, using RGB and depth data. Our novel data set offers several gesture categories labeled from a dictionary of 20 Italian sign gesture categories. Several features make this new competition extremely challenging, including the recording of continuous sequences, the presence of distracter gestures (not included in the dictionary), the relatively large number of categories, the length of the gesture sequences, and the variety of users. To attack such a difficult problem, several modalities are provided in the data set, including audio, RGB, depth maps, user masks, and user skeletal model. The presentation of the data set and the results obtained in the Mutimodal Gesture Recognition Challenge are explained in the next sections.

2. PROBLEM SETTING AND DATA

The focus of the challenge is on *multiple instance, user independent learning of gestures from multi-modal data*, which means learning to recognize gestures from several instances for each category performed by different users, drawn from a vocabulary of 20 gesture categories. A gesture vocabulary is a set of unique gestures, generally related to a particular task. In this challenge we focus on the recognition of a vocabulary of 20 Italian cultural/anthropological signs, see Figure 1 for one example of each Italian gesture.

In all the sequences, a single user is recorded in front of a Kinect^{TM}, performing natural communicative gestures and speaking in fluent Italian. The main characteristics of the dataset of gestures are:

• 13.858 gesture samples recorded with the Kinect^{TM} camera, including audio, skeletal model, user mask, RGB, and depth images.
• RGB video stream, 8-bit VGA resolution (640×480) with a Bayer color filter, and depth sensing video stream in VGA resolution (640×480) with 11-bit. Both are acquired in 20 fps on average.
• Audio data is captured using Kinect^{TM} 20 multi-array microphone.
• A total number of 27 users appear in the data set.
• The data set contains the following number of sequences, development: 393 (7.754 gestures), validation: 287 (3.362 gestures), and test: 276 (2.742 gestures), each sequence lasts between 1 and 2 minutes and contains between 8 and 20 gesture samples, around 1.800 frames. The total number of frames of the data set is 1.720.800.
• All the gesture samples belonging to 20 main gesture categories from an Italian gesture dictionary are annotated at frame level indicating the gesture label.
• 81% of the participants were Italian native speakers, while the remaining 19% of the users were not Italian, but Italian-speakers.
• All the audio that appears in the data is from the Italian dictionary. In addition, sequences may contain distracter words and gestures, which are not annotated since they do not belong to the main dictionary of 20 gestures.

This dataset, available at `http://sunai.uoc.edu/chalearn`, presents various features of interest as listed in Table 1.

Table 1: Easy and challenging aspects of the data.

Easy
Fixed camera
Near frontal view acquisition
Within a sequence the same user
Gestures performed mostly by arms and hands
Camera framing upper body
Several available modalities: audio, skeletal model, user mask, depth, and RGB
Several instances of each gesture for training
Single person present in the visual field
Challenging
Within each sequence:
Continuous gestures without a resting pose
Many gesture instances are present
Distracter gestures out of the vocabulary may be present in terms of both gesture and audio
Between sequences:
High inter and intra-class variabilities of gestures in terms of both gesture and audio
Variations in background, clothing, skin color, lighting, temperature, resolution
Some parts of the body may be occluded
Different Italian dialects

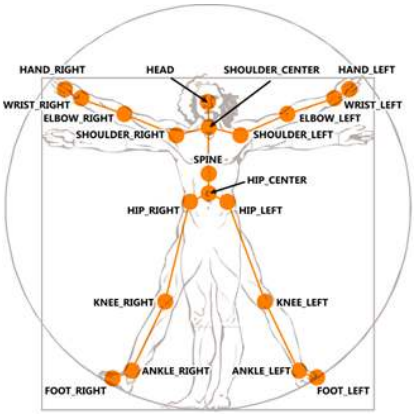


Figure 2: Skeleton joint positions.[7]

2.1 Data format and structure

We provide the X_audio.ogg, X_color.mp4, X_depth.mp4, and X_user.mp4 files containing the audio, RGB, depth, and user mask videos for a sequence X, respectively, see Figure 3. We also provide a script in order to export the data in Matlab, which contains the following Matlab structures:

- **NumFrames:** Total number of frames.
- **FrameRate:** Frame rate of the video in fps.
- **Audio:** structure that contains WAV audio data.
  - **y:** Audio Data
  - **fs:** Sample rate for the data.
- **Labels:** Structure that contains the data about labels contained in the sequence, sorted in order of appearance. The labels considered to the 20 gesture categories as shown in Figure 1.
  - **Name:** The name given to this gesture.

