Watson: Computer Vision based Software for Summarization of Crime Scenes

Dr. S.V. Kedar, Satvik Sharma, Amit Sarje, Avadhoot Kabadi, and Aditya Sathe

Article Info
Page Number: 3157 - 3170
Publication Issue:
Vol 71 No. 4 (2022)

Article History
***Article Received**: 25 March 2022*
***Revised**: 30 April 2022*
***Accepted**: 15 June 2022*
***Publication**: 19 August 2022*

Abstract

In today's world, the crime investigation has reached a significant height due to high power and accurate computing technologies of the present time. For any such general investigation, evidences play an important role. Photographs of such scene give the idea of the objects, samples or areas involved in a crime scene. To analyse the digital evidences, various machine learning methods and algorithms are being developed/used. (E.g. blood/weapon/body detection, 3d reconstruction etc.). However, we also need the reports or the shortened summary of such scenes which may need the human intelligence. Inspired by this idea, in this paper, we have introduced our proposed system which generates the summary of a crime scene provided the images of it from different angles. We have also discussed the algorithms, techniques, and limitations of our system.

Keywords: Crime investigation, Forensic Science, Computer vision, Machine learning, Image Summarization.

1 Introduction

The population of the human race is increasing at a greater pace than before. The civilized system has increased uncivilized criminal activities. Technological usage to control, reduce and investigate these crimes is also increasing. The pace is considerable. Increase in media like photos and images has drastically helped the authorities to solve the investigations of such crimes. With advancement in artificial intelligence techniques, using methodologies like reconstructing crime scenes in 3D reimagine crime for better investigation, using computer vision to reduce ineffectiveness of surveillance cameras, 4D reconstruction and VR for better reprocessing of crime scenarios, identifying dangerous situations for crime prevention has been done over the years. This work is enormously considerable towards using technology in reducing crime. Along with this, we have proposed the use of computer vision and its current facilitating methodologies in crime investigation in this paper. This paper introduces the technique which focuses on reducing the workload of doing the paperwork after a crime.

The objective of this implemented project is to generate a summary of a crime scene using the photograph provided. This application of Computer Vision technology in crime investigation makes the hectic work of humans easier and quicker. Use of technology reduces the ill practices of intentionally written misleading reports. It improves accuracy of summarization of crime scenes by capturing clues that a human eye might miss to identify as the application will be training and improving over and over. The straightforward motto is to reduce human effort and increase the transparency of investigations using Computer Vision. This bridges the gap that has been created in the existing technologies that help in surveillance of such situations. It attempts to fit in the existing working system of authorities rather than completely changing the work method that is previously attempted in various mentioned above papers.

1.1 Related Work:

Attention based object detection and Object Localization are two main techniques used nowadays to find objects in a given image. As far as the motive of our work of summary generation has not been touched. But related work like one-line caption generation has already been introduced to the community.

Kiros et al. (2014a) introduced the first method to make use of neural network in caption synthesis, which made use of a model which was multimodal log-bilinear that was biased by image attributes. As a significant exception to this global representation technique, Karpathy and Fei-Fei in reference paper [19] collected characteristics from various image parts using an object detector of R-CNN with respect to paper [20] and created independent captions for every part. The dimensional link among the recognised objects was not modelled because each region received its own caption.

Image descriptions were developed by Fang et al. in reference paper [21] by first recognising words related to distinct places inside the picture. The dimensional association was achieved by producing spatial response maps for the target words using a complete CNN applied to the picture. The authors did not manually model any of the linkages among the spatial regions in this case as well.

The most successful subsequent study was using an object detector to draw out visual characteristics and an attention LSTM for producing captions. Yao et al. presented two Graph Convolutional Networks in reference paper [22] to add global context: a spatial graph of relationship and a semantic graph of relationship that arrange the relationship among two boxes into 11 types, like "cover", "inside", "overlap".

(Cho et al., 2014) with respect to reference paper [23] The encoder-decoder framework of machine translation is ideally suited because it is equivalent to "translating" a picture to a sentence.

2 Data-set preparation and Pre-processing:

Our domain requires crime scene photographs for training the model. For any generalized machine learning model, the quantity of dataset is a crucial necessity. Since getting criminal photographs in large amounts is not practically possible as the required forensic digitalized data is confidential, we created our own dataset using screenshots of crime scenes from high quality video games.

