Article **Journal Computer Technology**
June 2023, Vol.7 No.18 8-22

Artificial vision techniques at the frontiers of video surveillance

Técnicas de visión artificial en fronteras de la video vigilancia

PÉREZ-ESCAMILLA, Javier´*†, MENDOZA-GUZMÁN, Lorena, CRUZ-GUERRERO, René and PORRAS-MUÑOZ, Rolando

Tecnológico Nacional de México / ITS del Oriente del Estado de Hidalgo. Carretera Apan-Tepeapulco Km 3.5, Colonia Las Peñitas, C.P. 43900, Apan Hidalgo, México.
´ Tecnológico Nacional de México / ITS del Occidente del Estado de Hidalgo. Paseo del Agrarismo 2000, Carretera Mixquiahuala - Tula, Km. 2.5. Mixquiahuala de Juárez, Hidalgo, C.P.42700.

ID 1st Author: *Javier, Pérez-Escamilla* / **ORC ID:** 0009-0008-4090-2259, **CVU CONAHCYT ID:** 939609

ID 1st Co-author: *Lorena, Mendoza-Guzmán* / **ORC ID:** 0009-0005-7802-6352, **CVU CONAHCYT ID:** 1289555

ID 2nd Co-author: *René, Cruz-Guerrero* / **ORC ID:** 0000-0003-1276-2419, **CVU CONAHCYT ID:** 551299

ID 3rd Co-author: *Rolando, Porras-Muñoz* / **ORC ID:** 0009-0006-1065-9695, **CVU CONAHCYT ID:** 1289586

DOI: 10.35429/JCT.2023.18.7.8.22 Received: January 15, 2023; Accepted June 30, 2023

Abstract

The present work addresses the task of identifying a predatory behavior of robbery of homes or businesses. The proposed objective is the detection of blunt elements used in the commission of the crime, limiting the context to barrettes, covered faces, people and gates (doors or windows). The proposal addresses the task of object identification applying Single Shot Detectors (SSD). Due to its versatility and the physical resources applied, the structure of SSD ResNet50 V1 FPN 640x640 has been chosen from the TensorFlow Model Zoo to train and validate the classification. This has been built in five classes, for the training and validation set, an average of 50 annotations per class have been processed. Additionally, a support function was worked on in the detection of human activity. The evaluated model obtained a mA of 69% in the detection of objects and in the identification of criminal behavior it showed a performance of 69%.

Predatory behaviour, Attention mechanisms, Deep learning

Resumen

La investigación, aborda la tarea de la identificación de una conducta predatoria de robo a casa habitación o negocio. El objetivo planteado es la detección de elementos contundentes usados en la comisión del delito, limitando el contexto a barretas, rostros cubiertos, personas y cancelería (puerta o ventana). La propuesta, aborda la tarea de identificación de objetos aplicando Single Shot Detectors (SSD). Por la versatilidad y los recursos físicos aplicados se ha optado por la estructura de SSD ResNet50 V1 FPN 640x640, del Zoológico de modelos de TensorFlow para entrenar y validar la clasificación. Éste se ha construido en cinco clases, para el conjunto de entrenamiento y validación se han procesado en promedio 50 anotaciones por clase. Adicionalmente, se trabajó una función de apoyo en la detección de la actividad humana. El modelo evaluado obtuvo una mAP de 69% de precisión en la detección de objetos y en la identificación de la conducta delictiva mostró un desempeño del 69%.

Conducta predatoria, Mecanismos de atención, Aprendizaje profundo

Citation: PÉREZ-ESCAMILLA, Javier, MENDOZA-GUZMÁN, Lorena, CRUZ-GUERRERO, René and PORRAS-MUÑOZ, Rolando. Artificial vision techniques at the frontiers of video surveillance. Journal Computer Technology. 2023. 7-18:8-22.

* Correspondence to Author (E-mail: javierperez@itsoeh.edu.mx)
† Researcher contributing as first author.

Introduction

Artificial intelligence (AI) has developed human capabilities. "Enhanced artificial intelligence" is the new frontier in the implementation of smart devices, which in turn opens up new possibilities for solving problems associated with security and video surveillance [1]. Various artificial intelligence techniques have resurfaced with great interest, one example being Deep Neural Networks (DNN), which perform well for identifying faces, vehicles, weapons, and other objects associated with a security system [2]. The preparation of the inputs, as well as their pre-processing, represent the challenge of correct selection for the good performance of a machine learning model [3].

Object detection has gained emphasis in AI topics, where neural networks have made the greatest contribution [4]. In this case, the process requires collecting diversity in the training data, assuming a relative position of the object, scanning the input or scene, performing a classification to determine the class of the object, and obtaining the relative position of the object [5]. Various techniques are applied to the position of the object, taking into account the characteristics of the model, e.g, [6].

Computer vision (CV) is presented as a mature and extensive element, easy to integrate into projects where image processing is required. CV is the part of AI behind the comprehension and manipulation of videos and images [7]. Multiple computer vision tools are incorporated as a solution system for a specific problem [8]. The integration of this into technology is all the rage, through face recognition for the security of cell phones, banking transactions, and autonomous, land and air vehicles, just to name a few examples. Object detection is no exception.

Object localization differs from classification, while the task of classifying is to determine a label for an object, localizing allows finding the position of the object within an image. Multiple objects can be located, for this action, the location algorithms specialize in making cuts to the image to determine areas of interest or regions of interest (ROI). In general, DNN models have been shown to perform better [9].

The model used in this research focuses on the detection of movements similar to criminal behavior, limiting the context to people who violate a door or window with a blunt object such as a crowbar and whose faces are covered. So then, we know the elements that make up the scene to recognize human action. The descriptions and references given in [10] have been used to describe the behavior. Using a labeling tool, annotations have been made for each image that correspond to the proposed classes.

