Edge Detection in Video Streaming By Canny Algorithm Using Texture Analysis

P. S. NAVEEN KUMAR, K.DIVIYA

Assistant Professor, Dept of MCA, St.Ann's College of Engineering & Technology, Chirala,
PG Student, Department of MCA, St.Anns College of Engineering & Technology, Chirala

ABSTRACT-: The main purpose of the project develops the real time detection. It will be useful in medical diagnosis center, for x-ray or such types of scanning. Here we are executing the real time edge detection with multiple types of algorithms. At last we showing that canny is the best process for edge detection. The continuously captured system will be shown in discrete of edge detection type. Most computer systems normally do with computer vision with gray scale image. We cannot identify the features in gray scale intensities. so we are proposing edge detection with by threshold process with different types of image processing techniques, by using python software.

Keywords-: Canny edge detection, Smoothing process, Feature extraction,

INTRODUCTION

Edge detection is extensively used in image segmentation to divide an image into areas corresponding to different objects. In a picture, an edge is normally defined as an abrupt change in colour intensity. Human's eyes use a much more complicated method to find edges. This is because we have two eyes (therefore stereoscopic vision and depth perception) as well as our incredible inference skills (we can "see" the grey square above, despite it being obscured by the circle). Despite this, most computer vision systems must do with one (normally grayscale) camera, so change in colour intensity is the next best thing. Edges occur in parts of the image with strong intensity contrast, which often represent object boundaries. Edges characterize object boundaries useful for identification of object in a scene such as an X-Ray image. Edges characterize object boundaries useful for identification of object in a scene such as an

X-Ray image. Determining bone edges is important because it can provide surgeons with important information for diagnosis, which in turn enables them to give better treatment decision to their patients. Edge detection is extensively used in image segmentation to divide an image into areas corresponding to different objects. Image segmentation is widely used in many areas including.

RELATED WORK:

Anny Edge Detection is a popular edge detection algorithm. It was developed by John F. Canny in

1. It is a multi-stage algorithm and we will go through each stages.

2. Noise Reduction: Since edge detection is susceptible to noise in the image, first step is to remove the noise in the image with a 5x5 Gaussian filter. We have already seen this in previous chapters.

3. Finding Intensity Gradient of the Image: Smoothened image is then filtered with a Sobel kernel in both horizontal and vertical direction to get first derivative in horizontal direction (Gx) and vertical direction (Gy). From these two images, we can find edge gradient and direction for each pixel as follows:

Journal of Engineering Sciences
Vol 11, Issue 6, JUNE /2020
ISSN NO:0377-9254

Edge_Gradient(G)=G2x+G2y−−−−−−−√Angle(θ)=tan−1(GyGx)

Gradient direction is always perpendicular to edges. It is rounded to one of four angles representing vertical, horizontal and two diagonal directions.

EXISTING METHODOLOGY:

Local Binary Patterns are among the recent texture descriptors. The original LBP operator replaces the value of the pixels of an image with decimal numbers, which are called LBPs or LBP codes that encode the local structure around each pixel [19–21]. Each central pixel is compared with its eight neighbors; the neighbors having smaller value than that of the central pixel will have the bit 0, and the other neighbors having value equal to or greater than that of the central pixel will have the bit 1. For each given central pixel, one can generate a binary number that is obtained by concatenating all these binary bits in a clockwise manner, which starts from the one of its top-left neighbor. The resulting decimal value of the generated binary number replaces the central pixel value. The histogram of LBP labels (the frequency of occurrence of each code) calculated over a region or an image can be used as a texture descriptor of that image. The size of the histogram is 2P since the operator LBP(P,r) is able to generate 2P different binary codes, formed by the *P* neighboring pixels. Recently, several LBP variants have been developed in order to improve the texture description [22,23].

For describing a segmented region with LBP descriptors, in our work, we use eight neighboring points (P=8) with three radii (r=1, r=2, r=3), each with three modes (uniform, rotation invariant, uniform and rotation invariant). Thus, there are nine LBP descriptors. The final descriptor is given by the concatenation of all. It is worth noting that despite the use of nine LBP descriptors, the final one is described by 3×(59+36+10)=315 variables only.

Thresholding process:

the simplest thresholding methods replace each pixel in an image with a black pixel if the

image intensity is less than some fixed

constant T (that is,), or a white pixel if the image intensity is greater than that constant. In the example image on the right, this results in the dark tree becoming completely black, and the white snow becoming completely white.

