Smart camera monitoring System

K.kautham[1], A.Bharath Kumar[2], M.Nithyanandhan[3], S.Aravinda krishnan[4]
[1,2,3]Students of Computer Science and Engineering at SRM Institute of Science and Technology
[4]Assistant professor, Department of CSE, SRM Institute of Science and Technology,
Tamilnadu, India.

Received: February 21, 2019 Accepted: March 24, 2019

ABSTRACT:*Data collection has reached higher heights in recent years. This is due to result of storing and processi ng large amount of unstructured data. Though technologies support processing large amount of data, there is cost involved for this. That is high storage cost. Due to collection of large amount of data, huge space is required in Dat acenter. Our objective is to reduce the volume of data stored in Datacenter.*

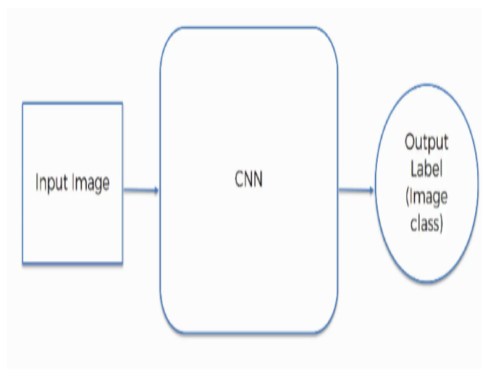
With recent development, lot of unstructured data is stored in form of images and videos. Continuous video is recor ded all over world and they are stored for review later if required. Continuous video recorded takes huge amount o f space. In our project, we propose to reduce the data stored by avoiding storing unstructured data.

Key Words: *convolutional neural network, unstructured data, cloud storage.*

I.INTRODUCTION

The main ideology behind this project is to reduce the memory required to store the data. The system stores the video only when it is seen any unstructured data. The unstructured data is differentiated by comparing each frame of the video using the convolutional neural networks. The video is made upon a certain sequence of the frame occurring respectively. The frame is nothing but an image made up of pixels which are in the range of 0 to 255. If the system captures only when there is a movement in the surrounding help to look into a certain video range only rather checking the whole video section. Each frame is converted into a grayscale image in order to reduce the complexity of comparing process because the grayscale image will show the image kind of black and white so it is easy to convert the visual picture into the matrix which consist of 1 and 0. When the area is black it converted into 1 and when the area is white then it is converted into 0.

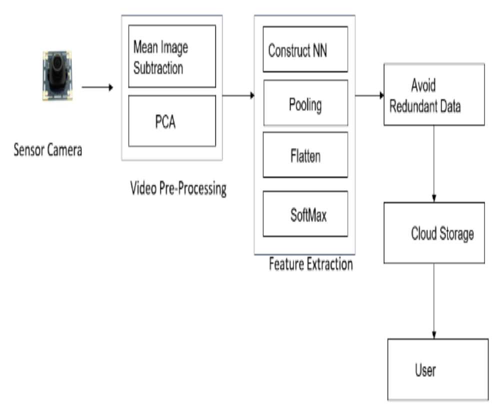
Then the each matrix is loaded into the neural network in order to compare with each other. Because it is in the order of matrix the slight changes in the pixel or a change in the pixel due to the change in the video will be noticeable abruptly. The main idea of the project is to capture only the video has movements and not wasting the memory by storing the video which has no movement. It will reduce the memory size and also manpower to reduce the time of checking a certain movement in the video.



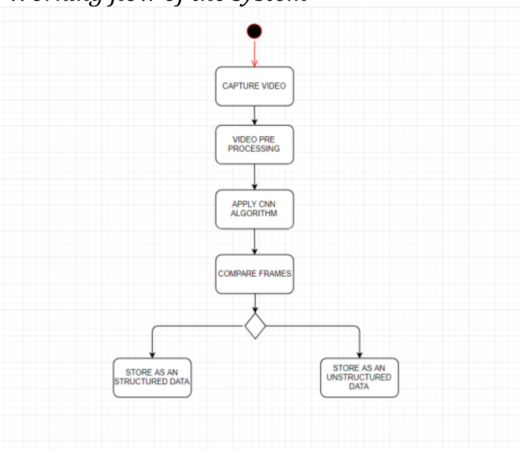
Reducing the data due to certain factor like duplicity, no movement occurred during the certain time are all reduced .also the system is capable of capturing the face whenever it is occurs with the specific time.