Previous works

Hu et al. (2015), ask the question Why do Convolutional Neural Networks (CNN) work so well? They propose a set of deep neural network architectures, oriented to face recognition, using a set of labeled photos called LFW or Labeled Face in the Wild. It builds models based on the deep neural network AlexNet [11], where it applies combinations of three convolution layers, a dense layer and an output dimensionality reduction function (softmax). Something that stands out is that it makes 30 cuts to the LFW instances for training and concludes that the differentiation between the different models is supported by the scale and the regions of the input to be reviewed [12].

Nam et al. (2018) proposed a scale-invariant pyramid model, considering that a better input can improve the performance of a CNN. They study the characteristics of different architectures, such as DeepFace, DeepID, FaceNet, and the VGG. They develop a set of patterns, under the assumption that some features are not evaluated by the convolutional network, limiting its performance. In addition, they consider that in video surveillance, the entrance has variations in pose, lighting, expression, and image quality. They use a public pool called LFW and a database called CCTV. The implementation of mechanisms that use characteristics not considered before, improves the performance of the RED vs. a CNN VGG, having better results when the input is 200x200 pixels [13].

ISSN 2531-2197

Wong et al. (2017), proposed an object tracking system based on the premise of simple labeling, placing classification in a second stage. Thus, it streamlines the crawl process and shows that it improves crawl performance when combined with the algorithm's object state information. They face the challenge of detection and classification from an online video by applying the multiple object tracking algorithm, MOT for its acronym in English Multiple Object Tracking. The proposed model is based on: identifying objects through SEF algorithms, (Shape Estimating Finder); track features with a combination of CACTuS, for Competitive Attentional Correlation Tracker using Shape, and FL, for Feature Learning. In a second stage, it applies the classification of the objects, so it concludes its work with great precision [14].

Padmaja, Myneni and Patro, (2020) raised the possible applications of the recognition of multiple activities and multiple objects, MAMO, Multi Activity-multi object recognition. They recognize the challenges in public areas such as hospitals and universities, early warning, tracking, and intelligent video surveillance, as well as the integration of existing components and algorithms such as YOLO and alike. They describe the different components and collections of images for training, as well as the use of metrics based on classification. It highlights algorithms like DeepLandmark and the effects of stock rankings. In addition, it provides a comparison of the methods and goals to be achieved in new developments [15].

Lu et al. (2017), addressed the object position prediction task. They design a system of LSTM, Long Short Term Memory. They propose a tensor based on the position of the detection and a characteristic matrix of the object. They solve the regression and association error computation problems in the trace. It contributes substantially by reversing the direction of reproduction of a video. The model is based on a recurrent network with a normalization layer, where using SSD it identifies and tracks the element, using object detection, object classification, and object position for feature matrix extraction. They conclude with good precision in the approximation of the trajectory of the object [16].

Sai, Sasikala (2019) implemented an application for object detection and counting using CV and SSD tools based on Fast R-CNN, from the Fast Regional Convolutional Neural Network. They focus their efforts on input pre-processing, with 70 instances of the “gun” and “knife” classes. By making adjustments to the model, they limit the number of bounding boxes. They conclude with a great ability to identify and count multiple objects [17].

Materials and methods

In the development, various concepts and techniques of computer science are applied, here, the most relevant are mentioned to provide a general context of the work.

Machine learning (ML)

Machine learning (ML) is the science and art where it is possible for the machine to simulate the learning process. Through a study of the problem, the associated data is modeled, supported by mathematical and/or statistical tools. Then, it is required to program an algorithm that helps the trial-and-error process, so the task will be automated. As a result, the proposed model is capable of adapting to changes. For problems that have a solution, a set of adaptations or rules may be required, and the performance may change depending on the algorithms applied. An additional advantage is that for complex, solvable problems, ML techniques will be able to find a solution. Lastly, for problems that fluctuate, ML algorithms can be adapted. Sometimes ML systems can provide relevant information on the problem at hand. ML can be divided into 2 categories of algorithms: supervised ones, such as linear regression and classification; unsupervised ones such as clustering, association rules, visualization and dimensionality reduction [18].

Deep Neural Networks

Marvin Minzky coined the term Perceptrons, which describes the behavior of a neuron and defines it as an inference engine that identifies the characteristics of an object, in order to classify it within a defined hierarchy [19].

ISSN 2531-2197

Artificial Neural Networks (ANN). They respond to an imitation of human parallel processing capacity, considering the fundamental difference that machines are sequential, these networks acquire knowledge based on experience, are adaptable, fault tolerant, and have nonlinear behavior [20].

In the context of ML, deep learning is about the variants and ways of connecting the ANNs; it takes as its inspiration the structures and connections of the human brain, thus elements called neurons perform specific tasks with the data. Here, the connections are dozens of networks and millions of neurons. The importance of knowing the types of layers and operations of a neural network enables researchers to propose models for specific tasks, so we can list the components in a deep network [21]:

1) Convolutional layer

This layer operates on the model input, performing the calculation of convolution operations. A convolution is an array sweep operation by a filter or kernel, resulting in a new array.

$$y_{rc}^{l} = \sum_{i=1}^{Fr} \sum_{j=1}^{Fc} y_{(r+1-1)(c+j-1)}^{l-1} w_{ij}^{l} + b^{l} \quad (1)$$

The output y_{rc}^{l} de $\{r, c\}$, is a function of Fr, Fc, which is the number of columns and rows in the pass filter. w_{ij}^{l} is the value of the filter at position {i,j}. $y_{(r+1-1)(c+j-1)}^{l-1}$ represents the value of the input to this layer at position {r+i-1,c+j-1}. b^{l} is bias value.