To write a report for any particular crime scene different parameters are considered. After a extensive analysis we found the major parameters that defines a crime scene. Those parameters include human presence, the blood if any, surroundings, description of victim’s attire, number of victims if any, wounds or injuries on body if any, time of day when the incident was reported, etcetera.

Due to the lack of a suggested data set for training the model, one has been constructed for the proposed work. So, 171 images were taken from different high-quality games, multiplied by a factor of 10 using image multiplier libraries and 200 images from MS COCO are collected. Hence, the total images collected are 1700+ for the dataset. Each of these cropped samples is then re-sized to a 299 × 299 size image.

For every image we need captions describing the parameters those were mentioned before. To handle this kind of mapping, we created a dataset as a json file (image_id as key and caption to be its value).

Each image is renamed and saved with an id that corresponds to the json file's caption id. Multiple captions relating to each image are added to expand the size of the data-set. Both DatasetA and MS COCO include five reference sentences per image, but some of the photos in the dataset have more than five references to ensure consistency throughout our datasets. For MS COCO, we merely used basic tokenization. We employed a predetermined vocabulary size of ten thousand for our experiments.

2.1 Creating JSON file:

All we did now is build a JSON file with the target variable and a list of models to run our data through. Similarly, we can change the data before providing it to the model by pre-processing it. We only need to pass the natural language representation of the operation we need to perform in the JSON file. For simplicity of use, the json file has the same format as the coco dataset.

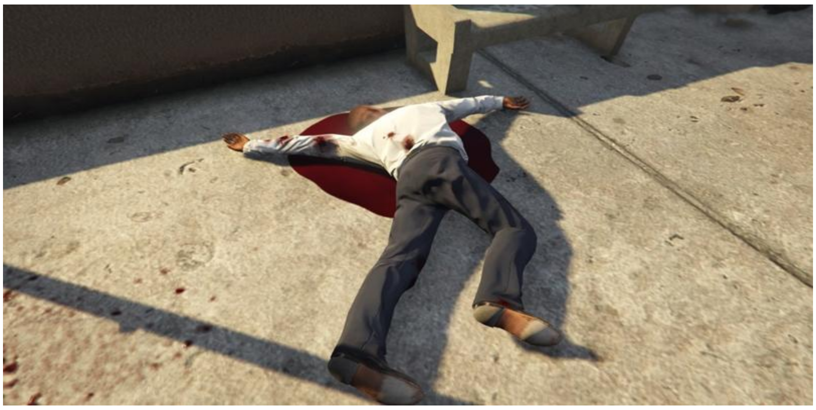
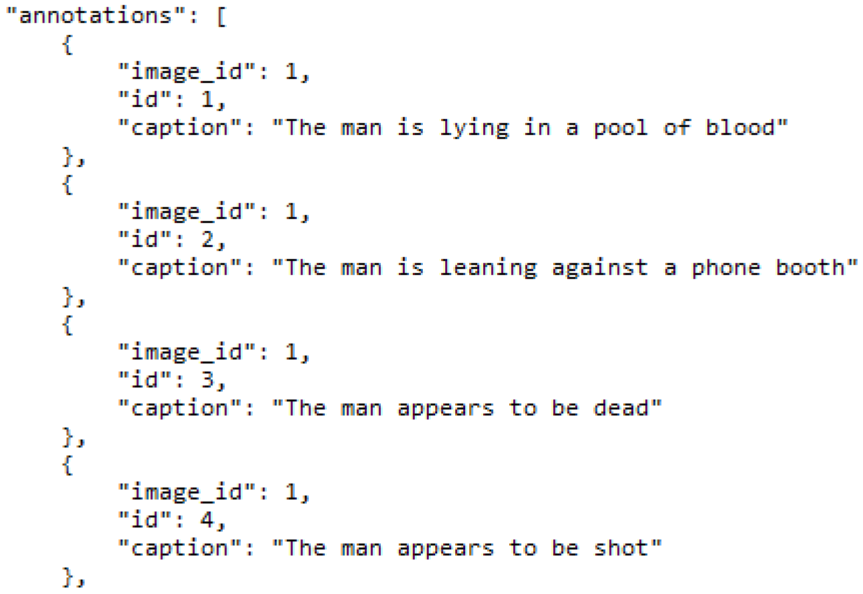


Fig. 1. Crime Scene sample image. This image is a sample image from dataset which is used for summary generation.



```
"annotations": [
    {
        "image_id": 1,
        "id": 1,
        "caption": "The man is lying in a pool of blood"
    },
    {
        "image_id": 1,
        "id": 2,
        "caption": "The man is leaning against a phone booth"
    },
    {
        "image_id": 1,
        "id": 3,
        "caption": "The man appears to be dead"
    },
    {
        "image_id": 1,
        "id": 4,
        "caption": "The man appears to be shot"
    },
```

Fig. 2. Screenshot JSON File. This image shows mapping of images are with respective captions.