II.WORKINGOFTHESYSTEM

A. Architecture diagram structureofthesystem



B. Working flow of the system



1)Step1:Extracting real time video.

ThevideofromtheCCTVaresubmittedasaninputinto systemorjustusingopencomputervisioncodethewe bcamcanbeusedasaninput.CapturevideousingOpe nCV(ComputerVision).When the video is captured using webcam it is stored locally but it is capturing using the CCTV it require wire to transfer the data or an cloud technology to store it virtually.

2)Step2:Video pre-processing.

Mean Image Subtraction, Remove noise in the image

Dimensionality reduction using PCA,Compresses data and Reduce Loss of Data using dimensionality reduction. In order to reduce the complexity of the process of comparing the image resolution is reduced accordingly.

3)Step3:Detectingpatterns using CNN algorithm.

Using algorithms the image is converted into matrix according to the pixels of the images and then each frame is compared in order to find the difference action which occurs.

4)Step4:Avoid redundant data.

Compare features from subsequent frames.

Avoid Redundant Data and Drop similar frames. Simply avoiding the frames which has fame pixels which mean no change occurs between this two frame.

5)Step5:Transmitting data to the Cloud for storage.

Cloud storage is the thing which helps to store the data virtually which mean no need of any hardware devices to store the data ,so the client can send their data or information to the user to retrieve it through the internet

III.ALGORITHM:ConvolutionalNeuralNetwork

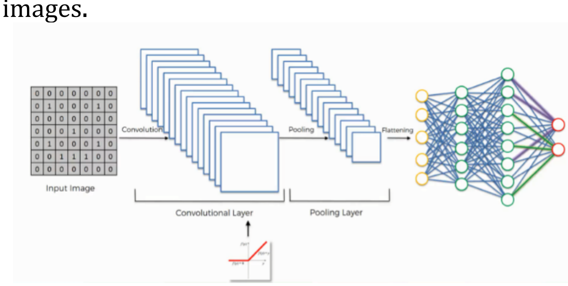
A convolutional neural layer or simply known as a ConvNet(CNN) is a one of the feature of the deep neural networks which is mostly used to carefully studying visual images. CNN uses shift invariant or space invariant artificial neural networks which have translation invariance features and shared-weights architecture which helps to use different types of multilayer forms to minimal the preprocessing.

Comparing to other algorithms to image classification CNN uses very compatibility level pre-processing, which means neural networks learns in the traditional algorithms about filter.so this independence from prior knowledge and human effort in feature design is one of the crucial cons.

CNN were derived from that biological processes with the resembles of the neutral formation of the animal or human brain which is nerves are connected with each and every other knobs which is quiet complex to learn about it.Individual cortical neurons react to upgrades just in a confined zone of the visual field known as the open field. The open fields of various neurons incompletely cover with the end goal that they spread the whole visual field.

How do CNNs work?

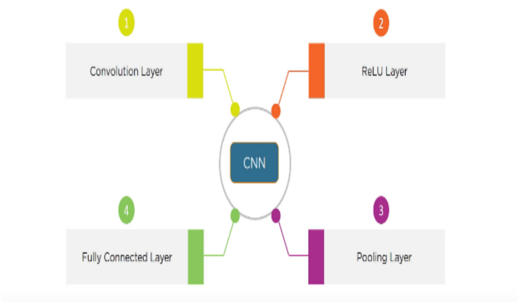
Images are made up of pixels. Each pixel is represented by a number between 0 and 255. Therefore each image has a digital representation which is how computers are able to work with images.



The procedure process as pursue:

We begin off with an information image. We apply channels or highlight maps to the picture, which provides us a convolutional layer. We then separated the linearity of that picture utilizing the rectifier function. The picture ends up prepared for the pooling step, the motivation behind which is furnishing our convolutional neural system with the personnel of "spatial invariance" which you'll see clarified in more detail in the pooling instructional exercise.

