Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 4): pp. 3903-3910
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.04.426

Artificial Intelligence Based Computer Vision For Virtual Fencing Security System

Anuj Singh[1]*, Bhawnesh kumar, Upendra singh[1], Pankaj singh[1], Tanmay Anand[1], Dr. Namrata Prakash[2]

[1]Department of Computer Science and Engineering, Graphic Era deemed to be University, Dehradun, India.

[2]Associate Professor, School of Management, Graphic Era Hill University, Dehradun
*Email: anujsinghrawat72@gmail

Abstract— Abstract: Virtual fences are effective for monitoring people, vehicles, and their movements. Critical infrastructure sectors such as defense bases, emergency services, financial services, food and agriculture can all benefit significantly from the use of virtual fences, whether used independently or as part of a multilayered perimeter security system. Nowadays security systems are in high need, their importance has been increased in decades. Virtual walls, such as fences, offer a secure boundary without the need for a physical barrier. Critical infrastructure, such as buildings, is protected against incursion. a virtual fence that surrounds their perimeter. Virtual fencing surveillance is one of the important parts of security system for activity or intrusion detection. Computer vision is one the useful field of Artificial Intelligence. Using this tech, we would solve most of the problems including security surveillance system.

Keyword – Activity detection, Computer Vision, OpenCV, CNN Introduction.

I. Introduction

Virtual fencing security systems are gaining some hikes for activity detection in a particular area for intrusion obstruction. They are used in parking as keeping number of vehicles ingoing and outgoing.[11] Virtual fencing is being upgrading in recent years for making virtual boundaries as flexible and accurate as possible. Computer Vison is used for virtual fencing, which is one the biggest field of AI which gives a computer to able to see an object. CNN (Convolutional Neural Network) are or simple background subtraction used in detecting activity in ROI (Region of Interest).[6] Region of Interest is the region on the picture or a video which is focused upon using bounding box of variable or fixed shape and size[5].

There are many approaches for securing a particular area and time of decision is decreasing from algorithms to algorithms.[10] Virtual fencing means creating a virtual boundary over an image or a video which works as fence in it. Some uses only the virtual fencing part for securing a region which fails when a legal or authorized person also enters and some uses only the face detection part for it which is also less efficient as it may miss the area where a intrusion can get through. [2]

Two important computer vision problems are object classification and object detection. In the first scenario, the system is anticipated to correctly identify the dominant item.[8] In the second instance, it must provide precise labelling and positioning details for every object in a photo. However, for now, we'll focus on detection. There are certainly other exciting elements of computer vision, such as image segmentation.

The virtual fence is meant to serve as a boundary, barrier, or enclosure that is not reliant on tangible materials. There are many different sorts of buildings, with "no protective barrier" acting as their common denominator, depending on how it is used [9].

This paper uses contour for detecting human in specified virtually defined area and then face detection authorization using another cam. This two-level security can be very useful in places like banks, offices for restricting unauthorized person to enter the specified area.[1] This will reduce the threat of intrusion as one of the systems would be able to catch the intruder. after the detection part, AI surveillance system will alert the specified organization or person and this part will be controlled by IoT system which will work wirelessly and securely.[7] Different sensors and actuators will help tackling problems such as night vision or hidden elements using heat sensors.

II. Need of AI based virtual fencing security system

In this technological generation many of us is trying to tackle the many real-life threats and problems including theft, intrusion etc.[1] Reviewing the video after someone enters a restricted area to figure out what they took, destroyed, or left behind is insufficient. This is a proactive strategy for staying in the lead. Your mobile device's notification enables you to quickly check video of someone crossing the border illegally to see whether any circumstances necessitate taking action. You can then notify the proper authorities while the trespassing is still occurring [14].

This system can be used in army areas, banks, offices, historical places, museums for restricting unauthorized person from entering to a restricted places and alert the owner or authorized head immediately.[12] For example, an unauthorized person is entering the cabin part of the bank, the system first detects whether a person entered the cabin or not if yes then the face cam would detect whether he/she is authorized in this area or not [2].must be indented. All paragraphs must be justified, i.e. both left-justified and right-justified.