[7]Image from http://msdn.microsoft.com/en∼us/library/microsoft.kinect.jointtype.aspx


(1) *Vattene*


(2) *Viene qui*


(3) *Perfetto*


(4) *E un furbo*


(5) *Che due palle*


(6) *Che vuoi*


(7) *Vanno d'accordo*


(8) *Sei pazzo*


(9) *Cos hai combinato*


(10) *Nonme me friega niente*


(11) *Ok*


(12) *Cosa ti farei*


(13) *Basta*

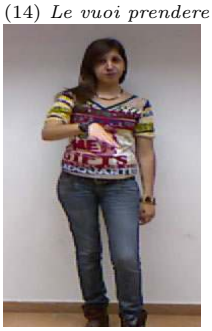

(14) *Le vuoi prendere*


(15) *Non ce ne piu*


(16) *Ho fame*


(17) *Tanto tempo fa*


(18) *Buonissimo*


(19) *Si sono messi d'accordo*


(20) *Sono stufo*

Figure 1: Data set gesture categories.

1. vattene
2. vieniqui
3. perfetto
4. furbo
5. cheduepalle
6. chevuoi
7. daccordo
8. seipazzo
9. combinato
10. freganiente
11. ok
12. cosatifarei
13. basta
14. prendere
15. noncenepiu
16. fame
17. tantotempo
18. buonissimo
19. messidaccordo
20. sonostufo

- **RGB:** This matrix represents the RGB color image, expressed in 8-bit VGA resolution (640×480) with a Bayer color filter.
- **Depth:** The Depth matrix contains the pixel-wise z component, VGA resolution (640×480) represented with 11 bits. The value of depth is expressed in millimeters.
- **UserIndex:** The user index matrix represents the player index of each depth pixel. A non-zero pixel value means that a tracked subject occupies the pixel, and a value of 0 denotes that no tracked subject occupies the pixel.
- **Skeleton:** An array of Skeleton structures is contained within a Skeletons array. It contains the joint positions, and bone orientations comprising a skeleton. The format of a Skeleton structure is:
  - **JointType:** Skeleton joints that make up a tracked skeleton. Figure 2 visualizes these joint types.

1. HipCenter
2. Spine
3. ShoulderCenter
4. Head
5. ShoulderLeft
6. ElbowLeft
7. WristLeft
8. HandRight
9. HipLeft
10. KneeLeft
11. AnkleLeft
12. FootLeft
13. HipRight
14. KneeRight
15. AnkleRight
16. FootRight

- **JointPosition:** It contains the joint positions in the next three coordinates:
  - **WorldPosition:** The world coordinates position structure represents the global position of a tracked joint. The format is **X, Y, Z** which represents the x, y, and z components of the subject´s global position (in millimeters).
  - **PixelPosition:** The pixel coordinates position structure represents the position of a tracked joint. The format of the Position structure is **X, Y** which represent the x and y components of the joint location over the RGB map (in pixels coordinates).
  - **WorldRotation:** The world rotation structure contains the orientations of skeletal bones in terms of absolute transformations and is formed by a 20×4 matrix, where each row contains the W, X, Y, Z values of the quaternion related to the rotation. The world rotation structure provides the orientation of a bone in the 3D camera space. The orientation of a bone is relative to the child joint and the Hip Center joint still contains the orientation of the player/subject.

3. PROTOCOL AND EVALUATION

The timeline of the competition was as follows:

• **April 30th, 2013**: Beginning of the challenge competition, release of first data examples.
• **May 20th, 2013**: Full release of training and validation data. Training data with ground truth labels.
• **August 1st, 2013**: Encrypted Final evaluation data and ground truth labels for the validation data are made available.
• **August 15th, 2013**: End of the challenge competition. Deadline for code submission. The organizers start the code verification by running it on the final evaluation data and obtaining the team scores.
• **August 25th, 2013**: Deadline for fact sheets.
• **September 1st, 2013**: Release of the verification results to the participants for review.

