Федеральное государственное автономное

образовательное учреждение высшего образования

«Национальный исследовательский

Томский государственный университет»

IX Международная научно-практическая конференция

«Информационно-измерительная техника и технологии»

в рамках Международного форума

«Интеллектуальные системы 4-й промышленной революции»

ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНАЯ ТЕХНИКА И ТЕХНОЛОГИИ

СБОРНИК ТЕЗИСОВ КОНФЕРЕНЦИИ

Спонсор

Организаторы

21-24 ноября 2018 года

г. Томск

as where x1 - x4 are fed data from a cloud of points. The overall picture (Fig. 2) looks like matrices.

5. **Conclusion**:

This article used a point cloud to create an octree. The results show that when using uniformly structured models and parallel processing, the results and implementation time of the algorithm are faster at each stage of performing image processing tasks in three dimensions. This is such an environment that allows you to very quickly build a map based on the octree. And SLAM octree has specific features. We directly choose and play specifically to solve this problem. Creating a more computationally competitive computing environment is a good result and a good condition. That is the context to continue to study.

References:

[1]. D. V. Shashev., S. V. Shidlovsky. Morphological processing of binary images using tunable computing environments, 2015, pp.19-26.

[2]. Gonzalez R., Woods R. Digital Image Processing: Trans. from English M .: Technosphere, 2005, pp.1072.

[3]. Ivanov V. A., Kirichuk V. S. Features of the work of detection algorithms for a fallen person by the sequence of images of the scene // Avtometriya, No. 2, 2011, pp.15–25.

[4]. Shapiro L., Stockman J. Computer vision: Trans. from English M .: BINOM. Laboratory of Knowledge, 2006, pp.752.

[5]. Shidlovsky S.V. Automatic control. Reconfigurable systems: Proc. allowance. Tomsk: Publishing house Tom. University, 2010, pp. 168.

[6]. Shidlovsky S.V. Automatic control. Tunable structures. Tomsk: Tomsk State University, 2006, pp.288 .

[7]. Kalyaev I. A., Levin I. I., Semernikov E. A., Shmoilov V. I. Reconfigurable multi-pipeline computing structures. Rostov n / D: SSC RAS Publishing House, 2008, pp.393.

[8]. Khoroshevsky V. G., Kurnosov M. G., Mamaylenko S. N. Spatially-distributed multi-cluster computing system: architecture and software // Vestn. TSU. Management, computing and computer science, № 1 (14), 2011, pp.79–84.

[9]. https://en.wikipedia.org/wiki/Boolean_algebra

[10] https://www.kudan.eu/kudan-news/an-introduction-to-slam/

Research method detection human face in video streams

Nguyen The Cuong[1,2]

Supervisor: Vladimir Ivanovic Syryamkin[1]

1. Tomsk State University, 2. Vietnam Maritime University

cuongntit@vimaru.edu.vn

Human face recognition is a field of study in the field of computer vision. Face recognition methods are now divided into different directions according to different criteria. Research article on face detection techniques in video streams, using the OpenCV library.

Keyword: face dectection, face recognize, feature based, feature extractrion.

1. Introduction

The face recognition has a rich data source and requires less controlled interaction and can be found in real life as well as data on the net. It's method is also divided into several directions: Identification with input data is a 2D still image (is most popular), but the future will probably be 3D FR (because, if layout of many 2D cameras will give the data in kind of 3D and deliver better, more reliable results) can also be divided into two directions: to do with image data and to do with video data. In reality, people divide the face recognition methods into three categories: global approaches, such as Eigenfaces-PCA [1], Fisherfaces-LDA [2]), based on local feature based (LBP, Gabor wavelets [3]) and hybrids (a combination of two global and local features). Local-based methodologies have been proven to be superior in working under uncontrolled data conditions. It can be said that the development history of human face recognition is the development of selective extraction methods used in image feature extractrion systems. Specific applications of face recognition are based on two identification models: identification (1-N problem), and verification (problem 1-1). In the identification problem, we need to determine the identity of the test image, but in the verification problem, we need to determine whether the two images belong to the same person.

2. Phases in a face recognition system

To build a face recognition system, it is not easy. The first step is "face detection" - it means detecting the image in the input (image database, video ...) and cut the face to perform the identification. The second step is "image preprocessing" including image alignment and normalization (here we referring to straight-line images). The third step is "character selection". In this step, a method of extracting certain characteristics (LBP, Gabor wavelets, [4] ...) will be used with the facial image to extract image specific information. As a result, each image is represented as a feature vector; the next step is the identification or classification step, Identification or label of the image - that is the image of the one. At the classification step, usually the method k-nearest neighbor (k-NN) [5] will be used, in practice the use of Support Vector Machine [6]) does not bring worse results.