To make thresholding completely automated, it is necessary for the computer to automatically select the threshold T. Sezgin and Sankur (2004) categorize thresholding methods into the following six groups based on the information the algorithm manipulates (Sezgin et al., 2004):

- **Histogram shape**-based methods, where, for example, the peaks, valleys and curvatures of the smoothed histogram are analyzed
- **Clustering**-based methods, where the gray-level samples are clustered in two parts as background and foreground (object), or alternately are modeled as a mixture of two Gaussians
- **Entropy**-based methods result in algorithms that use the entropy of the foreground and background regions, the cross-entropy between the original and binarized image, etc.[1]
- **Object Attribute**-based methods search a measure of similarity between the gray-level and the binarized images, such as fuzzy shape similarity, edge coincidence, etc.
- **Spatial** methods [that] use higher-order probability distribution and/or correlation between pixels

JES Journal of Engineering Sciences
Vol 11, Issue 6, JUNE /2020
ISSN NO:0377-9254

- **Local** methods adapt the threshold value on each pixel to the local image characteristics. In these methods, a different T is selected for each pixel in the image.

PROPOSED METHODOLOGY:

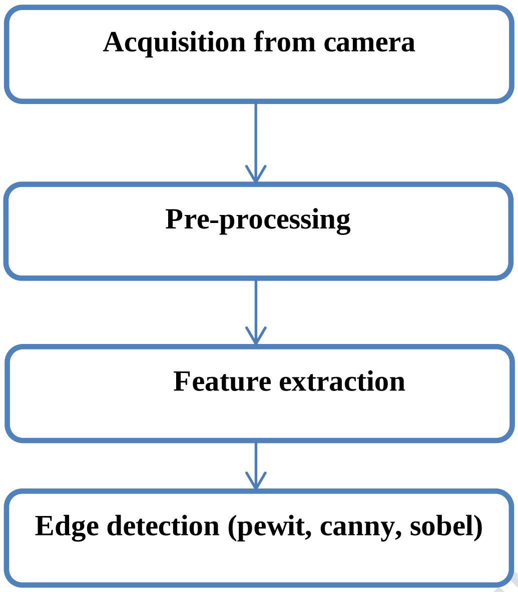


IMAGE AQUIZATION:

Image Acquisition in Digital Image Processing. In **image processing**, it is defined as the action of retrieving an **image** from some source, usually a hardware-based source for **processing**. It is the first step in the workflow sequence because, without an**image**, no **processing** is possible.the image that is is aquired completely unprocessed.

Now the incoming energy is tranformed intto voltage by the combination of input electrical power and sensor materail that is responsiveto a particular type of energy being detected.the output voltage waveform is the response of the sensor and digital quantity is obtaibned from the each sensor by digitizing its response.

Also we can defines as digital image is produced by one or several image sensors, which, besides various types of light-sensitive cameras, include range sensors, tomography devices, radar, ultra-sonic cameras, etc

Depending on the type of sensor, the resulting image data is an ordinary 2D image, a 3D volume, or an image sequence. The pixel values typically correspond to light intensity in one or several spectral bands (gray images or colour images), but can also be related to various physical measures, such as depth, absorption or reflectance of sonic or electromagnetic waves, or nuclear magnetic resonance.

PRE-PROCESING:

Pre-processing is a common name for operations with images at the lowest level of abstraction -- both input and output are intensity images.o The aim of pre-processing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing

Feature Extraction:

The acquired data is first subjected to a pre processing step. Besides filtering for noise removal, this step also processes the signal for achieving invariance to selected inspection parameters. For instance, in the case of inspection data acquired at different inspection frequencies the signals are first transformed to an equivalent signal at a reference value of the inspection frequency parameter. Similarly, the overall classification performance of the system can be rendered invariant to other selected parameters [3]. In the second step, discriminatory features in the signal are extracted. Feature extraction serves to reduce the length of the data vector by eliminating redundancy in the signal and compressing the relevant information into a feature vector of significantly lower dimension. The Discrete Wavelet Transform (DWT) is particularly effective at extracting features at multiple resolution levels in ultrasonic signals which are inherently non-stationary in nature. A

JES Journal of Engineering Sciences
Vol 11, Issue 6, JUNE /2020
ISSN NO:0377-9254

second set of features based on Principal Component Analysis (PCA) also calculates the statistical properties of a set of neighboring A-scans. The rationale underlying this approach is that the irregular nature of the IGSCC causes the variance of signals in the crack region to be vastly different from the variance of reflections obtained from a counter bore region. The PCA exploits this information to discriminate between cracks and counter bores.