The challenge consisted of two main components: a development phase (April 30th to Aug 1st) and a final evaluation phase (Aug 2nd to Aug 15th). The submission and evaluation of the challenge entries was via the *Kaggle* platform [8]. The official participation rules were provided on the website of the challenge. In addition, publicity and news on the ChaLearn Multi-modal Gesture Recognition Challenge were published in well-known online platforms, such as LinkedIn, Facebook, Google Groups and the ChaLearn website.

During the development phase, the participants were asked to build a system capable of learning from several gesture samples a vocabulary of 20 Italian sign gesture categories. To that end, the teams received the development data to train and self-evaluate their systems. In order to monitor their progress they could use the validation data for which the labels were not provided. The prediction results on validation data could be submitted online to get immediate feed-back. A real-time leaderboard showed to the participants their current standing based on their validation set predictions.

During the final phase, labels for validation data are published and the participants performed similar tasks as those performed in previous phase, using the validation data and training data sets in order to train their system with more gesture instances. The participants had only few days to train their systems and upload them. The organizers used the final evaluation data in order to generate the predictions and obtain the final score and rank for each team. At the end, the final evaluation data was revealed, and authors submitted their own predictions and fact sheets to the platform.

[8] https://www.kaggle.com/c/multi-modal-gesture-recognition

4. EVALUATION METRIC

For each unlabeled video, the participants were instructed to provide an ordered list of labels R corresponding to the recognized gestures. We compared this list with the truth labels T i.e. the prescribed list of gestures that the user had to play during data collection. We computed the Levenshtein distance $L(R,T)$, that is the minimum number of edit operations (substitution, insertion, or deletion) that one has to perform to go from R to T (or vice versa). The Levenhstein distance is also known as 'edit distance'. For example: $L([124],[32]) = 2$, $L([1],[2]) = 1$, $L([222],[2]) = 2$. The overall score is the sum of the Levenshtein distances for all the lines of the result file compared to the corresponding lines in the truth value file, divided by the total number of gestures in the truth value file. This score is analogous to an error rate. For simplicity, in what follows, we call it Error Rate, although it can exceed 1.0. A public score appeared on the leaderboard during the development period and was based on the validation data. Subsequently, a private score for each team was computed on the final evaluation data released at the end of the development period, which was not revealed until the challenge was over. The private score was used to rank the participants and determine the prizes.

5. RESULTS

The challenge attracted high level of participation, with a total of 54 teams and near 300 total number of entries. This is a good level of participation for a computer vision challenge requiring very specialized skills. Finally, 17 teams successfully submitted their prediction in final test set, while providing also their code for verification and summarizing their method by means of a fact sheet questionnaire.

After verifying the codes and results of the participants, the final scores of the top rank participants on both validation and test sets were made public: these results are shown in Table 2, where winner results on the final test set are printed in bold. In the end, the final error rate on the test data set was around 12%.

5.1 Statistics on the results

Figure 5 shows the correlation of the validation and test error scores obtained by the top ranked participants of the challenge. One can see that most of them obtain similar results in both sets. However, there exist a few outliers that show non-correlated results among validation and test scores. Most of the participants that achieved top positions in test scores also achieved high recognition rates on the validation set. However, some participants that achieved low

RGB Depth User mask Skeletal model

Figure 3: Different data modalities of the provided data set.

Table 2: Top rank results on validation and test sets.

TEAM	Validation score	Test score
IVA MM	0.20137	**0.12756**
WWEIGHT	0.46163	**0.15387**
ET	0.33611	**0.16813**
MmM	0.25996	0.17215
PPTK	0.15199	0.17325
LRS	0.18114	0.17727
MMDL	0.43992	0.24452
TELEPOINTS	0.48543	0.25841
CSI MM	0.32124	0.28911
SUMO	0.49137	0.31652
GURU	0.51844	0.37281
AURINKO	0.31529	0.63304
STEVENWUDI	1.43427	0.74415
JACKSPARROW	0.86050	0.79313
JOEWAN	0.13653	0.83772
MILAN KOVAC	0.87835	0.87463
IAMKHADER	0.93397	0.92069

error on validation, considerably increased their error on the test set. It could be mainly related to overfitting of method parameters on the validation data, which could not be able to generalize to the variability of new users present on the test set. On the other hand, the final scores on the test set are in general lower than in the validation set. This may be produced because more data is trained by the participants when testing for final evaluation set, and thus, if properly trained, this small final improvement is expected. The best public score on the validation set achieved during the period of the challenge taking into account all team submissions over time is summarized in Figure 4.