III. Architecture and Features of proposed Model

Our model detects a person in given region of interest using opencv and numpy library in python. Opencv library is dedicated towards the computer vision purpose. Used to absdiff function for tracking the movement of the target object by background subtraction.[3] As the area of interest is stationary a fixed the function would remove the stationary part and find the object which have moved in frames thus detecting its motion[4] Findcountours function of opencv is used to find the edges or[14] boundaries of same intensity or color. This is used for object detection to identify that which object is actually moving. This entire procedure is activated for only a particular region which can be redefined [5].

Now comes the human activity part, this model will detect whether the certain object is walking, jogging, sitting or standing. This model is being used for keep a track of authorized and unauthorized person.[6] If a person is running when entered the region it may be unauthorized person which get into the prohibited region.[7] For Activity recognition Neural network of 240 input vector elements are used as 80 pieces for 3 accelerometers fig1(x, y, z). These accelerometers decide with x, y, z parameters whether the person I sitting, standing, running, walking, etc.

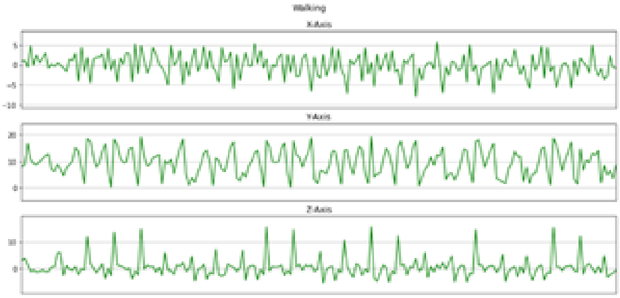
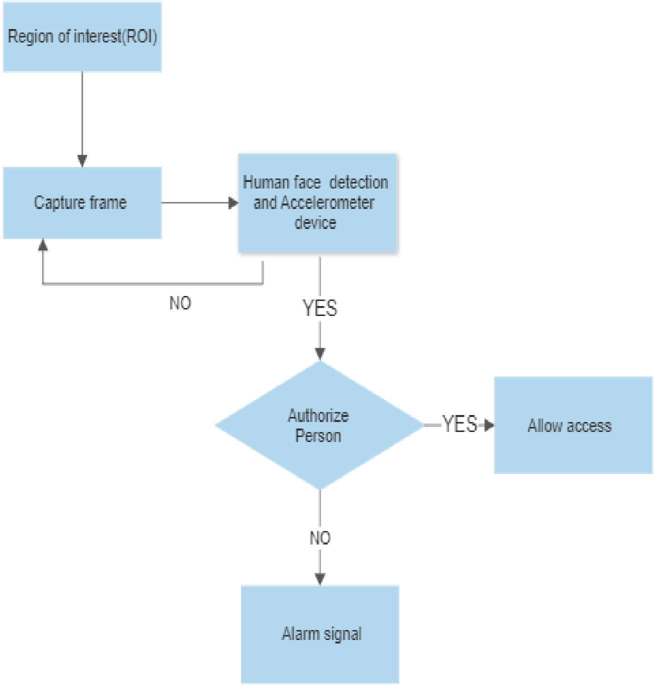


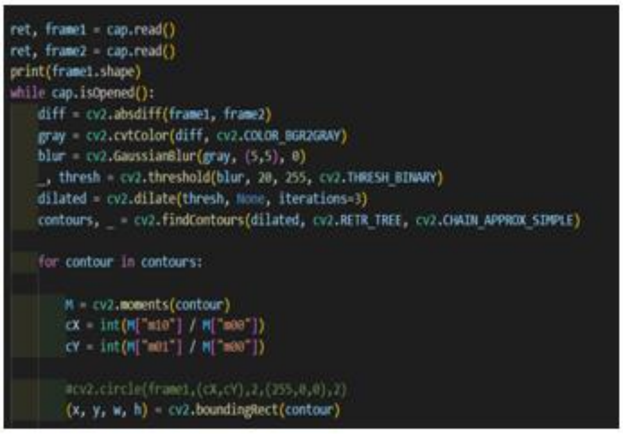
Fig1: walking accelerometer

after activity and virtual fencing the face detection part comes. Face detection model uses face_recognition and opencv library for the identification for authorized face provided to the model.[6] If the face is not trained for a particular person the model will declare it as unauthorized.