Suppose every image is marked with image_id (in above figure 2.1 “image_id: 27”) and then filled with captions. For the above figure given “caption” to “image_id: 27” are: [“The man is lying in a pool of blood”, “The man was shot four times in the back”, “The man appears to be dead”, “The man is wearing a T- shirt”]. Data is pre-processed in this way.

3 Proposed Model

3.1 Attention Mechanism Model for Image Caption Generation Details:

Among the two famous models for caption generation

1. Attention based
2. Object localisation.

We focused upon attention-based feature extraction.

3.2 Encoder: Description of Convolutional Features:

This model accepts input as a single raw image. Further it produces a caption. This caption (say) v. A sequence of 1-of-K encoded words is encoded by v.

$$V = \{v_1, \ldots, v_C\},\ v_i \in R^K$$

where caption length is C and the vocabulary size is K.

A set of feature vectors is also referred to as annotation vectors. We employ a CNN to extract a set of feature vectors. L vectors generated by the extractor, each corresponds to a D-dimensional portrayal of a portion of the picture.

$$M = \{m_1, \ldots, m_L\},\ m_i \in R^D$$

The previous work used a layer which was fully connected. In contrast to previous work, we draw out characteristics from a layer of lower convolutional to get a correlation among feature vectors and sections of 2-D picture. The decoder can particularly concentrate on some portions of a picture by picking a “all feature vectors subset”.

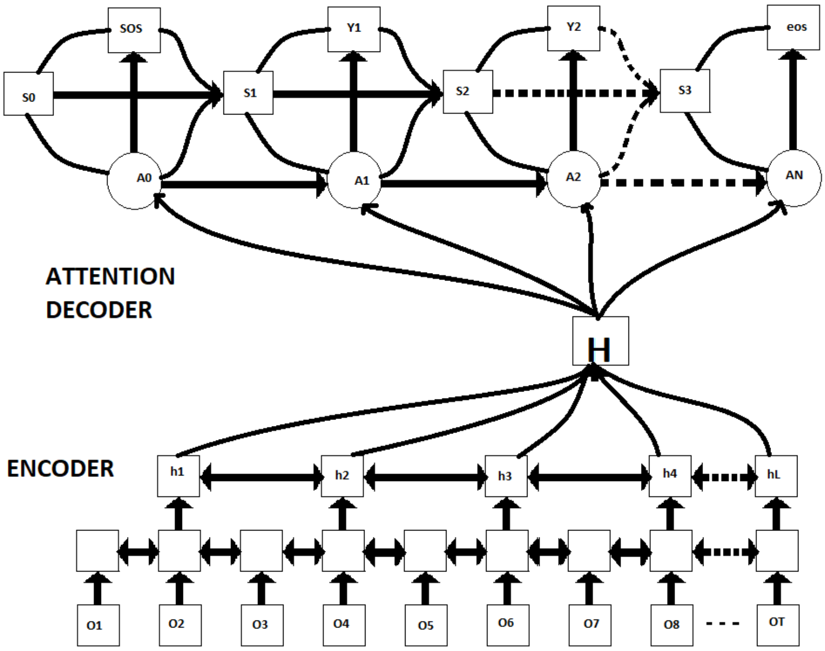


Fig. 3. Architecture of Encoder Decoder.