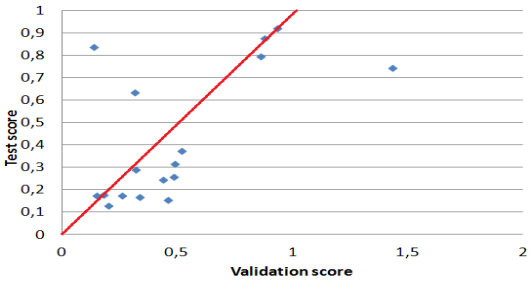


Figure 5: Correlation results among the validation and test results of the top ranked participants.

In Figure 6, we show the histograms of validation and test scores based on the best score achieved by each team on each of the two sets. One can see that the scores on the validation set become more sparse, and the teams that finally submitted their predictions to the test set, except for two cases, achieved scores inferior to 1 Levenstein score error.

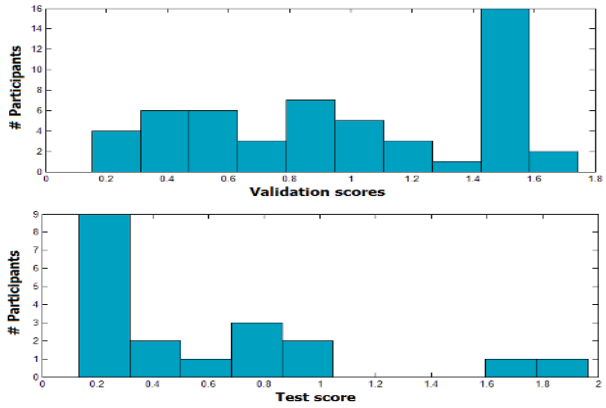


Figure 6: Validation and test scores histograms.

5.2 Fact sheets

We asked the participants to fill out a survey about the methods employed. The top ranked 17 test participants filled out this survey. We briefly summarize the results next.

From the questions within the survey, the most relevant aspects for the challenge where: the modalities considered for the methods, the temporal segmentation methodology applied, the considered fusion strategy, as well as the recognition techniques considered. We additionally comment the programming language used by the participants. The information about modalities, segmentation strategy, fusion, classifier, and programming language are summarized in Figures 7, 8, 9, 10, and 11, respectively. The details of each particular team strategy are shown in Table 3.

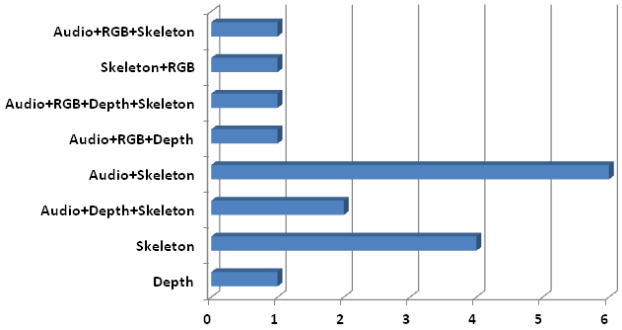


Figure 7: Modalities considered.

Looking at the considered modalities in Figure 7, one can see that none of the teams considered only audio information, and most of them used multiple modalities for describing the data. In particular, skeleton was the most considered

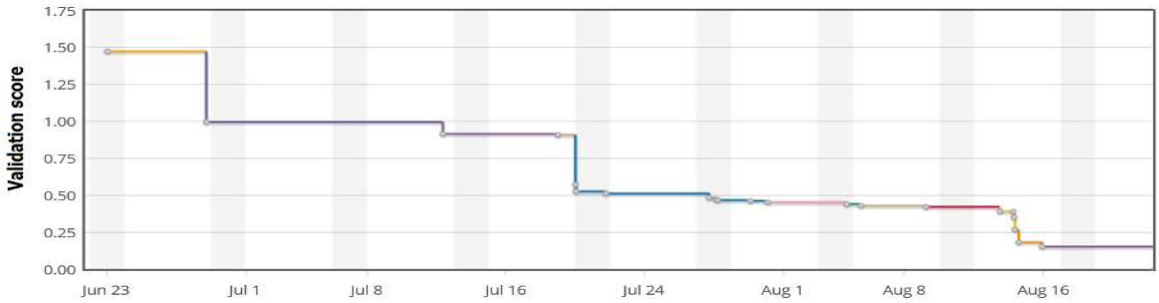


Figure 4: Best public score obtained in the validation set during the Challenge.

