Poznan University of Technology
Faculty of Computing
Institute of Computing Science

Bachelor's thesis

BIBLIOTEKA DO ROZPOZNAWANIA GESTÓW DLA KONTROLERA LEAP MOTION

Katarzyna Zjawin, xxx
Michał Nowicki, 95883
Olgierd Pilarczyk, xxx
Jakub Wąsikowski, xxx

Supervisor
dr inż. Wojciech Jaśkowski

Poznań, 2013

Tutaj przychodzi karta pracy dyplomowej;
oryginał wstawiamy do wersji dla archiwum PP, w pozostałych kopiach wstawiamy ksero.

Contents

Chapter 1

Lorem Ipsum

1.1 Lorem

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc fringilla facilisis massa. Donec ultrices dignissim quam. Phasellus vel erat quis diam tempor porttitor. Maecenas purus ligula, varius id, suscipit in, congue quis, erat. Vivamus erat purus, laoreet rutrum, aliquet eleifend, porta a, erat. Etiam id neque id nunc porttitor nonummy. Mauris pulvinar mi quis elit. Sed placerat. Cras dictum neque vel odio. Sed a augue.

Aenean eu ligula. Nulla sit amet metus et risus vehicula elementum. Vivamus a eros. Etiam lorem odio, iaculis ac, vulputate vitae, cursus a, tortor. Sed sagittis volutpat libero. Integer ornare, dui nec vulputate auctor, ante mi lacinia ipsum, nec mollis ipsum libero at arcu. Suspendisse ut ligula. Nunc metus. Proin mollis pellentesque ante. Donec dictum. Sed purus ante, placerat eget, vestibulum in, lobortis ut, pede.

Sed blandit. Fusce blandit turpis id eros auctor dictum. Ut porttitor mollis magna. Nam iaculis sem vel velit. Sed sollicitudin libero non quam. In hac habitasse platea dictumst. Praesent tristique, justo vitae mollis fringilla, nisl ligula feugiat lacus, ac vestibulum mi turpis in nisi. Aenean tristique fermentum erat. Integer consectetuer erat quis ante. Morbi gravida hendrerit nulla. Etiam magna.

1.1.1 Ipsum

Curabitur adipiscing. Nam justo quam, dapibus eu, dignissim eu, hendrerit consequat, odio. Praesent mi ligula, vulputate et, ullamcorper ut, pellentesque ut, massa. Suspendisse potenti. Morbi interdum nisi vel justo. Aenean in tortor id mi dapibus porttitor. Nunc convallis. Nullam vitae felis. Maecenas fringilla, lorem quis aliquam laoreet, elit justo tincidunt nisi, ac sollicitudin sapien nulla non nisi. Etiam et mauris. Nunc scelerisque rhoncus leo.

Morbi posuere nisi id libero. Etiam tincidunt facilisis sem. Suspendisse vulputate, lectus quis volutpat consectetuer, urna massa vestibulum tellus, nec vehicula dolor orci id odio. Nam faucibus sem eu nisl ullamcorper lobortis. Sed quis ipsum. Morbi volutpat pellentesque erat. Quisque ipsum velit, aliquet eu, imperdiet ut, commodo sed, est. Proin euismod porta enim. Mauris sit amet purus. Nunc ac turpis. Nunc tellus magna, sagittis in, pellentesque vitae, semper et, sem. Vivamus et odio. Suspendisse tincidunt mattis leo. Morbi pharetra. Quisque fermentum, lectus eget feugiat iaculis, enim neque accumsan felis, id hendrerit enim erat et ante. Nunc blandit commodo est. Fusce venenatis urna quis lorem. Aenean nisi. Vivamus ornare velit in magna imperdiet dapibus.

1.1.2 Aliquam Venenatis

Cras non enim eget mi aliquam venenatis. Duis posuere imperdiet quam. Maecenas egestas. In tempus tortor quis pede. Aenean eget mauris sit amet ipsum scelerisque porttitor. Vestibulum nisl. Vestibulum id velit eget tellus ultrices laoreet. Sed pellentesque, odio eget adipiscing commodo, turpis lorem placerat lacus, et interdum quam mi quis odio. Nam et magna. Pellentesque vitae nisl. In erat est, iaculis nec, sodales imperdiet, dictum at, eros. Nullam eget leo. Praesent nunc. Morbi id erat. Pellentesque vel felis vitae est consectetuer interdum. Fusce venenatis. Nam rhoncus tortor et sem. Sed ut sem aliquet urna congue suscipit. Nulla eget risus in purus molestie feugiat. Phasellus non libero vel odio vehicula placerat.