3.3 Decoder: LSTM Network:

We employ an LSTM network reference to "Hochreiter & Schmidhuber" of 1997 to construct an inscription by synthesizing single word at each step of time based on the prior hidden state, a frame of reference vector, and words that were previously generated. Using the notation $T_{s,t}$:$R^s \rightarrow R^t$ to represent a basic connected transformation with learned parameters,

$$\begin{array}{lclc} i_t & & \sigma & E y_{t-1} \\ f_t & = & \sigma \quad T_{D+m+n,n} & h_{t-1} \\ o_t & & \sigma & \hat{z}_t \\ g_t & & \tanh & \end{array} \qquad (1)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \qquad (2)$$

$$h_t = o_t \odot \tanh(c_t) \qquad (3)$$

Here, $h_{t,}$ o_t, c_t, f_t, i_t are the hidden, output, memory, forget, and input state of the LSTM, respectively. The vector $\hat{z} \in R^D$ is the reference frame vector, taking the visible information linked with a selected input place, as explained down. $E \in R^{m\times K}$ is a matrix which is embedding. Let 'm' represent the embedding and 'n' represent the LSTM dimensionality respectively. Let σ represent the logistic sigmoid activation and ⊙ represent the element-wise multiplication respectively.

In clear words, the reference vector $\hat{z}_t$ (equations (1) – (3)) is a dynamic depiction of the relatable region of the picture input at time t.

We provide a technique φ for computing z_t from the annotation vectors a_i, i = 1, ..., L, which correspond to the features drawn out at various picture regions. The technique produces a constructive weight denoted as α_i for every location i which can be comprehended as either the corresponding importance to give to region point i in mixing the a_i's together or the chance that location i is the best place to concentrate for producing the upcoming word (the "hard" but a bit random and uncertain attention technique). An attention model f_{att} computes the weight α_i of each annotation vector a_i. We use a multilayer perceptron conditioned on the prior hidden state h_{t-1} for this purpose.
Bahdanau et al. (2014) proposed a soft form of this attention technique. We emphasize that the “hidden state” undergoes change as the RNN output progresses through its output series: where the net tries to find the upcoming is determined by the previous series of text.

$$e_{ti} = f_{att}(a_i, h_{t-1}) \quad (4)$$

$$\alpha_{ti} = \exp(e_{ti}) / \Sigma^{L}_{k=1} \exp(e_{tk}) \quad (5)$$

The context vector $\hat{z}_t$ is computed after the weights (which add to one) have been computed.

$$\hat{z}_t = \varphi(\{a_i\}, \{\alpha_i\}), \quad (6)$$

The φ is a function that produces a single vector as output. It takes their weights and a set of annotation vectors. An average of the vectors of annotations fed through 2 distinct MLPs (init, h and init, c) predicts the LSTM's “hidden state” and “initial memory state”:

$$c_0 = f_{init,c}(1 / L \, \Sigma^{L}_{i} a_i)$$

$$h_0 = f_{init,h}(1/L \, \Sigma^{L}_{i} a_i)$$

Given the LSTM state, the context vector, and the preceding word, (with reference to Pascanu et al., 2014) to calculate the output word probability, we utilize a deep output layer:

$$p\,(y_t \,|\, a,\, y^{t-1}{}_{1}) \propto \exp\,(L_o\,(E\, y_{t-1} + L_h h_t + L_z \hat{z}_t)) \tag{7}$$

The are learned parameters are $L_z \in R^{m\times D}$, $L_h \in R^{m\times n}$, $L_o \in R^{K\times m}$ and E which are initialized randomly.

4 Experimental Analysis:

4.1 Calculating accuracy:

We split our dataset with an offset of 0.1. We trained our model with 90% of the dataset and the remaining part of it was reserved as the test data. To calculate the accuracy, we evaluated the Hamming distance between the actual caption and predicted caption. Consider two strings with same length. The Hamming distance among those two is the number of positions at which the correlated letter/character is distinct. So, we checked the percentage by which the characters were different in actual and predicted captions and if the percentage value is less than 20%, we defined the respective captions as similar. With the mentioned method we could achieve an accuracy of 90-92%.

5 Results:

The system used for training the model is ASUS intel i5. The time for the epoch is around 450-500 seconds for a single epoch. The loss function of a neural network could be defined as the error of its predictions over a fixed dataset. The below mentioned image comprehends the loss plot as a two-dimensional line graph which represents the No. of epochs and loss on X and Y axis respectively.