2) Dense layer

This layer represents classical neurons in perceptrons. Its main function is linear regression or classification.

3) Activation function

Activation functions bring the non-linearity of the functions and add multi-layer activations at the output of the network. Softmax and RELU are of interest.

$$Softmax = f_j(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (2)$$

$$RELU = f(x) = max(0 + z_i) \quad (3)$$

4) Pooling layer

This layer is used to reduce the output volume of the convolutional layers. It allows the perceptron to increase the field of perception of the network, thus adding viable features to improve the classification capacity of the model.

$$h_{xy}^{l} = max_{i=0..s, j=0...s} h^{l-1}(x+i)(y+j) \quad (4)$$

5) Dropout layer

It is the layer that allows you to avoid overtraining. It eliminates the contributions of some neurons next to the input and output connections, based on a random probability.

6) Batch Normalization

This layer increases the stability of the training by normalizing the outputs of the previous activation layers. It applies the operations by subtracting the provided mean and dividing it by the standard deviation.

$$y = \frac{\lambda}{\sqrt{Var[x]+\epsilon}} x + (\beta - \frac{\lambda E[x]}{\sqrt{Var[x]+\epsilon}}) \quad (5)$$

Where, multiple mini batches β, sized m with mean and variance are processed and averaged among them in order to obtain the general mean and variance. The network is assumed to have a set of trainable parameters that is the input to the execution algorithm $\Theta = \{x^{(k)}\}\ _{k=1}^{K}$. So, by batch, the parameters of the activation functions are updated$\Theta \cup \{\gamma^{(k)}, \beta^{(k)}\}_{k=1}^{K}$.

7) Epoch

An epoch is a training cycle of the neural network where all the model operations are performed. The "weights" of the activation functions that offer the greatest gain to the model are stored here.

Object detectors

CV allows addressing unsupervised learning problems, this recent approach is becoming known as unsupervised visual learning. Here, elements of the real world with dimensions of space and time become relevant.

ISSN 2531-2197

Objects are local outliers in the global scene, small in size and with a different appearance and motion than their larger background. It is then when the pixels and frames become relevant, but they have a difference in dimensionality. Algorithms such as VisionPCA [22] converge on quickly identifying the differentiation between the background and the objects in the frame. Thus, one can choose with high precision (not necessarily high recall) data samples that belong to a single object or category. Multiple techniques are viable, such as graphs, segmentation, and deep learning [23].

Object detectors apply a set of techniques that take advantage of the capabilities of DNNs, where convolutional operations are capable of mapping object classes. The use of bounding boxes (BB) is incorporated, by means of a hypothesis of the position of the object, a box of its location is drawn. In contrast, SSDs are a set of techniques and tools that may be larger than deep networks, but they do all the tasks in one fell swoop. In Figure 1 we show examples of R-CNN, for Regional CNN, and YOLO, for You Only Look Once [24]. A comparison chart is illustrated in Figure 1.

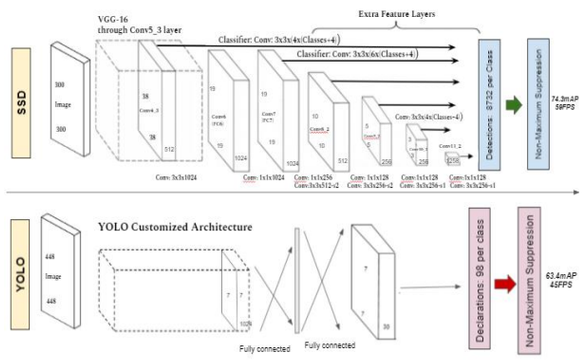

Figure 1 Comparison of activation diagrams. While SSD applies convolution operations seeking to perform object detection using "Multi-scale feature maps for detection" and then applies a classification process, YOLO, on the other hand, uses autoencoders.

In an input where multiple classes are available, the strategy of sub-mapping the input is advantageous, so that there is no longer just one ROI but several samples to verify. This sweep process is done using a regression strategy.

Fast R-CNN

An improvement in the processing of SSDs implies that, by having multiple object instances in one input, classification can be done with limited physical resources. We show an abstraction of the model in Figure 2. For the Fast R-CNN model, shared compute and memory resources are used. So, then it improves detection applying a sub-sampling of the h x w input by a network of H x W samples. Each input is reduced with max-pooling. In the model, the input is processed using convolution and activation functions with softmax. BB are worked on with regression. A loss function is applied in the context, previously, annotations of the ROIs have been made for the classification. A re-example thread is executed in the background, where Stochastic Gradient Descent (SGC) is used as the optimization function [25].

1) Classification function:

$$L(p,u,t^u,v) = L_{cls}(p,u) + [u \geq 1]L_{loc}(t^u,v) \quad (6)$$

The location and classification are associated for each ROI, where $L_{cls}(p,u) = logp_u$ is the logarithmic loss for the true class u.

2) Loss function:

$$L_{cls}(t^u,v) = \sum_{i\in\{x,y,w,h\}} smooth_{L1}(t_i^u - v_i) \quad (7)$$

Where:

$$smooth_{L1}(x) = \{0.5x^2 if|x| < 1|x| - 0.5\ otherwise \quad (8)$$

The calculation of the partial derivative of loss for each example (ROI).

$$\frac{dL}{dx_i} = \sum_r \sum_j [i = i^*(r,j)]\frac{dL}{dy_{rj}} \quad (9)$$

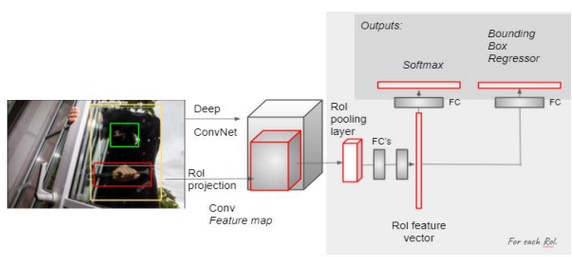

Figure 2 Fast R-CNN activation diagram. A set of mapping functions, a re-instance of the input, and activation functions for classification and regression for localization are denoted.