Edge detection:

Types of edges

Generally edges are of three types:

- Horizontal edges
- Vertical Edges
- Diagonal Edges

Why detect edges

Most of the shape information of an image is enclosed in edges. So first we detect these edges in an image and by using these filters and then by enhancing those areas of image which contains edges, sharpness of the image will increase and image will become clearer.

Here are some of the masks for edge detection that we will discuss in the upcoming tutorials.

- Prewitt Operator
- Sobel Operator
- Robinson Compass Masks
- Krisch Compass Masks
- Laplacian Operator.

Above mentioned all the filters are Linear filters or smoothing filters.

Prewitt Operator

Prewitt operator is used for detecting edges horizontally and vertically.

Sobel Operator

The sobel operator is very similar to Prewitt operator. It is also a derivate mask and is used for edge detection. It also calculates edges in both horizontal and vertical direction.

Robinson Compass Masks

This operator is also known as direction mask. In this operator we take one mask and rotate it in all the 8 compass major directions to calculate edges of each direction.

Kirsch Compass Masks

Kirsch Compass Mask is also a derivative mask which is used for finding edges. Kirsch mask is also used for calculating edges in all the directions.

Laplacian Operator

Laplacian Operator is also a derivative operator which is used to find edges in an image. Laplacian is a second order derivative mask. It can be further divided into positive laplacian and negative laplacian.

All these masks find edges. Some find horizontally and vertically, some find in one direction only and some find in all the directions. The next concept that comes after this is sharpening which can be done once the edges are extracted from the imageSharpening

Sharpening is opposite to the blurring. In blurring, we reduce the edge content and in Sharpening, we increase the edge content. So in order to increase the edge content in an image, we have to find edges first.

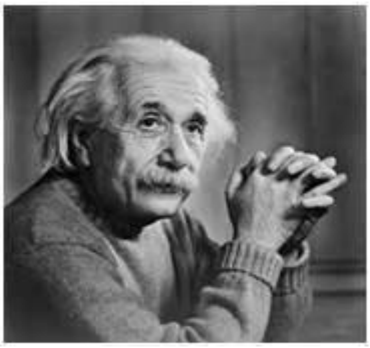
Edges can be find by one of the any method described above by using any operator. After finding edges, we will add those edges on an image and thus the image would have more edges, and it would look sharpen.

This is one way of sharpening an image.

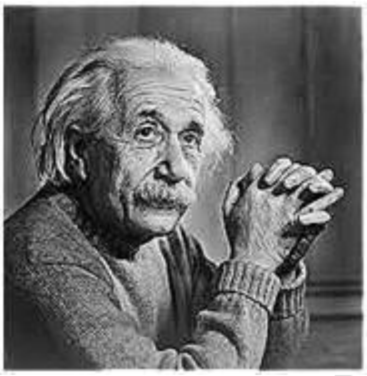
The sharpen image is shown below.

Vol 11, Issue 6, JUNE /2020
ISSN NO:0377-9254

ORIGINAL IMAGE



SHARPEN IMAGE



Prewitt operator is used for edge detection in an image. It detects two types of edges

- Horizontal edges
- Vertical Edges

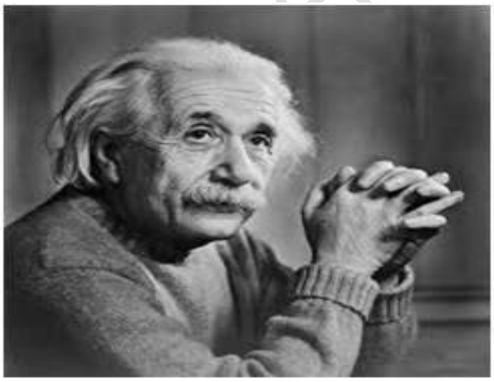
Edges are calculated by using difference between corresponding pixel intensities of an image. All the masks that are used for edge detection are also known as derivative masks. Because as we have stated many times before in this series of tutorials that image is also a signal so changes in a signal can only be calculated using differentiation. So that's why these operators are also called as derivative operators or derivative masks.

All the derivative masks should have the following properties:

- Opposite sign should be present in the mask.
- Sum of mask should be equal to zero.
- More weight means more edge detection.

Prewitt operator provides us two masks one for detecting edges in horizontal direction and another for detecting edges in an vertical direction.Sample Image

Following is a sample picture on which we will apply above two masks one at time.



After applying Vertical Mask

After applying vertical mask on the above sample image, following image will be obtained. This image contains vertical edges. You can judge it more correctly by comparing with horizontal edges picture.