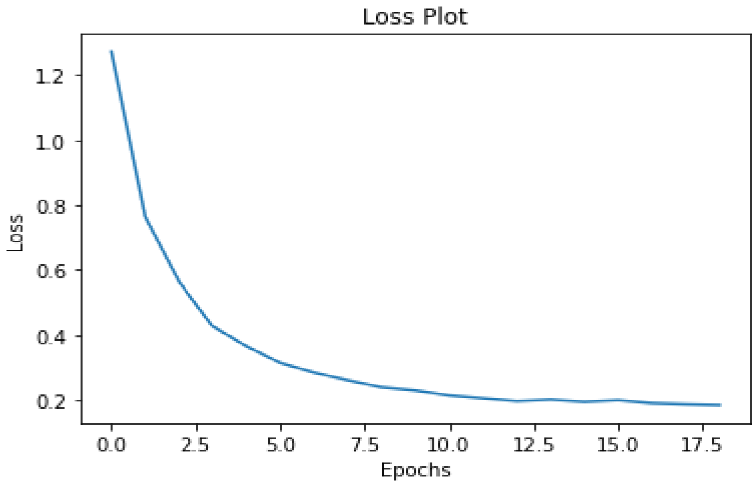


Fig. 4. Loss Plot

5.1 Example:

Below figures gives the idea of result at every epoch and the corresponding attention based feature extractions.

