Report

ABSTRACT

In this report we propose algorithms for merging images captured by two webcams to form a single image of greater dimension, and removing the redundant or repeated information as in case of overlap or using spatial interpolation to predict the missing portion between the original images. A column-based approach has been used in the algorithms. In case of overlap between images, the repeated portion is detected using auto-correlation and is removed. In case of missing data between the two images, the scheme used involves linear extension of edges surrounding the missing portion to form the missing edges followed by spatial interpolation to predict the missing pixels.

Designing a GUI for Taking images

Graphical User Interface is designed in MatLab as an interface to avoid entering commands everytime images need to be captured. The following GUI is designed to capture two images from two different webcams and then concatenate them.



The two images we simply concatenated to form single image.

Algorithm for Removing Overlap

When two webcams (assumed to be on the same horizontal level) are placed close enough such that their field of views coincide, the images captured by them will overlap. In order to remove the overlap before concatenating the images, the overlapping portion needs to be detected. The auto-correlation tool is a simple tool available for this purpose.

A number of different approaches can be made towards the problem. The following section explains the different approaches that can be followed.

A. Block by block method

This method involves dividing the two images into a number of blocks of fixed dimension and comparing them. The method has been discussed below.

Approach:

1. Divide the images obtained into blocks

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Image 1 Image 2

1. Select one block from first image

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Selected Block

3. Now this block is matched with subsequent blocks of the 2nd image. There may be a match or no match

In case of no match, blocks of 2nd image are shifted by some pixels to the right. The comparison is carried again.



|  |  |  |  |
| --- | --- | --- | --- |
|  | Found the match |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Blocks shifted by some pixels

4. After a match is found, remove the portion to the left of the matching portion and concatenate the reamaining image with the 1st image.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |



Resulting Image

Limitations:

1. Computation time is large.
2. Difficult to implement on digital level.

B. Pixel by pixel method

This method involves pixel by pixel comparison between two images.



Image 1 Image 2

Limitations:

1. Got false results at times.
2. Cumbersome with large computation time.

C. Columnar Approach

It is assumed here that the images are bounded by the same horizontal levels as the cameras are on the same horizontal level. The proposed algorithm uses the a vertical strip, 5 pixels wide (the rightmost 5 pixels) belonging to the first image and finds the autocorrelation of this strip with subsequent 5-pixel wide strips of the second image, starting from its leftmost pixels. The strips which match perfectly, pixel to pixel, will have autocorrelation value ‘1’ while those with no pixel matching, will have autocorrelation value ‘0’. Others will have values lying between 0 and 1. By calculating the autocorrelation coefficient of the base strip (1st image) with other strips (2nd image), a comparison is being made between the images. The strip for which autocorrelation coefficient with the base strip is above a given threshold, is the strip which gives the point where the overlap ends. The pixels to the left of this strip including this strip are then cropped out, and the remaining portion is concatenated with the first image.

Algorithm

Both images I and J are converted to grayscale as the correlation function works on 1-D and 2-D arrays. Initially, DWT decomposition technique was used to get decomposed images which would then be used for comparison. This method would decrease the computation time but was inaccurate at times.



Image 1 (I) Image 2 (J)

The base strip comprising last 5 horizontal pixels is stored in a dummy varaiable ‘test’.

Considering that at same horizontal plane

‘test’ is compared with the array starting from ‘set’ to the next 5 pixels.



Comparison with moving strip towards right.

Comaprison is done using correlation function ‘corr2’. If the value is more than a threshold value then the point where overlap ends has been found. If at this vertical level,match could not be found out then test is moved up and 2nd image is moved down to compare. In case match is found then image are combined.



Resulting Image

Features:

1. Reletively lesser computation time.

2. Need a seperate calibration program to be written to make it fully automated.

(the threshold value needs to compared with autocorrelation coefficient obtained)

References:

(1) Matlab Help

(2) [www.mathworks.com](http://www.mathworks.com)

Algorithm for Spatial Interpolation and Prediction of Missing Portion Between Images

The algortihms proposed by [1] and [4] for prediction of information in lost during transmission over a communication channel involve methods which provide very good results for:  
1. Prediction of a small patch of image

2. Usage of code using high performance computing device for heavy computations.

In our case we have the following restrictions to work within:

1.Relatively large strip needed to be predicted.

2.Algorithm needs to be simple enough to be implemented on digital level.

Approach:

 

Image 1 Image 2

1. Detecting edges

A small portion from the right end of the left image and from the left end of the second image is taken to be combined with an empty region having width equal to the region to be interpolated. The edges within this region are detected using a Sobel edge detector.



Using Sobel Operator we can detect edges in this image.

After detecting edges in binary image. Or we can say dominant edges depending upon threshold value for the definition of dominant edges. As we can see the middle portion is blank ,since it was the protion to be predicted. We first predict edges int his portion.

Edges can be variety of shapes, Namely lines,circles,ellipse.

Some edges will run thorugh this blank image,some will die out in this portion,some will retreat back to the portion they come from.

How to join the lines.

[1] presents extensive usage of Hough tranform and its usage which might be difficult to implement at digital level,Thus we devised a simpler method.

Only straight lines have been considered since the portion to be predicted would be quite narrow in the real world. The probability of curve which retreats back is less since the space to be predicted is less. Also ellipse, circle or any other curve can be regarded as straight lines for a small portion of length.

Now we will start first from left portion and search for first sequence of continuous lines and will find the end point of it . We then find out approximate connection at right cropped image based on its slope.

1.Here we can make use of Hough tranform to know most dominant line and their slope and perpendicular distance from origin.

2. Or we can make use algorith to detect dominant lines based on finding dominant lines out of some portion.

After that lines need to joined.For this to find out another end in the right portion of the image it start its search from most probable point i.e. according to slope calculations . If not found then it run the search operation away from point towards the direction of last searched point and if not found even then it found

After the edges are joined region based interpolation is done in order to combine the image.This can be done ycbcr format for color image or grayscale for black and white image.

We here use bilinear interpolation first and kernel function is applied to smoothen the image. We execute the code row wise. We first see if there is any edge in a given row. If no then a simple bilienar interpolation is used.

If there is an edge then color of left are forced till the edge and pixels of right is forced right upto the edge.

Image after joining edges Predicted Image



Resulting image after prediction

References

[1] *Hamid Gharavi, and Shaoshuai Gao, “*Spatial Interpolation Algorithm for Error Concealment*”,* presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2008)

[2] [www.mathworks.com](http://www.mathworks.com)

[3] [www.wikipedia.org](http://www.wikipedia.org)

[4] *Paul Salama, Ness B. Shrofl, Edward* ***b.*** Coyle, *and Edward J. Del, “*Error Concealment Techniques for

Encoded Video Streams*”*