And this two stage AI security detection models will work simultaneously and will activate only when the human detection model will detect an entrance in the region of interest. [7]

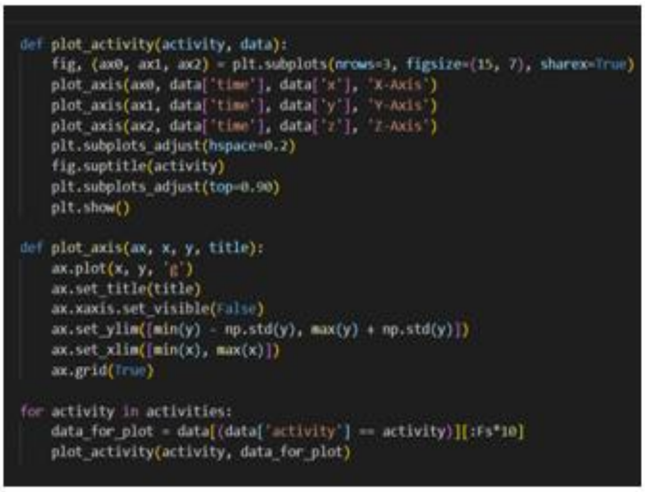


Operational flow chart of proposed model



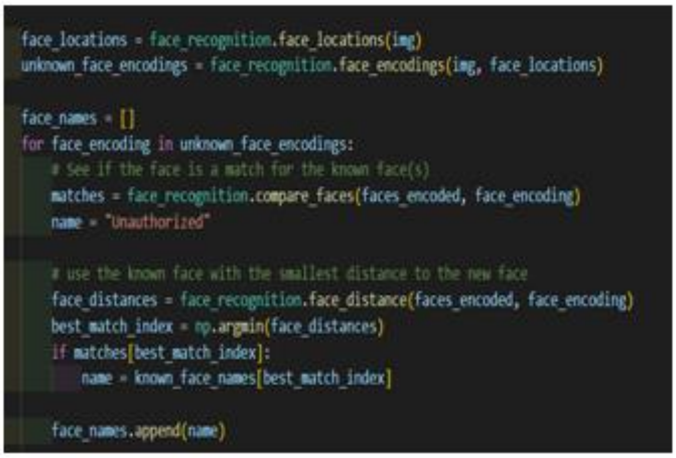
```
ret, frame1 = cap.read()
ret, frame2 = cap.read()
print(frame1.shape)
while cap.isOpened():
    diff = cv2.absdiff(frame1, frame2)
    gray = cv2.cvtColor(diff, cv2.COLOR_BGR2GRAY)
    blur = cv2.GaussianBlur(gray, (5,5), 0)
    _, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH_BINARY)
    dilated = cv2.dilate(thresh, None, iterations=3)
    contours, _ = cv2.findContours(dilated, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

    for contour in contours:

        M = cv2.moments(contour)
        cX = int(M["m10"] / M["m00"])
        cY = int(M["m01"] / M["m00"])

        #cv2.circle(frame1,(cX,cY),2,(255,0,0),2)
        (x, y, w, h) = cv2.boundingRect(contour)
```

1. Code snippet of human detection



```
def plot_activity(activity, data):
    fig, (ax0, ax1, ax2) = plt.subplots(nrows=3, figsize=(15, 7), sharex=True)
    plot_axis(ax0, data['time'], data['x'], 'X-Axis')
    plot_axis(ax1, data['time'], data['y'], 'Y-Axis')
    plot_axis(ax2, data['time'], data['z'], 'Z-Axis')
    plt.subplots_adjust(hspace=0.2)
    fig.suptitle(activity)
    plt.subplots_adjust(top=0.90)
    plt.show()

def plot_axis(ax, x, y, title):
    ax.plot(x, y, 'g')
    ax.set_title(title)
    ax.xaxis.set_visible(False)
    ax.set_ylim([min(y) - np.std(y), max(y) + np.std(y)])
    ax.set_xlim([min(x), max(x)])
    ax.grid(True)

for activity in activities:
    data_for_plot = data[(data['activity'] == activity)][:Fs*10]
    plot_activity(activity, data_for_plot)
```