After achieved success with the process pooling, we obtained pooled highlight map. Then straighten our pooled highlight map before embeddings into aArtificial neural system. All through this whole procedure, the system's structure squares, similar to the loads and the component maps, are prepared and more than once modified all together for the system to achieve the ideal execution that will make it ready to arrange pictures and articles as precisely as would be prudent.



Step 1: Convolution-feature detector
Step 1b: ReLU(rectified linear unit) Layer-where the input image is filtered by removing the black and keeping the grey and white portion to detect the gradual motion
Step 2: Pooling
Step 3: Flattening-flattening the data from pooled feature map
Step 4: Full Connection-class recognition

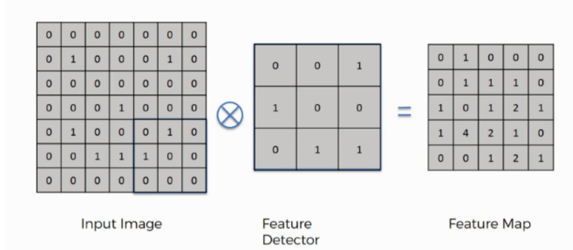
1. Convolution
A convolution is a combined integration of two functions that shows you how one function modifies the other.

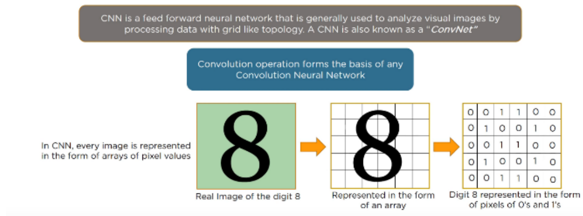
$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau$$
$$= \int_{-\infty}^{\infty} f(t-\tau)g(\tau)\,d\tau.$$

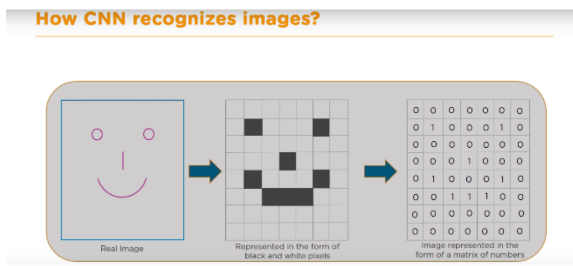
[The convolution function. Source: Wikipedia]

There are three important items to mention in this process: the input image, the feature detector, and the feature map. The input image is the image being detected. The feature detector is a matrix, usually 3x3 (it could also be 7x7). A feature detector is also known as a kernel or a filter.
Intuitively, the matrix representation of the input image is multiplied element-wise with the feature detector to produce a feature map, also known as a convolved feature or an activation map. The aim of this step is to reduce the size of the image and make processing faster and easier. Some of the features of the image are lost in this step.
However, the main features of the image that are important in image detection are retained. These features are the ones that are unique to identifying that specific object. For example each animal has unique features that enable us to identify it. The way we prevent loss of image information is by having many feature maps. Each feature map detects the location of certain features in the image.

The first functional block in our operational plan of effective process is promptly convolution operation. In the process of thiskey step, we will have to go through about feature detectors, which mainlyhelp as the neural network's ideal filters.

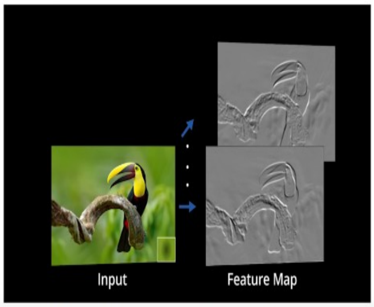






2. Apply the ReLu (Rectified Linear Unit)
In we apply the rectifier function to increase non-(being like a line) in the CNN. Images are produced of various objects that are not linear to each other. Without applying this function the image classification will be treated as a linear problem while it is a non-linear one.
Second step in the process is known as a ReLU or Fixes linear unit Which helps to increase the non-linearity in the images which is present.
Non linearity is classify into different categories we can figure it out by knowing the changes in the pixel in between, the border, the colors etc.
The rectifier serves to separate the much further to compensate for the that we may compel a picture when we place it through the convolution task.
What the rectifier work does to a picture like this all the dark components from it, continuing/bearing just those bringing over a positive esteem (the dim and white hues).
The extremely significant difference between the stable version of the image and the fixed one represent the development or increase over time/series of events or things of colors. After we