ISSN 2531-2197

RetinaNet

RetinaNet is a neural network model for object detection, where the use of the identity function becomes relevant. This function permeates the idea that a small neural network performs well and that it should be much easier to learn the "activation weights". So, it is a matter of passing the input directly to the output of the intermediate layers. The network adds the input to the output; this is called the residual block. This process consists of two paths: the first is a series of regular neuronal layers, or main branch; the second is a direct path from input to output, or shortcut branch. Batch normalization and RELU are operations applied to the main branch. [26] In Figure 3, the architecture is illustrated.

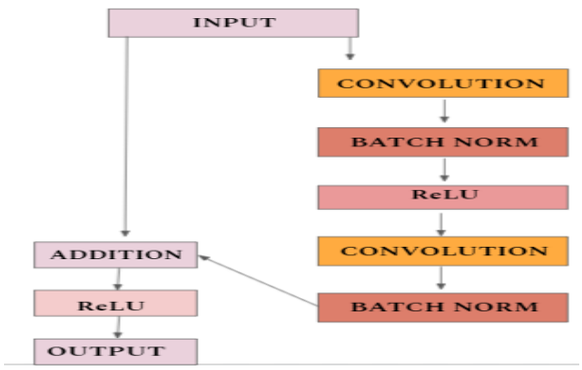

Figure 3 Residual network block. In the main branch, it consists of a convolution layer, normalization, linear rectification, convolution and normalization layer. In the secondary branch, the entry is added directly to the main branch. Finally, a linear rectification is performed and the output of the block is computed.

Metrics

The metrics of an ML model are basically calculated based on correctly classified instances, TP or True Positives, and FP or False Positives; and not correctly classified, TN or True Negatives, and FN or false negatives. The associated functions are: Accuracy, which measures the correctness of the model; Recall, which measures the model's ability to discriminate between classes; and Precision, which measures the model's ability to correctly recognize new instances between classes [27].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

$$Precision = \frac{TP}{TP+FP} \quad (12)$$

In the case of object detectors, the Mean Average Precision metric, mAP, is regularly included. The mAP value is useful for assessing location and segmentation model performances as well as classification. With this metric, the classification and localization of the model are assessed. For this, we need to know the annotation of the location and class called fundamental truth or Ground Truth. This is how a BB and a class label are denoted. Since we are checking a delimiter, the box, a metric is required that indicates exactly how the model has created that delimiter of the object. Here, the calculation of intersection over union (IoU) is the relation between the delimiter given by the model of the intersection with Ground Truth, on the union and the Ground Truth [28].

$$IoU = \frac{A \cap B}{A \cup B} \quad (13)$$

$$mAP = \frac{1}{n}\sum_{K=1}^{k=n} AP_k \quad (14)$$

Where n = number of class and $AP_k = Average\ Precision\ of\ class\ K$.

LabelImg

In the annotation corresponding to the Ground Truth values, the LabelImg tool is used. It was built by Massachusetts Institute of Technology (MIT), using the Python language and a plugin called 'Qt' for graphical controls. In it, annotations can be made in XML format, for YOLO format and for the set of visual objects called PASCAL Dataset. It saves the positions of the BB and the class to which each object in an image corresponds to. [29].

For the labeling of the classes, those used in the COCO (Common Context) set are taken as a reference, where the elements represented are found in their natural environment. Differences in labeling vary between different data sets; for example, in VOC, Visual Object Class, similarly, they mainly store the location and class of the object, either by segmentation or location of bounding boxes [30].

ISSN 2531-2197

Transfer Learning

Given the large amount of time required to train a neural network, the task of implementing solutions to specific problems is addressed by a knowledge transfer technique, Transfer Learning (TL). Here the benchmark in performance and relevance of the application domain plays an important role. A review of the models helps to determine those susceptible to apply and compare [31].

Color spaces and moments

In the intensity change analysis of pixels in an image, it allows to obtain crucial information about the objects in an image. In this case, the HSV color spaces, Hue Saturation and Value, highlight characteristics that are not denoted with the naked eye, converting an RBG color space to a more enriched one. This is how new associated colors appear, such as yellow, magenta, and cyan. This technique can help tracking objects based on pixel intensity. [32].

Deep learning in images is based on the composition of multiple nonlinear transformations. Data is processed into abstract semantic representations formed on the network (end-to-end paradigm). These representations have great flexibility and adaptation, where the quality of the representation is affected by the integrity of the training data; the robustness is affected by the computational cost; the robustness of the geometric transformation is limited; therefore, it requires the augmentation of data; geometric invariance is affected by time/space complexity. In a pre-convolutional neural network, aspects of frequency transformation, Texture, dimensionality, Moments and Moment invariants, discriminability, and robustness are considered [33].

Methodology to develop

The waterfall methodology has the characteristic of delivering the product until the end of the development cycle, where a set of deliverables is generated in a limited space of time and where it can be iterated [34].