Data for a face recognition system is divided into 3 volumes: training set, reference set, and set for identification. In many systems, the training file is identical to the reference file. The training set consists of images used for training, usually used to generate a projection subspace as a matrix and the commonly used method is Principal Component Analysis (PCA) [7], Whitened PCA (WPCA), Linear Discriminant Analysis (LDA) [8],

Kernel PCA (KPCA) [9]. The reference set includes known images that are projected onto the subtree at the training step. Training steps aim for two purposes: to reduce the dimension reduction of feature vectors because these vectors usually have large lengths (several thousand to several hundred thousand). very long. The second is to increase the discriminative between the different images class, in addition can reduce the distinction between images belong to a class (according to the method, the purse LDA, also known as Fisher Linear Discriminant Analysis, is a method of working with training sets where each object has multiple faces at different conditions. After making a reference reference to a subspace, the system saves the result as a matrix with each column of the matrix a vector corresponding to the image (known identifier) to perform the identification (or subclass). Identification is done with a set of probe images, after preprocessing, each image will be applied to the extracted feature (such as training and reference images) and projected into subspace. Next, the classifier will be based on the k-NN method. The identifier of an image that needs to be determined will be assigned the identifier of the image closest to it. Note that each image is a vector so it is possible to use the concept of distance function between two vectors to measure the difference between the images.

3. How to preprocess the image to face recognition

Once the image area is detected, you can use it to identify faces with the next image. However, if you were trying to simply perform face recognition directly on a normal photo, you would probably get less than 10% accuracy! It is extremely important to apply various image processing techniques before processing to standardize the images that you provide for a facial recognition system. Most facial recognition algorithms are extremely sensitive to light conditions, so if it is trained to recognize a person when they are in a dark room, then perhaps it wont recognize them in the a light room, etc. This problem is called "dependency lumination", and also many other problems, such as face should also be in a very suitable position in the image (such as eye is in the same pixel coordinates), suitable size, rotation angle, hair and makeup, emotions (smiling, angry, etc.), position of the lamp (left or top, etc.). This is why it is very important to use an image filter for processing before applying face recognition. You should also do things like remove the pixels around the face that are not being used, such as with an elliptical mask to show only the inner face area, not the hair and wallpaper, since they change a lot compared to no face. For simplicity, my face recognition system will show you that Eigenfaces uses grayscale images. So I will show you how to easily convert gray images, and then easily apply Histogram Equalization is a very simple method of automatic Standardize the brightness and contrast of your face mirror image. For better results, you can use face recognition colors (ideally with color chart accessories in HSV or a color space instead of RGB), or apply a variety of processing stages such as increased edge enhancement, border detection, motion detection, ... In addition, this code is resizing the image to a standard size, but this can change the aspect ratio of the face.

The pre-treatment steps are as follows:

s1. If a color image is converted to a gray image;

s2. Resizing an image is a suitable size, even if the scale changes.

s3. Create fixed size images;

4. Use the library in OpenCV for face detection

The OpenCV library makes it quite easy to detect a face in front of an image using the Haar Cascade Face Detector [10] (also known as the Viola-Jones method). The function "cvHaarDetectObjects" in OpenCV performs real face detection, but the function is a bit tedious to use directly, so it is easiest to use this function: *Haarcascade_frontalface_default.xml; Haarcascade_frontalface_alt.xml; Haarcascade_frontalface_alt2.xml; Haarcascade_frontalface_alt_tree.xml*.

5. Conclusions and future research directions

The article shows the techniques that need to be performed to detect the face. When moving, faces can be varied in size or rotated in different directions. The next step is to make sure that the face can be corrected when not in the vertical direction (can be turned left, right, face up, bent down) and the person moving with fast speed.

References:

[1] Christophoros Nikou. Image Analysis PCA and Eigenfaces. University of Ioannina - Department of Computer Science, URL: http://www.cs.uoi.gr/~cnikou/Courses/Image_Analysis/2011-2012/10_PCA_and_Eigenfaces_2spp.pdf, 2003.
[2] Cheng-Yuan Zhang, Qiu-Qi Ruan. Face Recognition Using L-Fisherfaces. Journal Of Information Science And Engineering 26, 1525-1537. URL: http://www.iis.sinica.edu.tw/page/jise/2010/201007_23.pdf, 2010
[3] Joost van de Weijer, Cordelia Schmid. Coloring Local Feature Extraction. Conference Paper, DOI: 10.1007/11744047_26. Publisher: 0302-9743. URL: https://www.researchgate.net/figure/Overview-of-a-local-feature-based-method-divided-in-a-feature-detection-and-a-feature_fig1_221304835, 2006.
[4] David Barina. Gabor Wavelets in Image Processing. URL: https://arxiv.org/abs/1602.03308, 2016.
[5] Salvador Garcia, Joaquin Derrac ; Jose Cano ; Francisco Herrera. Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study. Published in: IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 34), 2012.