establish the image, you will discover the colors changing more suddenly. The gradual change is no longer there. That points to shows the being like a line has been thrown out in the trash.



Real image is scanned in multiple convolution and ReLU layers for locating features

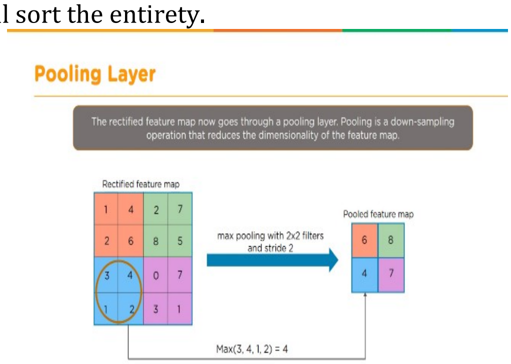
3. Pooling

Spatial invariance is a concept where the location of an object in an image doesn't affect the ability of the neural network to detect its specific features. Pooling enables the CNN to detect features in various images irrespective of the difference in lighting in the pictures and different angles of the images.

There are different types of pooling, for example, max pooling and min pooling. Max pooling works by placing a matrix of 2x2 on the feature map and picking the largest value in that box. The 2x2 matrix is moved from left to right through the entire feature map picking the largest value in each pass.

These values then form a new matrix called a pooled feature map. Max pooling works to preserve the main features while also reducing the size of the image. This helps reduce overfitting, which would occur if the CNN is given too much information, especially if that information is not relevant in classifying the image.

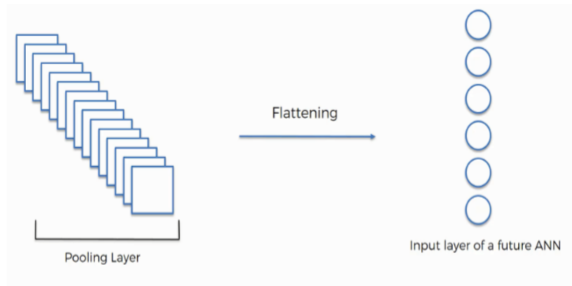
In this noticeable part, we'll appropriately cover pooling and will get the chance to perceive precisely how it for the most part functions. Our nexus here, be that as it may, will be a particular sort of pooling; max pooling. This part will instantly finish with a composed show made effectively using a visual intelligent apparatus that will sort the entirety.

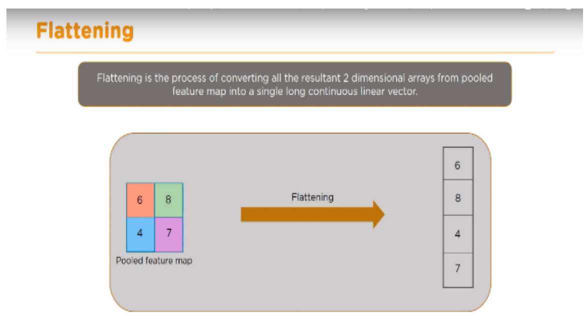


4. Flattening

Once the pooled featured map is obtained, the next step is to flatten it. Flattening involves transforming the entire pooled feature map matrix into a single column which is then fed to the neural network for processing.

In this process the process of how the system changes from pooled to flattened layer. The flattening process is done only to because in the next step we have to insert into the neural network, so flattening make the data easier to insert into the artificial neural network.