The application development cycle consists of six steps to achieve implementation. The complete cycle is illustrated in Figure 4. The first stage consists of searching for images associated with the problem. The second stage consists of tagging the objects that have been found in the application. The third stage consists of generating the training and test sets. The training stage consists of carrying out a set of adjustments to the model so that it can adjust the activation weights of the functions. The metrics preparation stage consists of activating the TensorBoard features so that it shows the performance graphs. The last stage consists of testing the model against a validation set.

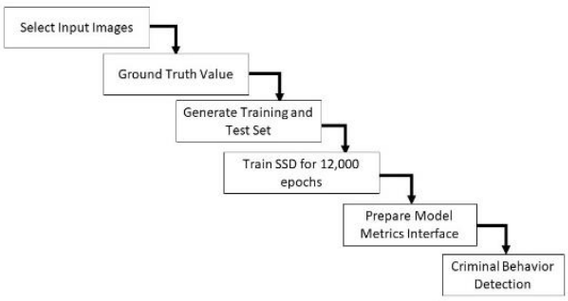

Figure 4 *Stages of work.* The parts of the process are illustrated. The last task verifies the functionality of the model,

1) Image selection

56 images were selected from the internet, they were sought to be in JPG format, 24 bits of color depth and 94 dpi of horizontal and vertical resolution. In addition, they will have some of the elements used in the breakage of the door scene. The images are in their natural state, that is, in the real context in which they occur. Note: All those images are used for academic and research purposes.

2) Ground Truth Value

Using the LabelImg tool, 56 XML (Extensive Markup Language) encoded files of bounding box truth value annotations were obtained. This work was done manually and sought to obtain an average of 56 annotations per class.

ISSN 2531-2197

3) Training and testing set

The training set and the test set were separated into 42 and 14 images, respectively. Compiled files called Tf Records are created, which are used with the ML software TensorFlow and Keras.

4) Train SSD at 12,000 epochs

Using Transfer Learning and the DNN model called ResNet 50 V1 FPN 640x640 SSD, the adaptation for the purpose of the project is made. The process begins by declaring the number of classes, the location of the training and test sets, the number of epochs, and the weights of the COCO dataset saved. The latter is a pre-trained model that is used to transfer the activation weights of the model. In this case, it was compiled using a computer with a Nvidia 730 video card, Ryzen 5 CPU, 16GB RAM, 128GB SSD.

5) Prepare model metrics interface

Tensor Board includes a set of visual tools for reviewing model metrics. In this case, a local service is enabled with output to port 6006. Thus, it is possible to access the graphs of the object detection metrics in Figure 5, where the output of the model is illustrated.

Figure 5 Tensorboard. The partial training results are shown in training and validation metrics, in this case at 8000 epochs.

6) Prepare model metrics interface

For behavior detection, a helper function is used. The validation of the condition that activates the scene is based on an If-Then-Else criterion to determine if the classes are present in the image. The bounding box labels displayed in the output are referenced.

Results and discussion

The shortage of images of criminal behavior to be treated limit their quality. In many cases, they are of low resolution and relatively small size. For the purpose of this research, a minimum standard of 96 dots per inch horizontally and vertically, 24 bits depth and in JPG format, from the English Joint Photographic Experts Group, were processed.

The dimensions of the scenes are varied, ranging from 148 x 341 and up to 808 x 818 in height and width, with a maximum width of 1200 pixels. Given the variety of sizes, the model performs a scaling to a fixed size of 640 x 640 pixels, in addition to using image enlargement, and horizontal rotation.

Figure 6 *Training Set*. The images with the required classes are highlighted. If there is difficulty finding the required number of images, the context of use is limited.

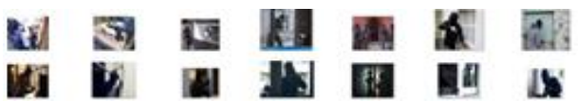

Figure 7 *Test set.* The number of images is reduced, but the annotations are approximately 14 per class.

Figures 6 and 7 show the proposed training and test sets. The images were obtained from the web, and their purpose is educational. The selection is due to the fact that objects are present in human action. The crowbar is the critical element to identify human activity, but the context of its use is also diverse, what is relevant is the human action that takes place, as shown in Figure 8.

ISSN 2531-2197

Figure 8 Image of the training set. In the illustration, there are the objects to be detected: the crowbar, covered face, person and the door. The action is to open a door with a crowbar.

Using LabelImg, annotations were made on the images. The system for labeling in the test set is illustrated in Figure 9. Figure 10 shows the labeling of an image for the training set.

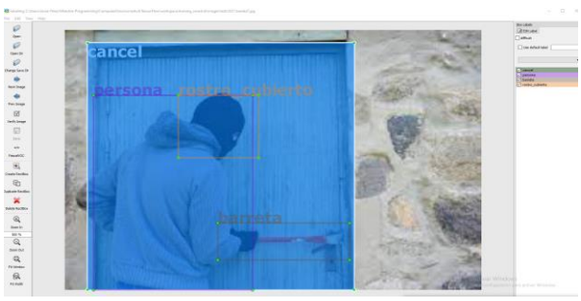

Figure 9 Use of LabelImg. In the test set. Object annotations were performed on the test set. In this image of the test set, the objects are clearly observed, however, the gate is partially obstructed by the person.

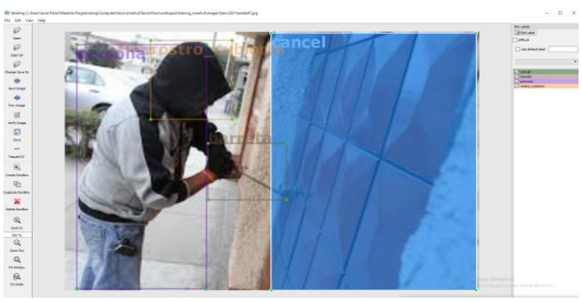

Figure 10 Use of LabelImg in the training set. Annotations of objects were performed on the training set. In this case, the objects are present, although the door is inclined and is different from a glass door.