2. Code snippet of activity detection

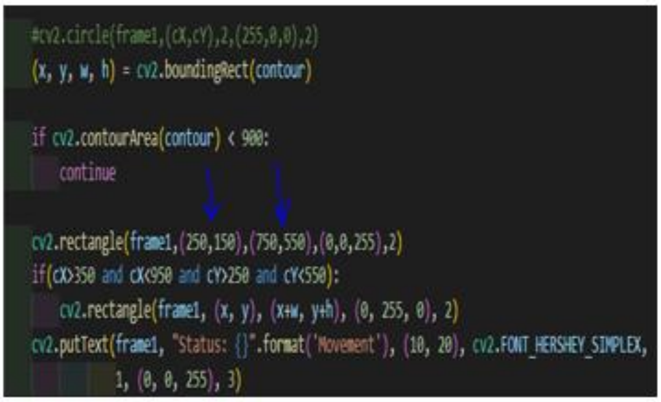


```
face_locations = face_recognition.face_locations(img)
unknown_face_encodings = face_recognition.face_encodings(img, face_locations)

face_names = []
for face_encoding in unknown_face_encodings:
    # See if the face is a match for the known face(s)
    matches = face_recognition.compare_faces(faces_encoded, face_encoding)
    name = "Unauthorized"

    # use the known face with the smallest distance to the new face
    face_distances = face_recognition.face_distance(faces_encoded, face_encoding)
    best_match_index = np.argmin(face_distances)
    if matches[best_match_index]:
        name = known_face_names[best_match_index]

    face_names.append(name)
```

3. Code snippet of face detection

IV. Implementation and Result analysis

The area of a video that you wish to filter or otherwise alter is called the region of interest (ROI)[9]. A ROI can be shown as a picture of a binary mask. The mask image assigns the value 1 to pixels inside the ROI and the value 0 to pixels outside the ROI. The toolkit provides a number of methods for defining ROIs and making binary masks can have variety of objects, including circles, ellipses, polygons, rectangles, and hand-drawn forms, are available in the toolbox for building ROIs in different shapes.[8] The things' form, location, look, and behavior can all be changed once they are formed[3].



```
#cv2.circle(frame1,(cX,cY),2,(255,0,0),2)
(x, y, w, h) = cv2.boundingRect(contour)

if cv2.contourArea(contour) < 900:
    continue

cv2.rectangle(frame1,(250,150),(750,550),(0,0,255),2)
if(cX>350 and cX<950 and cY>250 and cY<550):
    cv2.rectangle(frame1, (x, y), (x+w, y+h), (0, 255, 0), 2)
cv2.putText(frame1, "Status: {}".format('Movement'), (10, 20), cv2.FONT_HERSHEY_SIMPLEX,
            1, (0, 0, 255), 3)
```

4. Code snipped of ROI

Select a rectangular bounding box or region of interest in OpenCV (ROI). Previously, we had to handle mouse operations and define our own bounding box.[10] We may now use selectROI, a native OpenCV function, instead. The Region of Interest feature is used in several key picture functions in OpenCV.[8] Using a ROI, we may perform operations on a rectangular part of the picture. The standard phases in the ROI process include creating a ROI on an image, performing the needed action on this region of the picture, and lastly resetting the ROI.

During object detection in region of interest we can modify the shape of virtual fence by interpreting its four points marked in the above image.[14] This will make the model to detect object inside the given boundary.

In the below Fig 2, we have provided a rectangle of 750 x 350 and the model is detecting the people which are in the rectangular fence provided by us.[10]

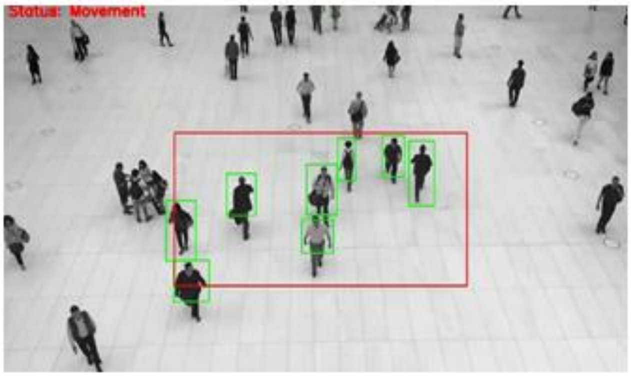


Fig. 2

Subject image ‘A’

But if we try any other which we have not provided, let’s take subject image ‘B’ it will declare it as unauthorized.[4]

Subject image ‘B’

The results are very accurate and are successfully proceeded.[7] The two staged security model have successfully detected the people inside the virtual fence and the human activity model is keeping track of the object inside it and the face detection modal have successfully differentiated between a authorized or unauthorized person during the training and testing part.