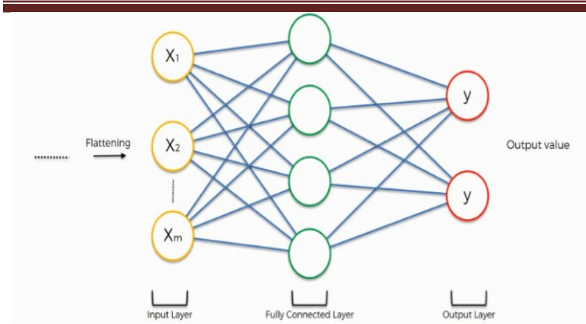


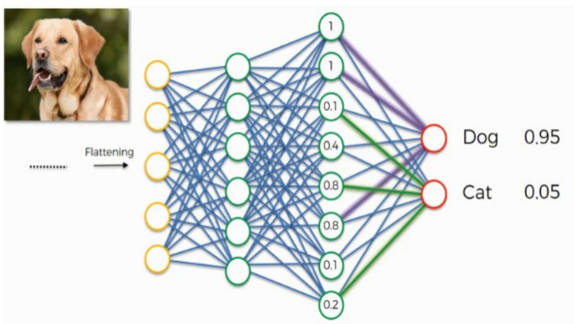
5. Full connection

After flattening, the flattened feature map is passed through a neural network. This step is made up of the input layer, the middle layer which is hidden which is connected with each of the input layer, and the final result(input) layer. The fully connected layer is similar to the hidden layer in ANNs but in this case it's fully connected. The output layer is where we get the predicted classes. The information is passed through the network and the error of prediction is calculated. The error is then backpropagated through the system to improve the prediction.

The final figures produced by the neural network don't usually add up to one. However, it is important that these figures are brought down to numbers between zero and one, which represent the probability of each class. This is the role of the Softmax function.

In the last part,all the process previously done are merged together like an neurons and finally how that neural networks done a process of classification of the images.





IV.Reference

1. S. Jeeva and M. Sivabalakrishnan, "Survey on background modeling and foreground detection for real time video surveillance," Procedia Computer Science, vol. 50, pp. 566–571, 2015, big Data, Cloud and Computing Challenges.
2. T.-Y. Wu, W.-T. Lee, and C. F. Lin,"Cloud storage performance enhancement by real-time feedback control and de-duplication,"in Proc Wireless Telecommun. Symp., Apr.2012, pp. 1–5. Video Surveillance and Storage Survey by IHS Technology. September 2014 https://www.emc.com/collateral/analyst-reports/ihs-video-surveillance-storage-intersection-it-physical-security.pdf.
3. Chowdhury, M. Mozammel Hoque, and Amina Khatun. "Image Compression Using Discrete Wavelet Transform." International Journal of Computer Science.
4. Yang, B., Zhong, J., Li, Y., & Chen, Z. (2017). Multi-focus image fusion and super-resolution with convolutional neural network. International Journal of Wavelets, Multiresolution and Information Processing, 15(04), 1750037. doi:10.1142/s0219691317500370
5. Syafeeza, A. R., Khalil-Hani, M., Liew, S. S., &Bakhteri, R. (2015). Convolutional Neural Networks with Fused Layers Applied to Face Recognition. International Journal of Computational Intelligence and Applications, 14(03), 1550014. doi:10.1142/s1469026815500145
6. Kuang, P., Ma, T., Li, F., & Chen, Z. (2018). Real-Time Pedestrian Detection Using Convolutional Neural Networks. International Journal of Pattern Recognition and Artificial Intelligence, 32(11), 1856014. doi:10.1142/s0218001418560141
7. Ergun, H., Akyuz, Y. C., Sert, M., & Liu, J. (2016). Early and Late Level Fusion of Deep Convolutional Neural Networks for Visual Concept Recognition. International Journal of Semantic Computing, 10(03), 379–397. doi:10.1142/s1793351x16400158.
8. TIVIVE, F. H. C., & BOUZERDOUM, A. (2006). APPLICATION OF SICoNNETS TO HANDWRITTEN DIGIT RECOGNITION. International Journal of Computational Intelligence and Applications, 06(01), 45–59. doi:10.1142/s1469026806001794