Using Transfer Learning, through Tensor Flow, a training process for the SSD was set up. Here, the annotations of the images of the training and test sets were transformed into Tfrecord format, which is the labeling format required by the model. The following hyper parameters for the training were configured: 12,000 steps, the number of classes and the evaluation metrics, highlighting that the evaluation and tuning metrics found in the COCO dataset were used. The steps are illustrated in Figure 11.

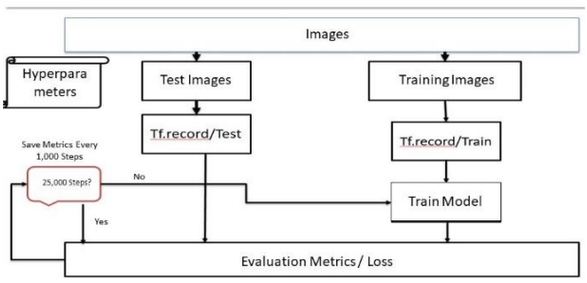

Figure 11 SSD training. The steps for the training design are shown, the hyper parameters are the fine-tuning of the model. Every 1000 steps, the metrics or checkpoint is saved.

Once the model has carried out partial training, the results shown in Tensor Board have been reviewed, in Figure 12 and Figure 13 we show the partial results in 12,000 processed epochs. It can be seen that the error in the classification and the total error tend to decrease with the passage of time.

Figure 12 *Loss function vs Classification.* This tends to decrease. In an optimal model, the curves tend to remain at a value between 0.5 on the Y axis. Some of the objects are not classified correctly, or in their case, they may be one on top of the other.

ISSN 2531-2197

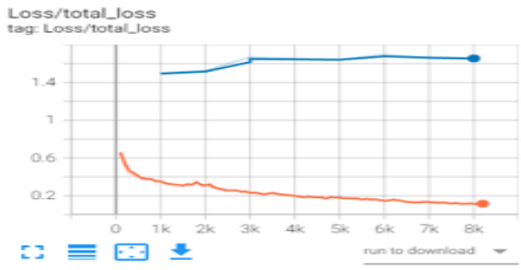

Figure 13 *Loss function vs total loss.* The process is more stable, but it does not tend to improve the curve tendency. An optimal model is close to 1 (orange line).

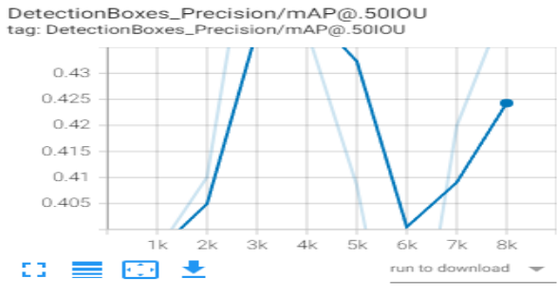

Figure 14 Precision graph at 8,000 epochs. The overall performance of location accuracy is shown. Here we demonstrate that occlusion and model confusion affect the precision of the bounding box

Figure 14 shows the IoU metrics, where the trend is low in the early stages of training. Although it shows instability, it is possible to adjust the IoU value, but the system may identify ROI as slightly different from the true value. Example in Figure 15.

Figure 15 Evaluation, at 6,000 steps. It is possible to observe the output expected by the process, in this case the detection of criminal behavior objects.

Figure 15 illustrates on the left side the figure used in the evaluation and on the right side the truth value annotations, in this case at 6,000 epochs, it is found in the Cancel object, covered face, person, but not the crowbar.

Figure 16 Image normalized in the process. The Tensor Board displays the normalized image. Denoted here is the color space of the blue area, which is a crowbar. However, the gate or other desired object is not denoted.

Figure 16 and Figure 17 illustrate the normalization stage of a training image, here the possibility of using HSV and moments in a manual identification is denoted.

Figure 17 Image normalized in the process. Tensor Board displays the normalized image. Here, the yellow and black color space is denoted, and the desired object (bar) is not distinguishable in color.

Figure 18 shows a normalized image of the training set. Here, the HSV is clearly highlighted, but the shape of the object is not completely clear, the door is transparent, increasing the difficulty.

Figure 18 Normalized training image. Color space denotes the crowbar in red pixels. The person is in a dark color space, but the door is not clearly visible.

ISSN 2531-2197

Figure 19 Image of the training set. Here, we illustrate the normalized image on the left and the raw image on the right. In this case, the normalization does not help in the detection of the object; the model must look for another activation function.

Figure 19 illustrates the complexity of detection. Hand tools can help achieve identification.

Figure 20 illustrates the detection of the crowbar, thus, the identification of the elements of the context has been achieved. For the detection of criminal behavior, it is verified if the 4 objects are present in the scene, taking as reference the labels of the BB.

Figure 20 Image in the validation, the detection of the crowbar is achieved at 9,000 times.

Financing

This work was financed by the Tecnológico Nacional de México (TECNM) and the state of Hidalgo through the software engineering and distributed systems research line of the ITS del Occidente del Estado de Hidalgo and the ITS del Oriente del Estado de Hidalgo.

Conclusions

The proposal meets the proposed objective by using border vision techniques in the video surveillance branch, where the identification of criminal behavior is done with an object detection model and a support function for the recognition of criminal behavior. The quality of the sets affects the final performance, where the quality and quantity of the inputs have an impact on the model, as mentioned in [33].

Using the strategy used in [11] the quality of the geometric transformations is compromised, but the advantage is that the scene contains the objects in their real environment.