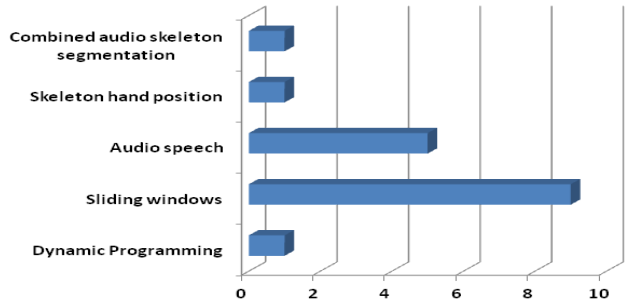


Figure 8: Segmentation strategy.

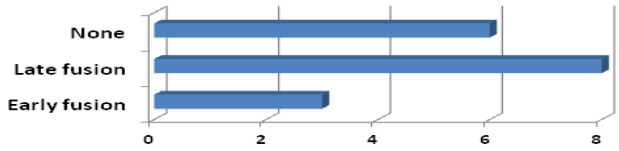


Figure 9: Fusion strategy.

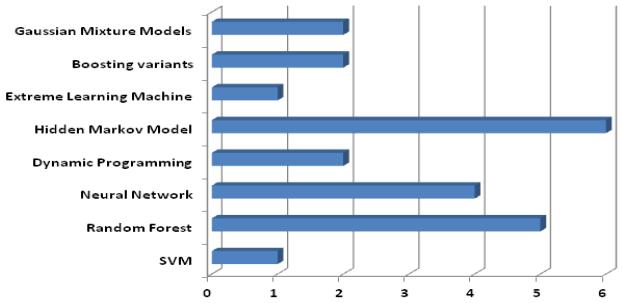


Figure 10: Learning strategy.

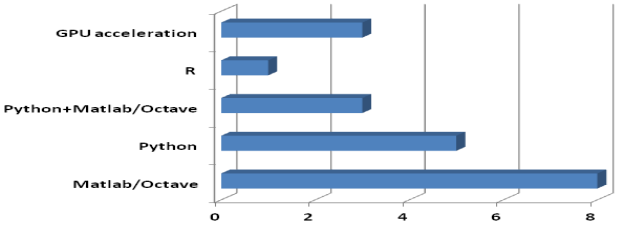


Figure 11: Programming language.

feature when no multiple modalities were used. In general, combining audio plus skeleton information was the predominant strategy among the participants, and the one considered by the first three ranked teams on the test set.

The considered temporal segmentation strategies are shown in Figure 8. One can see that two strategies were mainly applied: based on features and based on temporal windows or classifiers (such as Dynamic Time Warping). For the first case, audio information and joint positions were the most considered information to split the continuous sequence into gesture candidates. In the second case, Sliding-Windows technique was the preferred choice of the participants.

Regarding the fusion strategy (Figure 9), several authors did not apply any, since only one cue was used in their approach or different cues were independently used in different stages, such as one for temporal segmentation and the other one for describing segmented candidate gestures and final classification. Regarding the participants that fused different modalities (the majority of the teams), few of them combined feature vectors in an early fusion fashion before training a classifier. The preferred strategy was to train classifiers on different feature sets from different modalities and fuse the weighted outputs of classifiers in a late fusion fashion.

Regarding the classifiers (Figure 10), it is interesting to see the broad variety of strategies, covering most of the state-of-the-art Machine Learning strategies. Both discriminative and generative classifiers were considered, in all cases applying supervised learning. The preferred strategy was Hidden Markov Models. On the other hand dynamic programming, and in particular Dynamic Time Warping, which of often one of the preferred methods for gesture recognition, was not widely applied in our challenge (fourth choice). In particular, Random Forest and Neural Network variants were the second and third choice of the participants, respectively.

