Team name	JY
Team leader name	Ju Yong Chang
Team leader address, phone number and email	218 Gajeong-ro, Yuseong-gu, Daejeon 305-700, Republic of Korea (Phone) +82-10-5389-0755 (E-mail) juyong.chang@gmail.com
Rest of team members	-
Team website URL (if any)	-

Title of the contribution	Nonparametric gesture labeling from multi-modal data
General method description	We formulate the problem by the gesture labeling framework, which means that a gesture category label is assigned to each video frame. For that purpose, a generative probabilistic model is developed and the labeling problem is solved by the MAP approach. Our method relies on three features: skeleton joint position, skeleton joint distance, and color HOG vectors of both hands. The probability for each feature is modeled in a nonparametric manner. Temporal coherence is additionally assumed and it is enforced by a simple MRF model. To find the optimal solution, the dynamic programming method is used.
References	-

Describe data preprocessing techniques applied (if any)	Simple normalization (only for skeleton) and feature scaling (for both skeleton and color HOG)
Describe features used or data representation model (if any)	Three features including skeleton joint position, skeleton joint distance, and color HOG for both hands.
Data modalities used, i.e. depth, rgb, skeleton… (if any)	Skeleton and RGB.
Fusion strategy applied (if any)	Features are assumed to be independent each other. Their likelihood functions are probabilistically combined to produce the overall multi-modal likelihood.
Dimensionality reduction technique applied (if any)	Only PCA is applied.

Temporal clustering approach (if any)	-
Temporal segmentation approach (if any)	Our gesture labeling framework automatically segments the input video.
Gesture representation approach (if any)	-
Classifier used (if any)	The likelihood function for each test feature is nonparametrically defined from the training feature vectors. Specifically, its computation is performed by the nearest neighbor (NN) search.
Large scale strategy (if any)	-

Transfer learning strategy (if any)	-
Temporal coherence and/or tracking approach considered (if any)	Temporal coherence is simply enforce by assuming the MRF model with the temporal compatibility term.
Other technique/strategy used not included in previous items (if any)	Before classification, the input gesture is classified into two cases: left-hand oriented or right-hand oriented. The probability model is individually constructed for each case. This approach increases the discriminability of our method.
Method complexity analysis	In the training stage, the kdtree is constructed for efficient nearest neighbor searching. In the testing stage, 1D MRF labeling problem should be solved and this can be very efficiently done by using simple dynamic programming.

Qualitative advantages of the proposed solution	**Our approach is conceptually simple and easy to implement. Simple conventional features such as skeleton joint position and HOG for color image are adopted and they are intuitively used by the nonparametric nearest neighbor method. Our method has few parameters and heavy learning process is not necessary. Gesture labeling is carried out by 1D MRF inference and this can be done fast by using dynamic programming.**
Results of the comparison to other approaches (if any)	-
Novelty degree of the solution and if is has been previously published	Gesture recognition has been sometimes formulated in the labeling problem and usually handled by parametric approach such as HMM and CRF. As far as we know, nonparametric method such as ours has not been used for the gesture labeling problem. Probabilistic fusion of multi-modal features and consideration of left/right-handedness of gestures are other novel aspects of our approach. This has not been published.

Language and implementation details (including platform, memory, parallelization requirements)	**We implement our method using Matlab R2013b. Our system is based on four 2.4GHz Intel Zeon CPUs, 1TB memory, and Windows 8 OS. The parallel processing toolbox of Matlab is utilized to speed up.**
Human effort required for implementation, training and validation?	We have implemented our algorithm and tested it for about three months.
Training/testing expended time?	Training time is 1191.7 (sec) and testing time is 474.1 (sec). For both cases, the file loading time was not considered.
General comments and impressions of the challenge	Thank you so much for hosting this wonderful contest. This must be helpful in advancing the action/gesture recognition technology in computer vision.