Using the techniques used in [13] greatly improves the result, as long as the objects in the scene are not in occlusion. The moments of the scene help to determine some objects, but as long as they are clearly identifiable, regularly those that stand out in color.

The investigation takes the lessons regarding image transformation and uses a regression model for the bounding box. It addresses the tasks of security, video surveillance, and detection of multiple activities with different tools than those proposed in [14], [15], since RetinaNet and a manual tool for behavior identification are used. A solution similar to [17] is implemented, but focused on the identification of an action in human activity.

In this research, what was proposed by [16], where the consistent positioning of objects in a sequence of scenes is discussed, was not implemented. It is addressed in a later work.

The model was designed for an image resolution of 640 x 640 pixels, a lower resolution was found to significantly affect the accuracy metric.

Acknowledgments

We thank the Master's program in Computer Systems of the ITS del Oriente del Estado de Hidalgo (ITESA) and the educational program of Engineering in Computer Systems of the ITS del Occidente del Estado de Hidalgo. Also, Eduardo Daniel Montufar Romero for the revision on the English language revisions of this work.

ISSN 2531-2197

References

[1] Shinde, P. P., & Shah, S. (2018, August). A review of machine learning and deep learning applications. In *2018 Fourth international conference on computing communication control and automation (ICCUBEA)* (pp. 1-6). IEEE. Recover in july, 18, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/document/8697857. DOI:10.1109/ICCUBEA.2018.8697857.

[2] Stubbs, J. J., Birch, G. C., Woo, B. L., & Kouhestani, C. G. (2017, October). Physical security assessment with convolutional neural network transfer learning. In *2017 International Carnahan Conference on Security Technology (ICCST)* (pp. 1-6). IEEE. Recover in july, 18, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/8167800.DOI:10.1109/CCST.2017.8167800.

[3] Shanmugamani, R. (2018). *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras*. Packt Publishing Ltd.

[4] Papageorgiou, C. P., Oren, M., & Poggio, T. (1998, January). A general framework for object detection. *In Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)* (pp. 555-562). IEEE. Recover in july, 18, 2023, of IEEE Explore: https://ieeexplore.ieee.org/abstract/document/710772.DOI:10.1109/ICCV.1998.710772.

[5] Borji, A., Cheng, M. M., Hou, Q., Jiang, H., & Li, J. (2019). Salient object detection: A survey. *Computational visual media*, *5*(2), 117-150. Recover en july fr, 18, 2023, of Springer link: https://link.springer.com/article/10.1007/s41095-019-0149-9. DOI:https://doi.org/10.1007/s41095-019-0149-9.

[6] He, Y., Zhu, C., Wang, J., Savvides, M., & Zhang, X. (2019). Bounding box regression with uncertainty for accurate object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2888-2897). Recover in july, 18, 2023, of ARXEIV: https://arxiv.org/pdf/1809.08545.pdf. DOI: https://doi.org/10.48550/arXiv.1809.08545.

[7] Szeliski, R. (2010). *Computer vision: algorithms and applications.* Springer Science & Business Media.

[8] Jähne, B., & Haußecker, H. (2000). *Computer vision and applications*. Academic Press.

[9] Bai, X., Huang, M., Prasad, N. R., & Mihovska, A. D. (2019, November). A survey of image-based indoor localization using deep learning. In *2019 22nd International Symposium on Wireless Personal Multimedia Communications (WPMC)* (pp. 1-6). IEEE. Recover in july, 18, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/9096144.DOI:10.1109/WPMC48795.2019.9096144.

[10] Penanchino A. Cómo prevenir delitos, como reconocer actitudes sospechosas. Foro de Seguridad. Recover in july, 18, 2023, of Foro de seguridad latinoamericano: https://www.forodeseguridad.com/artic/prevenc/3099.htm.

[11] Ahonen T., Hadid A., and Pietikainen. M. Face recognition with local binary patterns. In Computer vision-eccv 2004, (pp 469–481). Springer, 2004 J. Recover in july 18, 2023, of Springer Link: https://link.springer.com/chapter/10.1007/978-3-540-24670-1_36. DOI: https://doi.org/10.1007/978-3-540-24670-1_36.

ISSN 2531-2197

[12] HU, G., Yang, Y., Yi, D., Kittler, J., Christmas, W., Li, S., & Hospedales, T. (2015). When Face Recognition Meets With Deep Learning: An Evaluation of Convolutional Neural Networks for Face Recognition. *Proceedings of the IEEE international conference on computer vision workshops*, pp. 142-150. Recover in july 18, 2023, of ARXIV: https://arxiv.org/pdf/1504.02351.pdf. DOI: https://doi.org/10.48550/arXiv.1504.02351

[13] Nam, G. P., Choi, H., Cho, J., & Kim, I. J. (2018). PSI-CNN: A pyramid-based scale-invariant CNN architecture for face recognition robust to various image resolutions. *Applied sciences*, *8*(9), 1561. Recover in july18, 2023, of MDIP: https://www.mdpi.com/2076-3417/8/9/1561. DOI:https://doi.org/10.3390/app8091561.

[14] WONG, S., Stamatescu, V., Gatt, A., Kearney, D., Lee, I., & McDonnell, M. (2017). Track everything: Limiting prior knowledge in online multi-object recognition. IEEE Transactions on Image Processing, 26(10), pp. 4669-4683. Recover in july 19,2023, of IEEE Explore: https://ieeexplore.ieee.org/abstract/document/7907205: DOI:10.1109/TIP.2017.2696744

[15] Padmaja, B., Myneni, M. B., & Patro, E. K. (2020). A comparison on visual prediction models for MAMO (multi activity-multi object) recognition using deep learning. *Journal of Big Data*, *7*(1), 1-15. Recover in july 19, 2023, of Springer Open: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00296-8DOI:https://doi.org/10.1186/s40537-020-00296-8.

