FEATURE DETECTION ON VIDEO DATA

Feature Detection on Video data.

By,

Deepak Soundararajan

(Group 11)

Other Group members:

Anoop Jatavallabha Vijayakumar

Narendra Kumar Sampath Kumar

Ravikiran Tangirala

Santosh Mandya Jayaram

ABSTRACT

Multimedia data such as audio, video, image are widely used in today's world. The need to represent the video in a compact form is increasing due to the size and dynamic nature of the video. Feature detection, selection and extraction in a multimedia data is very challenging due to the dynamic nature of the video. In this project, detection of the features from the video is done. Firstly, the video is divided into its component frames, then the frames are divided into blocks of data. Next, the features of the video are detected. The color distribution is detected from the image and represented by color histogram. The local features in the several frames of the video are detected and represented using Scale Invariant Feature Transform(SIFT) vectors. The detection of movement of the features from one frame to another frames is extracted and represented using motion vectors.

KEYWORDS

Histogram, SIFT, Feature detection, motion vectors, motion estimation

1. INTRODUCTION

1.1. TERMINOLOGY

- Color histogram – A graphical representation of distribution of colors in an image [1].
- SIFT – Scale Invariant Feature Transforms is an algorithm that finds local features in an image [2].
- Macroblock – Macroblock is a processing unit in image and video formats based on block transforms, it typically consists of 16x16 samples [3].
- Motion Vector – A key element in the process of motion estimation which is used to represent macroblock in picture when compared to another picture [4].
- Motion Estimation – A process of determining motion vectors that describe the transformation from one 2D image to another mainly between adjacent frames of a video sequence [5].
- Features – A feature is a piece of information that is relevant for solving the computational tasks related to the application [6].
- Feature detection – A process of finding whether an image point has an image feature [7].
- Frames - An electronically coded still image in a video [8].
- Feature Vector – A n-dimensional vector of numerical values that represent an object [9].

1.2. GOAL DESCRIPTION

In Task 1 of the project, a directory is given containing the video files, a resolution r is given to divide the video into r*r cells. For each cell in the given frame of the video, the color distribution of the data in the cell is detected and displayed into n bin color histogram where n bin divides the whole color space in n different groups and displayed.

In Task 2 of the project, the directory containing the video files are given, each frame in the video is divided into r*r cells. For each cell in each frame of the videos, local features are extracted using the Scale Invariant Feature Transform (SIFT) algorithm and represent as a SIFT vector which contains the location of the point in a x, y axis, scale, orientation and description vector of that SIFT key points which is displayed.

In Task 3 of the projects, the directory containing the video files are given, each frame in each of the video is divided into r*r cells. For macroblocks in each cell in each frame of the videos, motion vector of objects in that cell are detected and displayed.

1.3. ASSUMPTIONS

a. All the video files, for which the features need to be detected and extracted are presented in a single directory.
b. The assumption made in the project as a whole is that the resolution r for dividing each from into r*r cells be taken as 1,2 to 8 would be the ideal values in order to avoid high computation time.
c. The cell size r is taken such that it can divide the block without any remaining elements (ie) frame height and width are divisible by r.
d. The value of n (number of bins) for the n bin color histogram (Task 1) is between 0 to 255.
e. The macroblocks having their source before moving into the current frame are not represented under any of the r*r cells.
f. For Task 2, the sift library must first be compiled before running the MATLAB code.
g. The dirent.h library must be available in order to get the input directory and compute the motion vector for all the video files.
h. The output files for task 1 and 2 are stored in the directory where the video files are found.
i. The input directory for the task 3 contains only the video files.

2. DESCRIPTION OF THE PROPOSED SOLUTION

Task 1:

Input: Directory of video files (dir), Resolution (r), Number of Bins (n) for histogram.

Workflow of the proposed solution:

1. Get directory, resolution and number of bins for histogram from the user.
2. From the given directory read the number of video files in the directory.
3. With a video file as input, compute the number of frames in the given video file.
4. For each frame in the video file, convert the frame from RGB color scale to Y (Grayscale) color channel.
5. Compute the number of rows and columns for each of the r*r block.
6. With the rows and columns of each cell, convert the whole frame into r*r blocks.
7. Compute the color distribution for each block and represent them as n bin histogram.
8. Write the n bin histogram counts into the output file and save it to video directory.

Output: N-Bin Histogram for each of the r*r cells for all frames of all the files in the given directory

Task 2:

Input: Directory of video files (dir), Resolution (r).

Workflow of the proposed solution:

1. Get directory, resolution from the user.
2. From the given directory read the number of video files in the directory.
3. With a video file as input, compute the number of frames in the given video file.
4. For each frame in the video file, convert the frame from RGB color scale to Y (Grayscale) color channel.
5. Find the sift vector for the whole frame.
6. Compute the size of rows and columns for the r*r cell.
7. With the size of rows and size of columns and the x,y from the sift vector and compute the cell number on which the x,y of the sift vector belongs and assign the sift vector to that cell.
8. Display the sift vector and its cell into the output file and save it to video directory.

Output: SIFT vectors represented along with the cell number they belong to for all frames of all the files in the given directory

Task 3:

Input: Directory of video files (dir), Resolution (r)

Workflow of the proposed solution:

1. Get directory, resolution from the user.
2. From the given directory read the number of video files in the directory.
3. For each video file as input, get one frame and decode the video stream.
4. Now with the decoded side data of the frame, find the frame width, height, source of the macroblock along x axis, source of the macroblock along y axis, destination of the macroblock along x and y axis. For motion vector computation since we refer the current frame with previous or future frame, destination of the macroblock represents the current frame on which motion vectors have been determined.
5. With the destination x and y axis values, compute onto which r*r cell the value falls under.
6. Output this motion vector along with its frame, block number and video name and output in the C++ project directory.