Finally, Figure 11 summarizes the programming languages: mainly Matlab/Octave and Python languages were used, and in few cases the codes were speeded up by means of GPU programming.

Table 3 shows the particular strategy from these statistics applied for each team of the top ranked positions of the challenge. Interestingly, the three top ranked participants agree in the modalities and segmentation strategy considered, although they differ in the final applied classifier.

5.3 Summary of the winner methods

Table 3 shows the summary of the strategies considered by each of the top ranked participants on the test set. Next, we briefly describe in more detail the approach designed by the three winners of the challenge.

The first ranked team *IVAMM* on the test set used a feature vector based on audio and skeletal information, and applied late fusion to obtain final recognition results. A simple time-domain end-point detection algorithm based on joint coordinates is applied to segment continuous data sequences into candidate gesture intervals. A Gaussian Hidden Markov Model is trained with 39-dimension MFCC features

Table 3: **Team methods and results. Early and late refer to early and late fusion of features/classifier outputs. HMM: Hidden Markov Models. KNN: Nearest Neighbor. RF: Random Forest. Tree: Decision Trees. ADA: Adaboost variants. SVM: Support Vector Machines. Fisher: Fisher Linear Discriminant Analysis. GMM: Gaussian Mixture Models. NN: Neural Networks. DGM: Deep Boltzmann Machines. LR: Logistic Regression. DP: Dynamic Programming. ELM: Extreme Learning Machines.**

TEAM	Test score	Rank position	Modalities	Segmentation	Fusion	Classifier
IVA MM	0.12756	1	Audio,Skeleton	Audio	None	HMM,DP,KNN
WWEIGHT	0.15387	2	Audio,Skeleton	Audio	Late	RF,KNN
ET	0.16813	3	Audio,Skeleton	Audio	Late	Tree,RF,ADA
MmM	0.17215	4	Audio,RGB+Depth	Audio	Late	SVM,Fisher,GMM,KNN
PPTK	0.17325	5	Skeleton,RGB,Depth	Sliding windows	Late	GMM,HMM
LRS	0.17727	6	Audio,Skeleton,Depth	Sliding windows	Early	NN
MMDL	0.24452	7	Audio,Skeleton	Sliding windows	Late	DGM+LR
TELEPOINTS	0.25841	8	Audio,Skeleton,RGB	Audio,Skeleton	Late	HMM,SVM
CSI MM	0.28911	9	Audio,Skeleton	Audio	Early	HMM
SUMO	0.31652	10	Skeleton	Sliding windows	None	RF
GURU	0.37281	11	Audio,Skeleton,Depth	DP	Late	DP,RF,HMM
AURINKO	0.63304	12	Skeleton,RGB	Skeleton	Late	ELM
STEVENWUDI	0.74415	13	Audio,Skeleton	Sliding windows	Early	DNN,HMM
JACKSPARROW	0.79313	14	Skeleton	Sliding windows	None	NN
JOEWAN	0.83772	15	Skeleton	Sliding windows	None	KNN
MILAN KOVAC	0.87463	16	Skeleton	Sliding windows	None	NN
IAMKHADER	0.92069	17	Depth	Sliding windows	None	RF

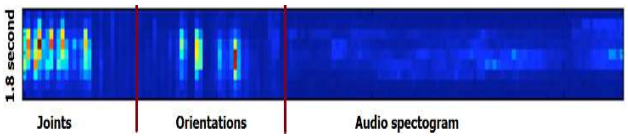


Figure 12: **ExtraTreesClassifier Feature Importance.**

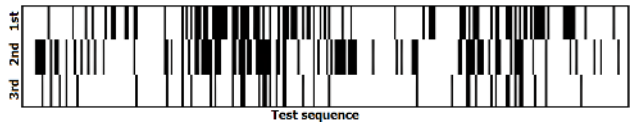


Figure 13: **Recognition of test sequence by the three challenge winners. Black bin means that the complete list of ordered gestures has been successfully recognized.**

and generates confidence scores for each gesture category. A Dynamic Time Warping based skeletal feature classifier is applied to provide complementary information. The confidence scores generated by the two classifiers are firstly normalized and then combined to produce a weighted sum. A single threshold approach is employed to classify meaningful gesture intervals from meaningless intervals caused by false detection of speech intervals.