[16] Lu, Y., Lu, C., & Tang, C. K. (2017). Online video object detection using association LSTM. In Proceedings of the *IEEE International Conference on Computer Vision*. (pp. 2344-2352). Recover in july 19, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/document/8237519. DOI:10.1109/ICCV.2017.257.

[17] Sai, B. K., & Sasikala, T. (2019). Object detection and count of objects in image using tensor flow object detection API. In 2019 I*nternational Conference on Smart Systems and Inventive Technology* (ICSSIT) (pp. 542-546). IEEE. Recover on july 19, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/8987942.DOI:10.1109/ICSSIT46314.2019.8987942.

[18] Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly Media Inc.

[19] Gallant, S. I. (1990). Perceptron-based learning algorithms. *IEEE Transactions on neural networks*, *1*(2), pp. 179-191. Recover on july 19, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/document/80230 DOI: 10.1109/72.80230.

[20] Villanueva Limón, F. J. (2017). *Algoritmo de protección con reconocimiento de patrones usando una red neuronal para líneas de transmisión*. [Master's Thesis]. Instituto Politécnico Nacional. Recover in july 19, 2023 of Repositorio Dspace: https://tesis.ipn.mx/bitstream/handle/123456789/22770/tesis_FJVL.pdf?sequence=1&isAllowed=y

[21] Alla, S., & Adari, S. K. (2019). *Beginning anomaly detection using Python-based deep learning.* New Jersey: Apress.

[22] Stretcu O, Leordeanu M (2015, September) Multiple frames matching for object discovery in video. In: *BMVC* (Vol 1m No. 2. p. 3). Recovery in july 19, 2023, of https://meilongzhang.github.io/assets/pdf/videoPCA/bmvc_videopca.pdf.

[23] Forsyth, D., & Ponce, J. (2011). *Computer vision: A modern approach* (p. 792). Prentice hall.

ISSN 2531-2197

[24] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (pp. 21-37). Springer International Publishing. Recover in july 19, 2023, of Springer Link: https://link.springer.com/chapter/10.1007/978-3-319-46448-0_2 DOI: https://doi.org/10.1007/978-3-319-46448-0_2.

[25] Girshick, R. (2015). Fast r-cnn. In Proceedings of the *IEEE international conference on computer vision* (pp. 1440-1448). Recover in july 19, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/document/7410526. DOI: 10.1109/ICCV.2015.169.

[26] Li, J., Li, C., Fei, S., Ma, C., Chen, W., Ding, F.,& Xiao, Z. (2021). Wheat ear recognition based on RetinaNet and transfer learning. *Sensors*, *21*(14), 4845. Recover in july 20, 2023, of MDPI: https://www.mdpi.com/1424-8220/21/14/4845. DOI: https://doi.org/10.3390/s21144845.

[27] Vujović, Ž. (2021). Classification model evaluation metrics. *International Journal of Advanced Computer Science and Applications*, *12*(6), 599-606. Recover in july 19, 2023, of Research Gate: https://www.researchgate.net/profile/Zeljko-Vujovic/publication/352902406_Classification_Model_Evaluation_Metrics/links/60dede1592851ca9449f17bb/Classification-Model-Evaluation-Metrics.pdf. DOI: 10.14569/IJACSA.2021.0120670.

[28] Tarang S. (2018) Medición de modelos de detección objetos - mAP - ¿Qué es la precisión media? on Towars data Science. Recover in july 19, 2023, of Towards Data Science: https://towardsdatascience.com/what-is-map-understanding-the-statistic-of-choice-for-comparing-object-detection-models-1ea4f67a9dbd.

[29] Python Software Fundation (2021) LabelImg on Python Package Index Recover in july 19, 2023, of Python: https://pypi.org/project/labelImg/1.4.0/

[30] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D.,… & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer, Cham. Recover in july 19, 2023, of Springer Link: https://link.springer.com/chapter/10.1007/978-3-319-10602-1_48. DOI: https://doi.org/10.1007/978-3-319-10602-1_48.

[31] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., ... & He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, *109*(1), 43-76. Recover in july 19,2023, of IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/9134370. DOI: 10.1109/JPROC.2020.3004555.

[32] T. Carron and P. Lambert (1994, Noviembre), Color edge detector using jointly hue, saturation and intensity. In *Proceedings of 1st International Conference on Image Processing* (Vol. 3, pp. 977-981). IEEE. Recover in july 19, 2023, of IEEE Xplore: https://ieeexplore.ieee.org/abstract/document/413699 DOI: 10.1109/ICIP.1994.413699.

[33] Qi, S., Zhang, Y., Wang, C., Zhou, J., & Cao, X. (2021). A survey of orthogonal moments for image representation: theory, implementation, and evaluation. A*CM Computing Surveys (CSUR)*, *55*(1), 1-35. Recover in july 20, 2023, of ACM Digital Library: https://dl.acm.org/doi/abs/10.1145/3479428 DOI:https://doi.org/10.1145/3479428.

ISSN 2531-2197

[34] Fagarasan, C., Popa, O., Pisla, A., & Cristea, C. (2021, August). Agile, waterfall and iterative approach in information technology projects. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1169, No. 1, p. 012025). IOP Publishing. Recover in july 20, 2023, of IOP Science: https://iopscience.iop.org/article/10.1088/1757-899X/1169/1/012025/meta. DOI 10.1088/1757-899X/1169/1/012025.

ISSN 2531-2197