After applying Horizontal Mask

Vol 11, Issue 6, JUNE /2020
ISSN NO:0377-9254

After applying horizontal mask on the above sample image, following image will be obtained.

Comparison

As you can see that in the first picture on which we apply vertical mask, all the vertical edges are more visible than the original image. Similarly in the second picture we have applied the horizontal mask and in result all the horizontal edges are visible. So in this way you can see that we can detect both horizontal and vertical edges from an image.

SOBEL:

The sobel operator is very similar to Prewitt operator. It is also a derivate mask and is used for edge detection. Like Prewitt operator sobel operator is also used to detect two kinds of edges in an image:

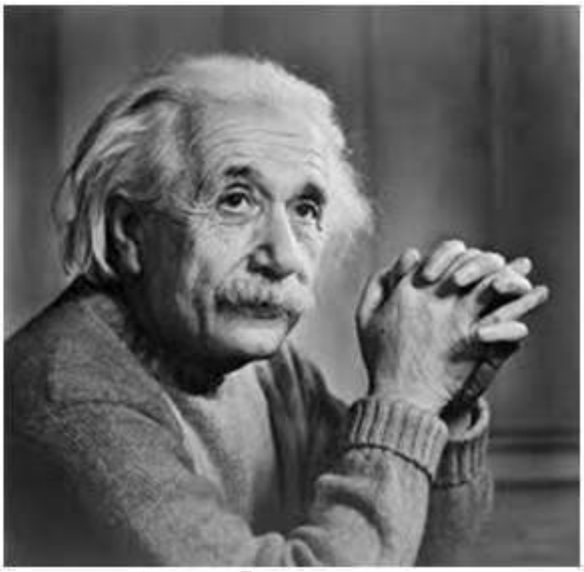
- Vertical direction
- Horizontal direction

Difference with Prewitt Operator

The major difference is that in sobel operator the coefficients of masks are not fixed and they can be adjusted according to our requirement unless they do not violate any property of derivative masks.

Sample Image

Following is a sample picture on which we will apply above two masks one at time.



After applying Vertical Mask

After applying vertical mask on the above sample image, following image will be obtained.

After applying Horizontal Mask

After applying horizontal mask on the above sample image, following image will be obtained

Comparison

As you can see that in the first picture on which we apply vertical mask, all the vertical edges are more visible than the original image. Similarly in the second picture we have applied the horizontal mask and in result all the horizontal edges are visible.

So in this way you can see that we can detect both horizontal and vertical edges from an image. Also if you compare the result of sobel operator with Prewitt operator, you will find that sobel operator finds more edges or make edges more visible as compared to Prewitt Operator.

This is because in sobel operator we have allotted more weight to the pixel intensities around the edges.

CANNY:

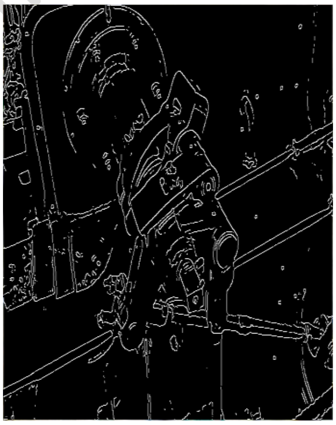
Canny Edge Detection is used to detect the edges in an image. It accepts a gray scale image as input and it uses a multistage algorithm.

You can perform this operation on an image using the **Canny()** method of the **imgproc** class, following is the syntax of this method.

Canny(image, edges, threshold1, threshold2)

This method accepts the following parameters –

- **image** − A **Mat** object representing the source (input image) for this operation.
- **edges** − A **Mat** object representing the destination (edges) for this operation.
- **threshold1** − A variable of the type double representing the first threshold for the hysteresis procedure.
- **threshold2** − A variable of the type double representing the second threshold for the hysteresis procedure.



CONCLUSION:

Using this edge detection detect the edges of images. This can be applicable by using pewit filter, canny filter, and canny filter.Various edge detection algorithms and detector design methods have been described and discussed in this paper.he resulting edge maps produced by the approximation to the second order derivative edge detection indicate that this model does accurately position the real edges. However, the problem of this approach is the high sensitivity to the image noise.he resulting edge maps produced by the approximation to the second order derivative edge detection indicate that this model does accurately position the real edges. However, the problem of this approach is the high sensitivity to the image noise. For this process previously we used local binary pattern. But we are using smoothing process.