Aenean velit. Vestibulum dui sem, scelerisque non, nonummy quis, accumsan eget, ipsum. Quisque nec est et ipsum bibendum venenatis. Donec eget pede. Sed vitae ante et metus pulvinar fermentum. Nullam scelerisque laoreet dui. Aliquam consectetuer. Quisque molestie eros a felis. Maecenas nec nisi. Praesent mollis nisl sit amet pede. Phasellus iaculis elementum ante.

Fusce lacinia magna ut lacus. In aliquet diam eget augue. Curabitur blandit nisi sit amet nulla. Donec nunc. Nullam pellentesque eleifend nisi. Proin ac elit. Fusce a neque. Nunc felis. Aenean sodales. Etiam a risus non elit commodo rhoncus. Ut mi. Cras gravida neque nec diam. Proin consequat vestibulum nunc. Aliquam mattis justo et purus. Sed feugiat.

Fusce fermentum. Nulla orci lectus, pellentesque sit amet, egestas sed, ornare a, erat. Vivamus ante odio, auctor ac, elementum eget, feugiat sit amet, enim. Aliquam venenatis semper erat. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aliquam non tellus. Mauris ornare massa iaculis magna. Morbi eu enim. Mauris ultrices. Vivamus dapibus aliquet nulla. Quisque consectetuer.

1.2 Pellentesque Habitant

Pellentesque est nisl, scelerisque in, dictum vel, dapibus nec, lacus. Donec eu quam sed ligula luctus volutpat. Donec ornare ornare purus. Mauris ligula eros, pulvinar nec, viverra in, tristique quis, nunc. Pellentesque at diam. Integer tempus risus vitae lorem. Vestibulum sed lorem id elit convallis rutrum. Quisque lobortis massa. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Maecenas purus. Etiam hendrerit malesuada libero. Sed non mauris.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Sed ligula. Quisque nec augue. Suspendisse id lacus. Mauris a turpis. Curabitur interdum nulla tempus justo. Sed nisl mauris, volutpat eleifend, suscipit in, ornare vitae, odio. Ut tempor, pede sed volutpat ornare, mauris elit laoreet turpis, non posuere lorem nunc eu risus. Curabitur vulputate, risus et auctor dapibus, ipsum nisl gravida nulla, vitae venenatis nibh libero id tellus. In accumsan aliquet augue.

In consectetuer, magna in ornare eleifend, neque leo tempor nibh, quis viverra neque sapien malesuada orci. Quisque eget magna. Integer tempus. Sed malesuada, orci a cursus posuere, justo magna laoreet mauris, at pulvinar sapien velit ut nisi. Sed sed augue quis libero hendrerit feugiat. Nam porta. In et massa. Curabitur blandit purus vitae risus luctus tristique. Sed ipsum. Morbi tincidunt, dolor ut aliquam luctus, magna libero sagittis risus, eget dignissim risus lacus quis magna. Donec tincidunt varius elit. Ut in velit. Curabitur dui velit, vestibulum in, tristique nonummy, venenatis et, eros. Maecenas mi leo, placerat vel, congue vitae, semper ut, nunc. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Mauris orci elit, hendrerit vestibulum, vulputate ut, venenatis et, mauris. Curabitur vitae tortor vitae elit vulputate congue. Mauris felis dui, lobortis vel, consectetuer id, cursus at, sapien.

Integer ut turpis. Duis odio ligula, lobortis ut, egestas a, porttitor id, ligula. Nunc vulputate, ipsum quis varius aliquet, est eros sagittis urna, interdum imperdiet lorem ipsum tincidunt nibh. Vivamus dictum porttitor diam. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Praesent at nulla. Praesent tincidunt ipsum a velit. Fusce ut ipsum sed purus scelerisque nonummy. Cras imperdiet velit eu nibh. Ut ultrices mauris et risus.

Chapter 2

Static gestures recognition

As was already mentioned, the detected gestures can be divided into two groups: static gestures and dynamic gestures. The static gestures can be understood as a chosen position and orientation of the fingers and hand in a single moment, while dynamic gestures are defined as a movement of the hand and fingers in time. The problem of recognition of those gestures is a subject of following chapter. Firstly, the proposed approach is presented, followed by the introduction to the evaluation scheme. In last section, the performed experiments are performed to examine the effectiveness in task of static gesture recognition.

2.1 Proposed methods

The static gesture recognition problem can be stated as a problem invariant to time. That means that for each detected hand, the position and orientation can be treated as a new data uncorrelated to previously classified data. While this assumption means that one can easily generate multiple samples from the sensors in short time, it also gives an opportunity to look at the static gesture recognition problem as a problem of classification.

While for most 2D gesture recognition problems simple classification algorithms seems to work well enough, the 3D data is more complication to model by the set of features and finally successfully label. While dealing with 3D data, the position and orientation of hand can be easily affected by the height of the hand above the sensor or small change in the orientation of the hand with respect to the sensor's cooridnate system. It is intuitively understood, that the system should recognize those gestures as the one as they are similar. To meet those requirement, one need to define what is meant by the „small" change in orientation resulting in treating the static gestures as the same.