The second ranked team *WWEIGHT* combined audio and skeletal information, using both joint spatial distribution and joint orientation. The method first searches for regions of time with high audio-energy to define 1.8-second-long windows of time that potentially contained a gesture. This had the effect that the development, validation, and test data were treated uniformly. Feature vectors are then defined using a log-spaced audio spectrogram and the joint positions and orientations above the hips. At each time sample the method subtracts the average 3D position of the left and right shoulders from each 3D joint position. Data is down-sampled onto a 5 Hz grid considering 1.8 seconds. There were 1593 features total (9 time samples $\times$ 177 features per time sample). Since some of the detected windows can contain distracter gestures, an extra 21st label is introduced, defining the 'not in the dictionary' gesture category. Python's scikit-learn was used to train two models: an ensemble of randomized decision trees (ExtraTreesClassifier, 100 trees, 40% of features) and a K-Nearest Neighbor model (7 neighbors, L1 distance). The posteriors from these models are averaged with equal weight. Finally, a heuristic is used (12 gestures maximum, no repeats) to convert posteriors to a prediction for the sequence of gestures.

Figure 12 shows the mean feature importance for the windows size of 1.8 seconds for the three sets of features: joint coordinates, joint orientations, and audio spectogram. One can note that features from the three sets are selected as discriminative by the classifier, although skeletal features becomes more useful for the ExtraTreesClassifier. Additionally, the most discriminative features are those in the middle of the windows size, since begin-end features are shared among different gestures (transitions) and thus are less discriminative for the classifier.

The third ranked team *ET* combined the output decisions of two designed approaches. In the first approach, they look for gesture intervals (unsupervised) using the audio files and extracts features from this intervals (MFCC). Using these features, authors train a random forest and gradient boosting classifier. The second approch uses simple statistics (median, var, min, max) on the first 40 frames for each gesture to build the training samples. The prediction phase uses a sliding window. The authors create a weighted average of the output of these two models [4]. The features considered were skeleton information and audio signal.

Finally, we extracted some statistics from the results of the three challenge winners in order to analyze common points and difficult aspects of the challenge. Figure 13 shows the recognition of the 276 test sequences by the winners. Black bin means that the complete list of ordered gestures was successfully recognized for those sequences. Once can see that there exists some kind of correlation among methods. Taking into account that consecutive sequences belong to the same user performing gestures, it means that some some gestures are easier to recognize than others. Since different users appears in the training and test sequences, it is sometimes difficult for the models to generalize to the style of new users, based on the gesture instances used for training.

We also investigated the difficulty of the problem by gesture category, within each of the 20 Italian gesture categories. Figure 14 shows for each winner method the devi-

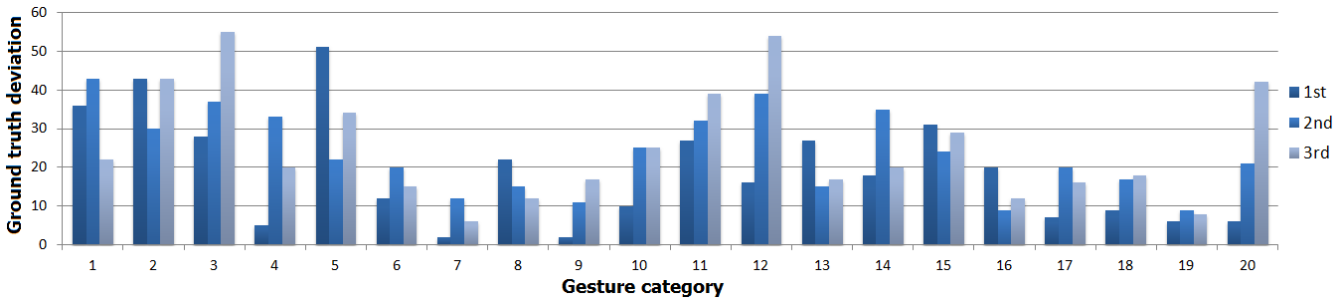


Figure 14: Deviation of the number of gesture samples for each category by the three winners in relation to the GT data.