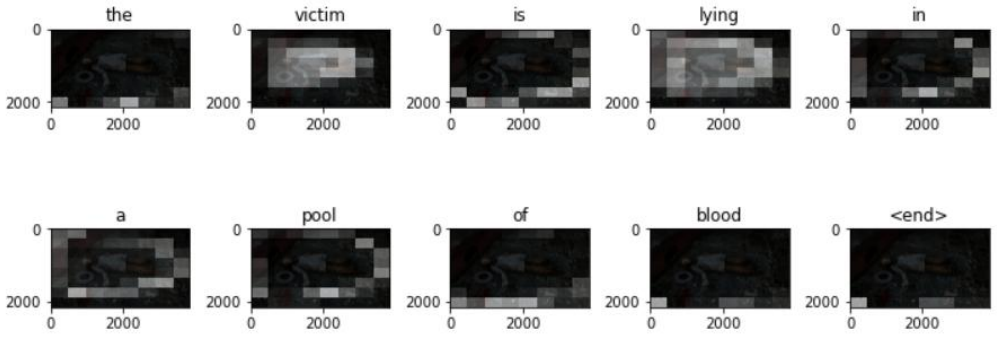


Fig. 5.1. Epoch 1

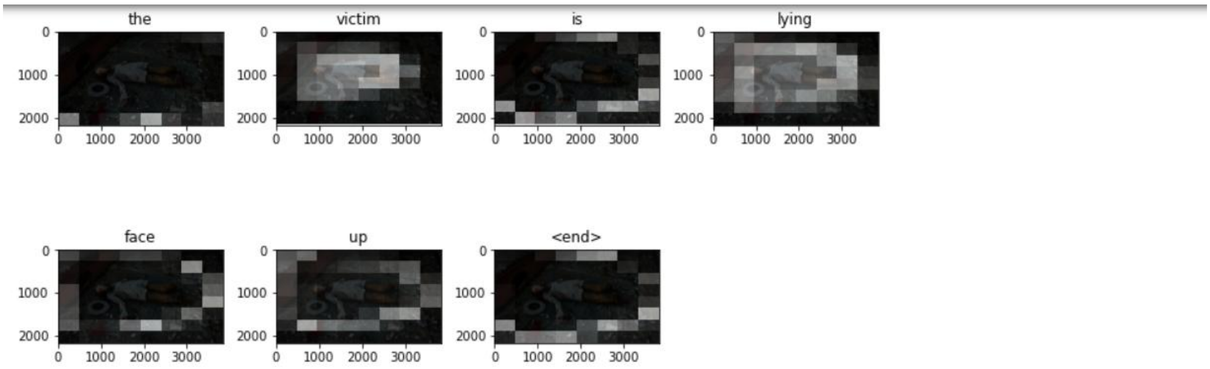


Fig. 5.2. Epoch 2

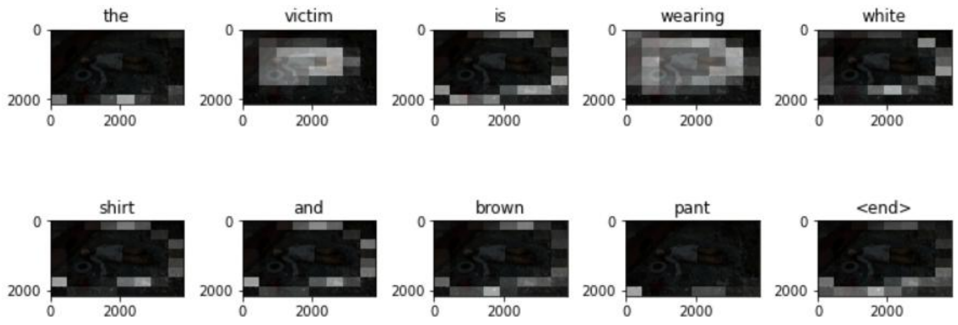


Fig. 5.3. Epoch 3

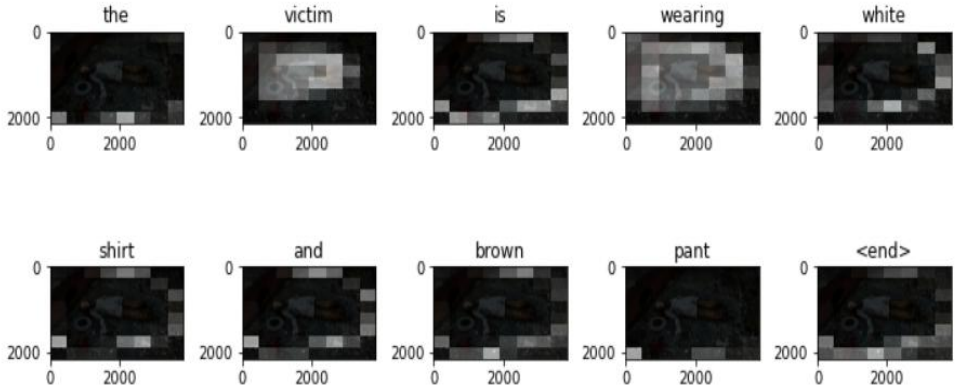
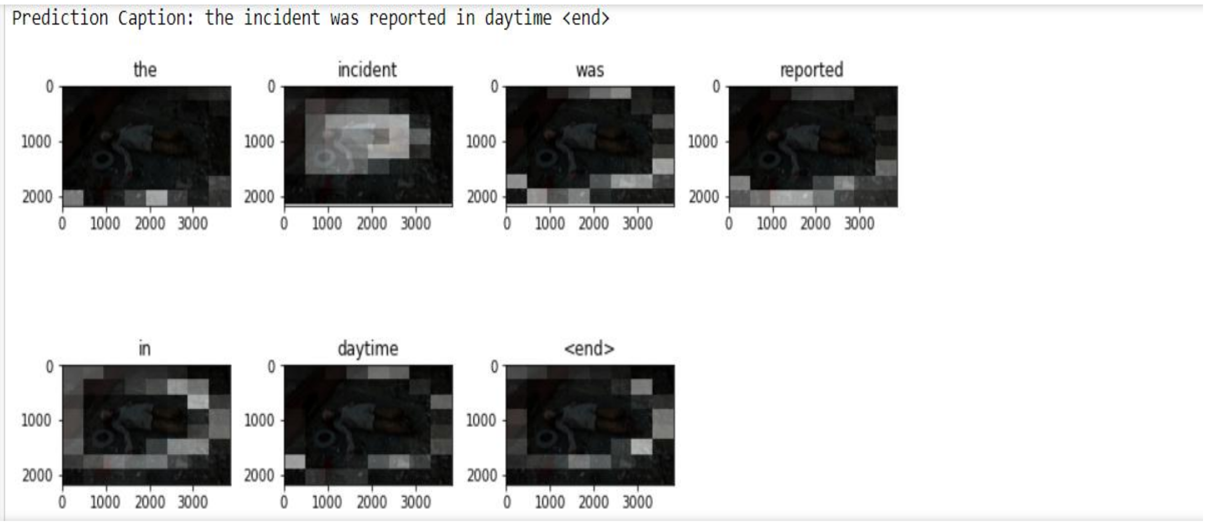


Fig. 5.4. Epoch 4



```
Prediction Caption: the incident was reported in daytime <end>
```