To meet those requirements, the Support Vector Machines [CC]

In proposed application, it was assumed that the gesture is treated as the same independently with respect to the translation, rotation and scale of the hand. This assumption means that the static gesture rotated by unknown angles, translated in sensor coordinate system and for different hand sizes can still be treated as the same gesture. Invariance to the rotation, translation and scale poses a great challenge to the recognition, but allows the future users of API to fully utilize the feasibility of the library. It is worth mentioning that, it does not reduce the possible applications of the library, as an assignment of static gesture to defined class allows to find the transformation between the model of the class and observed gesture.

2.2 Evaluation methodology

The problem of classification assumes that each sample consists of set of features, which describe this sample and can be used to distinguish it from the other samples. Additionally, each sample has a known or unknown label, which defined the membership of sample to the class. The samples with the known labels can be used to train the classification system to compute the membership to the classes for the samples. The computation is performed on previously mentioned sets of features.

In application of gesture recognition the classification be divided into two flows: the training part and the recognition part. In training part, the library will be provided with the samples of static gestures with known correspondences to the static gesture classes. From those samples, the sets of features are computed, which are used to train the classifier. The recognition part assumes to have trained classifier. The recognition part is provided with samples static gestures without labels. For each sample the sets of features are computed and then given as input to the trained classifier. The classifier returns the information of the gesture's class membership (label) of each sample.

In case of library, it is assumed that the learning process can be done offline, while strict online requirements has to be met in recognition part. To meet those requirements the Support Vector Machine is introduced[]. The SVM classification is commonly used technique in multiple areas of research as biology, robotics or IT for solving data classification problems []. Additional advantage of the SVM is possibility to use C++ library libSVM[], which provides an easy interface to utilize this classification methods in different problems.

While presented approach can be treated as state-of-the-art approach it still cannot be used without defining proper feature sets for gesture recognition. The naive solution would be to use the raw data from Leap motion sensor as the feature set. This solution was tested, but provided poor results as the proposed features were dependent on the position, orientation and scale of hand. Even small movement in any direction meant problems with stable recognition. The theoretical literature suggests to compute a set of features invariant to wanted transformations, which can allow to fully distinguish between different classes. Unfortunately, there are not available propositions to feature sets when it comes to the gesture recognition using the data even similar to the data provided by the Leap Motion sensor. Reminding, the Leap Motion for each frame allows to capture:

- position in X, Y, Z of each recognized finger
- unit vector of each recognized finger
- the width and height of each finger
- the normal vector of the hand

2.3

To propose and test the quality of the features, five static gestures were recorded:

- the peace sign,
- a fist,
- full hand with space between each finger,
- the sign X made with the forefingers of both hands,

- the sing "Time" used e.g. by coaches in Basketball games.

The sample data of each gestures were recorded using the continuous mode of recording, while moving the hands in different directions and changing the orientation of the hands. For each of the proposed gestures, each author recorded approximately 1000 samples.

Having samples with known labels, the utilization of the cross-validation scheme could be used. This method is used to find the optimal parameters of the classification system, while estimating the performance on the data not used in the training part. In standard version of the method, the gathered data is divided into two sets: one containing 20% of the data, the other 80% of the data. The 20% is used to train the classification system, while the rest of the gathered data is used to estimate the performance. The performance is estimated by calculating the number of cases when the classification system returned a label which matched already known label. The percent of correctly recognized labels to the total size of the testing set is known as recognition rate.

In proposed solution, the cross-validation was used to test different set of features on the prerecorded dataset. The first proposed vector of features consisted of:

- number of fingers in frame,
- the absolute angles between consecutive fingers,
- the euclidean distance between consecutive finger's tips.

After approach allowed to achieve XXX% of recognition rate and was unsatisfying from the perspective of feature application. Analysis of the finger numbering revealed that the fingers are numbered accordingly to the position in Z axis of the tip of the finger. This means that when features are approximately on the same position in Z axis, the numbering can change rapidly and proposed features compare different fingers. To achieve the features that would be invariant to the numbering of the fingers, the feature set was changed. Instead of containing the absolute angles and distances between consecutive fingers, it was proposed to contain the five greatest values of angles and five greatest values of distances between all combinations of finger pairings. This approach was tested on the same training set and allowed to increase the recognition rate to the 83.8812%. While the number of fingers in frame and five greatest angles between all possible angles between fingers are invariant to the translation, rotation and scale, the distances between tip position were dependent on the scale of hand. To achieve distances invariant to the scale of hand, it was proposed to exchange the five greatest distances for five greatest ratios of the distances between tips of all fingers to the maximal found distance between tips of all fingers. As this approach was expected to increase the recognition rate, it fell down to 77.0913%. More experiments with enlarging the set of features to having ten greatest angles and ten greatest distances showed slight fall in recognition rate to 83.7687%. The all results have been presented in table [].