FUTURE SCOPE:

For the purpose of edge detection we have canny filter, sobel filter, Palladian Filter and Pewit filter. Canny is the best for the detection of edge detection. in our project we used canny filter. by using this we can detects the edges easily. But in this processes time consuming is moderately high. to avoid this we can improve algorithm. Edge detection is used for the x-ray and we can use in future for the other applications like satellite applications, film applications etc.. For the identification process we can give colours four the image in the next generation.

REFERENCES:

[1] H. Fu, Z. Chi, *Combined thresholding and neural network approach for vein pattern extraction from leaf images*, IEE Proc.-Vis. Image Signal Process, vol. 153, no. 6, pp. 881-892, December 2006.

[2] Jibi. G. T, A. Kumar Dubey, M. T. Thomas, *Whether color, shape and texture of leaves are the key features for image processing based plant recognition? An analysis!*, Recent Developments in Control, Automation & Power Engineering (RDCAPE), pp.404-409,2017.

[3] J. Wäldchen and P. Mäder, *Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review*, Archives of Computational Methods in Engineering: State-ofthe-Art Reviews, pp. 1–37, 2017.

[4] DP Corney, Tang HL, Clark JY, Hu Y, Jin J, *Automating digital leaf measurement: the tooth, the whole tooth, and nothing but the tooth*. PLoS ONE 7(8), 2012.

[5] KJ Gaston, O'Neill MA, *Automated species identification: why not?*, Philos Trans R Soc Lond B Biol Sci, 359(1444):655–667,2004.

[6] Z A Jaffery and A. Kumar Dubey, *Scope and Prospects of Non-Invasive Visual Inspection Systems for Industrial Applications*, Indian Journal of Science and Technology, 9(4): 1-11, 2016.

[7] N. Valliammal, S. S. N. geethalakshmi, *A Novel Approach for Plant Leaf Image Segmentation using Fuzzy Clustering*, International Journal of Computer Applications, 44(13): 10-20, 2012.

[8]https://www.egr.msu.edu/classes/ece480/capstone/fall13/group04/docs/danapp.pdf

[9] S. Agaian, A. Almuntashri , Papagiannakis,A.T., *An Improved Canny Edge Detection Application for Asphalt Concrete*, Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics USA, pp.3683-3687, 2009.

[10] A Joly, Müller H, Goëau H, Glotin H, Spampinato C, Rauber A, Bonnet P, Vellinga WP, Fisher B, *LifeCLEF: multimedia life species identification*. In: International workshop on environmental multimedia retrieval 2014, Glasgow, 2014.

[11] J Wäldchen , Thuille A, Seeland M, Rzanny M, Schulze ED, Boho D, Alaqraa N, Hofmann M, Mäder P, *Flora Incognita – Halbautomatische Bestimmung der Pfanzenarten Thüringens mit dem Smartphone*. Landschaftspfege und Naturschutz in Thüringen 53(3):121–125, 2016.

[12] J. Sosa, G. Solano, J. Obico, *An application using canny edge detection and multilayer perceptron for recognizing leaves of tropical plants*, Mathematics and Computers in Biology and Biomedical Informatics, 2013.
[13] S. G. Wu et al., *A leaf recognition algorithm for plant classification using probabilistic neural network*," Proc. ISSPIT, pp. 11-16, 2007.
[14] B. Mhamed, S. Abid, F. Fnaiech, *Edge detection using wavelet transform and Neural Networks*, WULFENIA Journal, 20(5): 196-205, 2013.
[15] H. Fu and Z. Chi, *A two-stage approach for leaf vein extraction*,In International Conference on Neural Networks and Signal Processing, Nanjing, Jiangsu, China, I: 208-211, 2003.
[16] S. Bankar, Ajita Dube,Pranali Kadam, Sunil Deokule, *Plant Disease Detection Techniques Using Canny Edge Detection & Color Histogram in Image Processing*. International Journal of Computer Science andInformation Technologies, 5 (2), 2014.
[17] A. Salman, Ashish Semwal, Upendra Bhatt, *Leaf Classification and Identification Using Canny Edge Detector and SVM Classifier*, International Conference on Inventive Systems and Control (ICISC), 2017.

About Authors:

Mr.P.S.NaveenKumar, obtained his master of Technology in CSE from JNT University Kakinada and currently pursuing his Ph.D. in Computer Science and Engineering from Annamalai University chaidambaram.He has published 10+ papers published in reputed International/National Journals and conferences. He is a Life Member in IAENG. His research work focuses, Image processing, Data Mining, computer Networks and cloud computing.

K.Divya is pursuing her MCA in MCA Department. St. Ann's College of Engineering and Technology, Chirala, A.P. She received her bachelor of Science from ANU.