[6] R. Berwick. An Idiot's guide to Support vector machines (SVMs). Village Idiot - Massachusetts Institute of Technology, USA. URL: http://web.mit.edu/6.034/wwwbob/svm-notes-long-08.pdf
[7]. URL: https://en.wikipedia.org/wiki/Principal_component_analysis
[8]. Alaa Tharwat, Tarek Gaber, Abdelhameed Ibrahim, Aboul Ella Hassanien. Linear discriminant analysis: A detailed tutorial - in AI Communications 30(2):169-190, with 23,752 Reads- DOI: 10.3233/AIC-170729, 2017.
[9] Quan Wang. Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models. Rensselaer Polytechnic Institute, 110 Eighth Street, Troy, NY 12180 USA, URL: https://arxiv.org/pdf/1207.3538.pdf, 2014
[10] URL: https://docs.opencv.org/3.4.3/d7/d8b/tutorial_py_face_detection.html

Merging Brain Computing Interface (BCI) & Neural Networks for Better Authentication & Recognition

Qaseem Ramzan
Vladimir Syryamkin
National Research Tomsk State University; toggleTechs (a creativity platform)
qaseemramzan0@gmail.com

Brain Computing Interface (BCI) has been proved helpful for the different streams of technology, considering the sensitivity of data in the current era it is required to build new security protocols and authentication models. Just like other fields of technology Brain Computing Interface could also be useful for making the data security better by using BCI as an authentication method without any hard physical inputs. The focus of the issue shifts to 'recognition' of EEG signals pattern and making the authentication model self-learning to increase its efficiency. This leads us to involve Artificial Neural Networks in the authentication system to make it efficient and intelligent.

A brain computer interface (BCI), sometimes called a mind-machine interface (MMI), direct neural interface (DNI), or brain–machine interface (BMI), is a direct communication pathway between an enhanced or wired brain and an external device. BCIs are often directed at researching, mapping, assisting, augmenting, or repairing human cognitive or sensory-motor functions.

BCIs comprise an active area of research and could start to integrate advances from adjacent fields such as neuroscience, nanomaterials, electronics miniaturization, and machine learning. For example, one neuro-imaging research project is starting to make guesses as to what participants see during brain scans, purporting to be able to distinguish between a cat and a person. Merging this kind of functionality with BCIs might produce new applications. Other experimental BCI projects have been proposed. One is Neocortical Brain-Cloud Interfaces: autonomous nanorobots that could connect to axons and neuronal synaptic clefts, or embed themselves into the peripheral calvaria and pericranium of the skull. Another project, Brainets, envisions linking multiple organic computing units (brains) to silicon computing networks. A third project is Neural Dust, in which thousands of 10-100 micron-sized free-floating sensor nodes would reside in the brain and provide a computing processing network.

Combining Brain Computing Interface (BCI) with Neural Networks. Presently ever device is under a threat of security breach and taking into account the sensitivity of personal data and how human lives are half digital makes this issue an important one for humans. It is required to secure our devices with a more secure and innovative protocols instead of the mainstream methods. All the past authentication methods being used by us are breach-able and have a history of either hack attacks or simple security breach due to the weak type of authentication models. All of the authentication models we have so far need an input from the user. This input method is the main loophole which makes it not good enough to protect our devices.

If the authentication models will need a physical input from users then it is highly likely that the user is at risk because the input can be acquired either by trick or by force.

An authentication model based on EEG signals was proposed to overcome this problem and it could be a method of thoughts without any hard physical input. BCI is capable of reading the Brain Signals and then the proposed model in previous paper was developed to authenticate a user based on brain signals and matching the patter with the one stored in a database.

Authentication for BCI with Neural Networks. As we have developed a consensus above that Neural Networks can perform recognition in a very impressive way which is not possible otherwise. So a better thought is to use Neural Networks on the EEG Based Authentication Model and make the Neural Network learn the patterns of user's authentication routine. Here the Neural Network will not only match the pattern to authenticate the users but it will learn gradually the changes in EEG Signals pattern and become more intelligent in a way to recognize the user.

The ultimate task will be to train the Neural Network to the point where it will not need to match the EEG Signals with the first every stored pattern but it will be able to recognize the user based on the later learnings which were learnt in result of trainings.

References:

[1] Brain Computer Interface. [Electronic resource] URL: https://en.wikipedia.org/wiki/Brain–computer_interface
[2] History of Brain Computer Interfaces. [Electronic resource] URL: http://www.brainvision.co.uk/blog/2014/04/the-brief-history-of-brain-computer-interfaces
[3] Qaseem Ramzan. Evolution of BCI and Proposed EEG Based Authentication Model. [Electronic resource] URL: https://www.matec-conferences.org/articles/matecconf/abs/2018/14/matecconf_imet2018_01006/matecconf_imet2018_01006.html