To increase the recognition rate an attempt to gather more training data was performed, which end with recognition rate equal to XX%. While using more data, it is worth noticing the growth of training time. In case of 5000 samples the typical training process took approximately 6 hours. This computing time can be unacceptable by the users of the library, so the test with another SVM library libLinear[] was performed. The libLinear's implementation of SVM utilizes the linear kernels, which are useful for large data training sets with multiple number of features. This library reduced the training time to about 20 seconds, but the best obtained recognition rate was 64.524% for libLinear compared to the 83.8812% for the libSVM.

Additional tests consisted of recording additional five gestures:

- American Sign Language: "I love you" sign,

- sing "gun" created by putting thumb and forefinger up, while holding the rest fingers in a fist,
- all fingers in a fist with exception of thumb, which is up,
- sign simulating rotating a knob by two fingers,
- sign simulating rotating a knob by five fingers.

All of those signs are presented at figure [].

The firstly tested set of static gestures contained gestures, which were did not take into account the way how the Leap Motion works and for gestures like fist or 'X' the recorded data contained almost no information how to classify those gestures. That's why the experiments were repeated on the five gestures, which could be easily distinguishable by data provided by Leap Motion. To this experiment the gestures peace, hand, "I love you", fist with thump up and rotating knob by 5 fingers were chosen. For this classification problem with feature set defined by number of fingers, 5 greatest absolute angles and 5 greatest absolute distances between fingers, the recognition rate of 90.9532% was achieved.

The last test performed for static gestures recognition consisted of classifying the set of 10 possible static gestures. For this test the recognition rate of 73.2118% was achieved.

Chapter 3

Detection of dynamic gestures

3.1 Proposed methods

The dynamic gesture recognition problem is a problem, where the input data consist of several consecutive positions and orientations of hand and fingers. Moreover, the important factor for recognition is the time dependencies between data frames. The slower and faster gestures should be recognized as the same dynamic gesture.

The proposed solution utilizes parts of the solution used for recognition of the static gestures. Each frame of the captured data is described by the same features as in the static recognition part. The set of features for each frame is then processed by the Hidden Markov Model scheme.

Hidden Markov Model

Hidde Markov Models can be represented by the set: $(, , , , ,)$

where O is set of observations, X is a set of states, Y is a matrix defining the probabilities of transitions between states, Z is a matrix defining the probabilities of observations from each state. The best way to represent HMM is to use the structure of the graph with two types of vertices. This way, each state is represented by one type of vertices while observations can be shown as second type of vertices. The edges between states contain and are an equivalent to the eY matrix. There are no edges between vertices representing observations. The edges between states and observations also contain probabilities from the Z matrix. The problem of recognizing the dynamic gesture can be understood as a problem of finding the path of states in HMM which maximizes the combined probability given set of observations. It can be written as: xx There are three main algorithms connected with HMM:

- Forward-Backward algorithm,
- Viterbi algorithm,
- Baum-Welch algorithm.

The Forward-Backward algorithm is used to find the state that maximizes the likelihood given the set of observations. It achieves that by performing two operations. First of them **** .

The Viterbi algorithm is used the path of connected states that best explain the set of observations. It is algorithm based on the Forward-Backward, which also can understood as the usage of the Expected Maximization approach.

The Baum-Welch algorithm is the algorithm used to train the HMM. For each training example, the algorithms changes the probability and transition matrices to maximize the likelihood of observation.

In dynamic gesture recognition problem, each gesture can be modelled by the sequence of n states in which $kths$ state is connected by the edges to the k and $(k+1)$ state and to the all observations. The proposed architecture can be seen at fig. *. Having the problem of distinguishing m gestures translates to the m sequential graphs.

3.2 Evaluation methodology

The proposed solution utilizes Hidden Markov Model[]

3.3 Experiments

Bibliography

[CC] Vladimir Vapnik Corinna Cortes. *Support-vector networks.*

Poznan University of Technology
Faculty of Computing
Institute of Computing Science

Typeset using LaTeX in Computer Modern.

BibTeX:

```
@mastersthesis{ key,
    author = "Katarzyna Zjawin \and Michał Nowicki \and Olgierd Pilarczyk \and Jakub Wąsikowski",
    title = "{Biblioteka do rozpoznawania gestów dla kontrolera Leap Motion}",
    school = "Poznan University of Technology",
    address = "Pozna{\'n}, Poland",
    year = "2013",
}
```