Fig. 5.5. Epoch 5

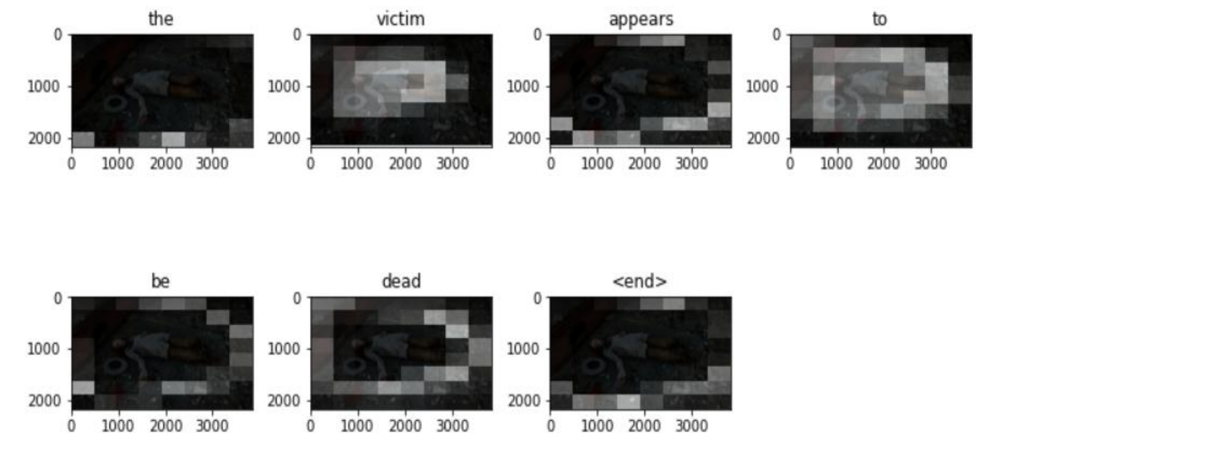


Fig. 5.6. Epoch 6

Results of the trained model are as shown below:

Explanation:

Input: “Image_id” = 84

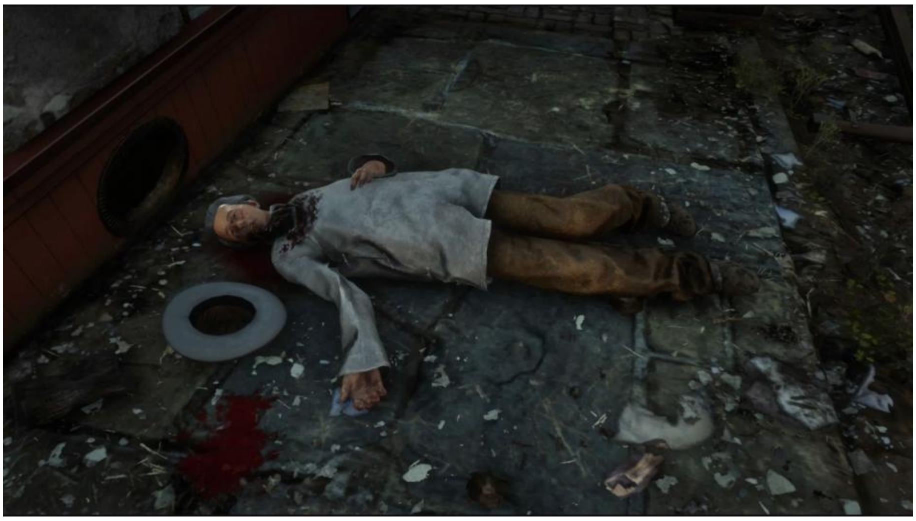


Fig. 5.6. Sample Tested Image

Output: “The victim appears to be dead. The incident was reported at nighttime. The victim is lying in the pool of blood. The victim’s body has fallen on the road. The victim has wounds on body. The victim is wearing white shirt and brown pants. Blood splatters could be seen.”

The above result was obtained by running our model multiple times on the input image. To avoid getting similar sentences in the result, we used a method which uses Levenshtein’s distance method algorithm and a hash table.

6 Limitations:

The work done till now is completed to produce a prototype result for achieving the objective. The limitations that are yet to covered are as follows:

- The dataset requires images of crime scenes. Obtaining real time crime scenes images which are highly confidential from authorities is an arduous task. Keep this in mind, we have used images from high resolution games which are near realistic images. But as the images are manually clicked, the images used for the data sets are minimal and adequate in number. This led to a small size of dataset. This may cause overfitting at times.

- 5.2. As an effect, the model is in its prototype phase which has led to output being simple statement summary. The number of captions to id of images is mapped in manual fashion which has created limitation of model getting less input to train leading to simple output summary statements.
- 5.3 There are various types of crimes out of which the model is currently trained to work only for crimes like murders and homicide crimes. The limitation is to make it work on crimes like robbery, frauds, and other non-human harming crimes, which could be solved by adding the datasets related to the specific crime.