Output: Motion Vector source along x and y axis, destination along x and y axis, motion vector block width and height and motion vector flags along with the video name, frame number and cell number.

3. SYSTEM REQUIREMENTS AND EXECUTION INSTRUCTIONS

System Requirements:

Processor: i3, i5, i7

RAM: 2 GB

Hard Drive: 500 GB

Software Requirements:

Programming Languages: MATLAB, C.

Operating System: Windows 7 or higher.

Software Tools: MATLAB, Microsoft Visual Studio.

Execution Instructions:

Task 1:

1. Download the MATLAB software for your operating system.
2. Install the MATLAB software with all functionalities.
3. Open the MATLAB code file “task1histogram.m”.
4. Compile the program and run the program.
5. Input directory containing the video files, and resolution to divide the cells.
6. Your output file output_chst.txt will be saved in the directory of the video

Task 2:

1. Download the MATLAB software for your operating system.
2. Install the MATLAB software with all functionalities.
3. Download the MEX add on for MATLAB.
4. Download the SIFT library.
5. Compile the sift file "sift_compile.m"
6. Open the MATLAB code file “task1sift.m”.
7. Compile the program and run the program.
8. Input directory containing the video files, and resolution to divide the cells.
9. Your output file output_sift.txt will be saved in the directory of the video.

Task 3:

1. Download the Microsoft Visual Studio for your operating system.
2. Install the Visual Studio with all functionalities.
3. Open the ConsoleApplication1 C++ project with Visual Studio.
4. Now run the application.
5. Input the directory and resolution to divide the cells.
6. Your output file output_mvect.txt will be saved in the project directory.

4. RELATED WORK

Yang et al describes SIFT algorithm as a target tracking algorithm that finds the target by matching features between two adjacent frames [10]. The feature points from the SIFT algorithm adapts the transformations in the image such as scale, rotations etc., but the efficiency of the SIFT algorithm is limited by the large computation filter modules. To improve the operation speed of the SIFT algorithm, the author describes a method in which a narrow window is set and extreme points are found in this space following which the feature points are detected and the direction in which they point to are calculated following which key point descriptors are detected. The paper also discusses on enhancing the robustness of the block matching by keeping only the directions with 80% of the peak are retained.

Hino et al describes motion vector estimation using gradient based method as a technique that uses relationship between the gradient of the image and intensity difference between the frames [11]. The author also proposes a two block gradient based technique to detect the motion vector on a variable block shape and size of an image where in for the variable block shape, the motion vector estimation is done at every pixel where as for the variable size the motion vector is estimated at each block. The author further describes the determination of the block size in a gradient based method influences the accuracy of estimation mainly at the motion border areas.

Acharjee et al describes the compression and decompression process for video communication as the same wherein the motion estimation, a process of examining the movement of objects in a sequential manner to get the vectors to represent the estimated motion is a most time consuming process [12]. The author proposes an algorithm that defines the motion vector at each pixel with the assumption that the pixels can move up to 3 positions. The author compares the new algorithm with other algorithms and describes that this algorithm reduces the computational time of estimating the motion vector.

5. CONCLUSIONS

The features of a video namely, color distribution, local features and motion vectors in a video were detected and represented using color histogram, SIFT vectors and motion vectors respectively. In future, these features will be used to compute and perform analysis on the video data.

6. REFERENCES

[1]. 9/12/2016, https://en.wikipedia.org/wiki/Color_histogram

[2]. 9/12/2016 https://en.wikipedia.org/wiki/Scale-invariant_feature_transform

[3]. 9/12/2016 https://en.wikipedia.org/wiki/Macroblock

[4]. 9/12/2016 https://en.wikipedia.org/wiki/Motion_vector

[5]. 9/12/2016 https://en.wikipedia.org/wiki/Motion_estimation

[6]. 9/12/2016 https://en.wikipedia.org/wiki/Feature_(computer_vision)

[7]. 9/13/2016 https://en.wikipedia.org/wiki/Feature_detection_(computer_vision)

[8]. 9/13/2016 https://en.wikipedia.org/wiki/Frame_(video)

[9]. 9/13/2016 https://en.wikipedia.org/wiki/Feature_vector

[10]. Cong Yang, Liu Wanyu, Zhang Yanli, Liang Hong. “The research of video tracking based on improved SIFT algorithm.” Mechatronics and Automation (ICMA), 2016 IEEE International Conference on.

[11]. Hino, Masatoshi, and Nozomu Hamada. "Gradient-based motion vector estimation using variable block shape." *Signal Processing Conference, 2002 11th European*. IEEE, 2002.

[12]. Acharjee, Suvojit, and Sheli Sinha Chaudhuri. "A new fast motion vector estimation algorithm for video compression." *Informatics, Electronics & Vision (ICIEV), 2012 International Conference on*. IEEE, 2012.

Other References:

1. Lecture - 24 Motion Estimate Techniques by Professor. Senguptha, IIT Kharagput. https://www.youtube.com/watch?v=Tm4C2ZFd3zE
2. http://www.mathworks.com/help/matlab/ref/histcounts.html
3. http://www.mathworks.com/help/matlab/ref/videoreader.html
4. https://www.mathworks.com/matlabcentral/answers/116190-i-want-to-divide-256-256-image-into-8-8-blocks-and-store-the-8-8-blocks-in-the-matrix-or-2d-array-so
5. http://www.gnu.org/software/libc/manual/html_node/Directory-Entries.html