ation between the number of gesture instances recognized and the total number of gestures, for each category. This was computed for each sequence independently, and adding the deviation for all the sequences. In that case, a zero value means that the participant method recognized the same number of gesture instances for a category that was recorded in the ground truth data. Although we cannot guarantee with this measure that the order of recognized gesture matches with the ground truth, it gives us an idea of how difficult the gesture sequences were to segment into individual gestures. Additionally, the sum of total deviation for all the gestures for all the teams was 378, 469, and 504, which correlates with the final rank of the winners. The figure suggests a correlation between the performance of the three winners. In particular, categories 6, 7, 8, 9, 16, 17, 18, and 19 were the ones that achieved most accuracy for all the participants, meanwhile 1, 2, 3, 5, and 12 were the ones that introduced the highest recognition error. Note that the public data set provides accurate label annotations of end-begin of gestures, and thus, a more detailed recognition analysis could be performed applying a different recognition measurement to Leveinstein, such as Jaccard overlapping or sensitivity score estimation, which will also allow for confusion matrix estimation based on both inter and intra user and gesture category variability. This is left to future work.

6. CONCLUSIONS

This paper describes the ChaLearn Multi-modal Gesture Recognition Challenge. For this purpose, we designed a large data set, including several people that perform gestures from a vocabulary 20 Italian sign gesture categories. Data also include distracter gestures to make the recognition task challenging. The modalities provided included audio, RGB, depth maps, user masks, and skeletal model. The datasets has been manually annotated to provide ground truth of temporal segmentation of the signal into individual gestures. The dataset has been made publicly available.

Different classifiers for gesture recognition were used by the participants. The preferred one was Hidden Markov Models (used by the first ranked team of the challenge), followed by Random Forest (used by the second and third winners). Although several state of the art learning and testing gesture techniques were applied at the last stage of the methods of the participants, still the feature vector descriptions are mainly based on MFCC audio features and skeleton joint information. This supports the use of complementary source of information, but it is expected that the use of more sophisticated features in a near feature will be useful to reduce the current error rate achieved in the data set. For instance, we think that structural hand information around hand joint could be useful to discriminate among gesture categories that may share similar trajectories of hand/arms.

Although the current error rate on the data set is about 12% using de Levenshtein edit distance among order of predicted gestures, it still offers range for improvement and test for other metrics given the provided ground truth, such are Jaccard overlapping index or sensitivity.

7. ACKNOWLEDGMENTS

The ChaLearn Multi-modal Gesture Recognition Challenge was organized thanks to the support of ChaLearn[9], the University of Barcelona Mathematics Faculty, the Universitat Autònoma de Barcelona, the Computer Vision Center, the Universitat Oberta de Catalunya, and the Human Pose Recovery and Behavior Analysis Group[10]. We thank the Kaggle submission website for wonderful support, together with the committee members and participants of the ICMI 2013 Multi-modal Gesture Recognition workshop for their support, reviews and contributions. We lastly would like to thank the students Pau Rodríguez, Meysam Madadi and Aaron Negrín for their help in the annotation of gestures. This work has also been partially supported by Pascal2 network of excellence, and the Spanish projects TIN2009-14501-C02-02, TIN2012-39051, and TIN2012-38187-C03-02.

8. REFERENCES

[1] M. Everingham, L. V. Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.

[2] I. Guyon, V. Athitsos, P. Jangyodsuk, H. Escalante, and B. Hamner. Results and analysis of the chalearn gesture challenge 2012. *ICPR*, 2012.

[3] A. Hernandez-Vela, N. Zlateva, A. Marinov, M. Reyes, P. Radeva, D. Dimov, and S. Escalera. Human limb segmentation in depth maps based on spatio-temporal graph-cuts optimization. *JAISE*, 4(6):535–546, 2012.

[4] Pedregosa. Scikit-learn: Machine learning in python. In *Journal of Machine Learning Research*, pages 2825–2830, 2011.

[5] J. Shotton. Real-time human pose recognition in parts from single depth images. *CVPR*, pages 1297–1304, 2011.

[9] ChaLearn: `http://chalearn.org`

[10] HuPBA research group: http://www.maia.ub.es/∼sergio/