7 Conclusion:

This implementation paper focused on the use of Inception-v3 CNN architecture to produce summary of crime scene images which uses the custom created dataset. This is of the Inception family which does many advances involving the use of Factorized 7 x 7 convolutions, Label Smoothing, and the usage of a supplementary classifier to promote label details further down the network. Along with current proposed system we could generate summaries of a crime scene with most commonly present parameters. We also introduced our own method to calculate the accuracy which came out to be 90-92%.

8 References:

1. Kasim Taskin, “Evaluating Augmented Reality and Computer Vision for Crime Scene Investigation”.
2. Erkan Bostanci, “3D Reconstruction of Crime Scenes and Design Considerations for an Interactive Investigation Tool”.
3. ALAYESANMI FEMI SAMSON, FRANCISCA OLADIPO & EMEKA OGBUJU , “An Object Detection and Classification Model for Crime Evidence Analysis Using YOLO” , Proceedings on Big Data Analytics & Innovation (Peer-Reviewed), Volume 3, 2018, pp.18-26.
4. Che-Yen Wen, 1,* Ph.D.; Chiu-Chung Yu, 1 M.S, “Image Retrieval of Digital Crime Scene Images”, Forensic Science Journal 2005; 4:37-45.
5. Ying Liu, Yanan Peng, Daxiang Li , Jiulun Fan, Yun Li , “Crime Scene Investigation Image Retrieval with Fusion CNN Features Based on Transfer Learning”.

6. Matthias Kraus , Thomas Pollok , Matthias Miller ,Timon Kilian ,Tobias Moritz ,Daniel Schweitzer, Jürgen Beyerer ,Daniel Keim ,Chengchao Qu and Wolfgang Jentner , “Toward Mass Video Data Analysis: Interactive and Immersive 4D Scene Reconstruction”

7. Haroon Idrees, Mubarak Shah, Ray Surette,” Enhancing camera surveillance using computer vision: a research note”.

8. Devishree D. S, Divakar K. M, Ashini K. A, Arnav Singh Bhardwaj, Sheikh Mohammad Younis,” Crime scene prediction and analysing its accuracy with frames using deep neural network”, ISSN: 2454-132X, Impact factor: 4.295, (Volume 5, Issue 2).

9. Drishti Jalgaonkar, Juilee Gund, Neha Patil, Madhura Phadke , “Detecting Crime Scenes using ML” , International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 07 Issue: 05 | May 2020.

10. Neil Shah, Nandish Bhagat & Manan Shah , “Crime forecasting: a machine learning and computer vision approach to crime prediction and prevention”

11. Fernanda A. Andaló, Siome Goldenstein , “Computer vision methods applicable to forensic science”, August 2013 ,Conference: Workshop of Theses and Dissertations, XXVI Conference on Graphics, Patterns and Images (WTD/SIBGRAPI '13)At: Arequipa, Peru

12. Surajit Saikia ,Eduardo Fidalgo , Enrique Alegre ,Laura Fernández-Robles , “Object Detection for Crime Scene Evidence Analysis Using Deep Learning” , September 2017 ,DOI:10.1007/978-3-319-68548-9_2 ,Conference: International Conference on Image Analysis and Processing

13. Richard Szeliski. Computer vision: algorithms and applications. 2010. Springer Science & Business Media.

14. Michael A Nielsen. Neural networks and deep learning. Vol. 2018. 2015. Determination press San Francisco, CA, USA

15. David G Lowe. “Distinctive image features from scale-invariant key points”. In: International journal of computer vision 60.2 (2004), pp. 91–110. Springer.

16. G. Bradski. “The OpenCV Library”. In: Dr. Obb’s journal of Software Tools (2000).

17. Marc Levoy and Turner Whitted. The use of points as a display primitive. Cite seer, 1985.

18. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention.Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio

19. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3128–3137, 2015.

20. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 580–587, 2014.

21.H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1473–1482, 2015.

22. T. Yao, Y. Pan, Y. Li, and T. Mei. Exploring visual relationship for image captioning. In European Conference on Computer Vision, pages 684–699, 2018.

23.Cho, Kyunghyun, van Merrienboer, Bart, Gulcehre, Caglar, Bougares, Fethi, Schwenk, Holger, and Bengio, Yoshua. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In EMNLP, October 2014.