V. Conclusion

In this paper we showed how our model worked successfully and gave the best results. Using three different models we have created a two staged AI based virtual fenced security system. The quality of video may vary results but will not affect that much. In future more advanced activity or object detection modal will come and could be used to for more accurate results.

REFERENCES

[1] Christina Umstatter, “The evolution of virtual fences: A review,” Computers and Electronics in Agriculture, vol. 75, pp. 10–22, 2011

[2] Fabrice RECLUS and Kristen DROUARD, “Geofencing for fleet & freight management,” in Proceedings of the 9th IEEE Intelligent Transport Systems Telecommunications Conference (ITST 2009), 2009, pp. 353–356

[3] Chris Stauffer and W. E. L. Grimson, “Learning patterns of activity using real-time tracking,” IEEE Transactions on pattern analysis and machine intelligence, vol. 22, pp. 747–757, 2000.

[4] Anuj Singh, Sachin Sharma, Kamlesh Chandra Purohit, and KC Nithin Kumar. "Artificial Intelligence based Framework for Effective Performance of Traffic Light Control System." In 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), pp. 1-4. IEEE, 2021.

[5] Anuj Singh, Tanmay Anand, Sachin Sharma, and Pankaj Singh. "IoT Based Weapons Detection System for Surveillance and Security Using YOLOV4." In 2021 6th International Conference on Communication and Electronics Systems (ICCES), pp. 488-493. IEEE, 2021.

[6] Anderson, D.M., Rus, D., Detweiler, C., Schwager, M., Vasilescu, I. Husbandry of free-ranging cows using virtual fencing concepts. Proceedings Multifunctional Grasslands in a Changing World. XXI International Grassland Congress, VIII International Rangeland Congress, Hohhot, China, June 30–July 5, p. 660.

[7] Anderson, D.M., Rus, D., 2009. Gathering cows using virtual fencing methodologies [abstract]. 62nd Society for Range Management Annual Meeting. Paper No. 2030-4.

[8] Butler, Z., Corke, P., Peterson, R., Rus, D. “Virtual fences for controlling cows”. Proceedings of the IEEE International Conference on Robotics and Automation, IEEE Computer Society Press, May 2004

[9] J. Niu, J. Lu, M. Xu, P. Lv, and X. Zhao, “Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting,” Pattern Recognition, vol. 59, pp. 225–233, 2015.

[10] J. Son, H. Yoo, S. Kim, and K. Sohn, “Real-time illumination invariant lane detection for lane departure warning system,” Expert Systems with Applications, vol. 42, no. 4, pp. 1816–1824, 2015.

[11] A. Mammeri, A. Boukerche, and Z. Tang, “A real-time lane marking localization, tracking and communication system,” Computer Communications, vol. 73, pp. 132–143, 2016.

[12] Sedlmair M, Meyer M, Munzner T (2012) Design study methodology: reflections from the trenches and the stacks. IEEE Trans Visual Comput Graph 18(12):2431–2440

[13] Stein M, Janetzko H, Lamprecht A, Breitkreutz T, Zimmermann P, Goldlucke B, Schreck T, Andrienko G, Grossniklaus M, Keim DA (2018) Bring it to the pitch: combining video and movement data to enhance team sport analysis. IEEE Trans Visual Comput Graph 24(1):13–22

[14] Tarrago´ R, Iglesias X, Lapresa D, Anguera MT (2016) A complementary study of elite fencing tactics using lag sequential, polar coordinate, and t-pattern analyses. In: Proceedings of the international conference on sequence analysis and related methods, pp 339–348 Vuillemot R, Perin C (2016) Sports tournament predictions using direct manipulation. IEEE Comput Graph Appl 36(5):62–71