UNIVERSITY OF INFORMATION TECHNOLOGY,
VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY

COMPUTATIONAL THINKING
FINAL PROJECT REPORT
CS117.L21

Topic: *Fire detection problem in public places.*

Instructor: ***Dr. Ngô Đức Thành***

Students:		
	Full name	**ID**
1	**Phạm Quang Vinh**	**19522526**
2	**Trương Xuân Linh**	**19521759**

HO CHI MINH CITY – 08/2021

TABLE OF CONTENTS

I. Introduction:

Today, climate change is becoming more and more serious, the average temperature is increasing, and fires are happening more and more, especially in tropical countries like Vietnam. Therefore, accurate and timely fire detection is extremely important.

With rapid economic development, the increasing scale and complexity of constructions has introduced great challenges in fire control. Therefore, early fire detection and alarm with high sensitivity and accuracy is essential to reduce fire losses. However, traditional fire detection technologies, like smoke and heat detectors, are not suitable for large spaces, complex buildings, or spaces with many disturbances. Due to the limitations of above detection technologies, missed detections, false alarms, detection delays and other problems often occur, making it even more difficult to achieve early fire warnings.

Recently, image fire detection has become a hot topic of research. The technique has many advantages such as early fire detection, high accuracy, flexible system installation, and the capability to effectively detect fires in large spaces and complex building structures. It processes image data from a camera by algorithms to determine the presence of a fire or fire risk in images.

Because of these reasons, we choose the topic: “Fire detection problem in the public places” for the final project.

Figure 1: The fire was recorded by CCTV

II. Problem Identification:

- ***Input:*** a frame (picture) of video stream from CCTV.

  Constrains:

    - CCTV is placed at a height of about 4 - 10 m above the ground.
    - The viewing angle of the camera is from 40 - 70 degrees.
    - Minimum video resolution greater than 480p.

- ***Output:***

  + The frame above with location (bounding box) of fires and humans.

  + The danger level of the fire (Text):

    - "Warning": if a dangerous fire is detected.
    - "No Warning": if the detected fire is not dangerous.

Examples:

Output

Input

Figure 2: Example of dangerous fire

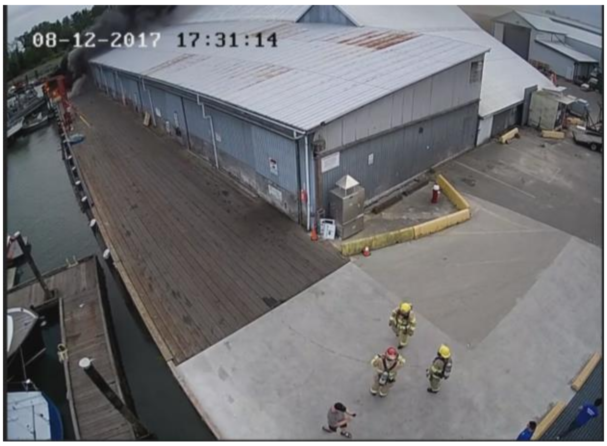

Figure 3: Example of no dangerous fire

III. Computational Thinking:

1. Decomposition:

With the big problem is fire detection, we decompose this problem into the following smaller sub-problems:

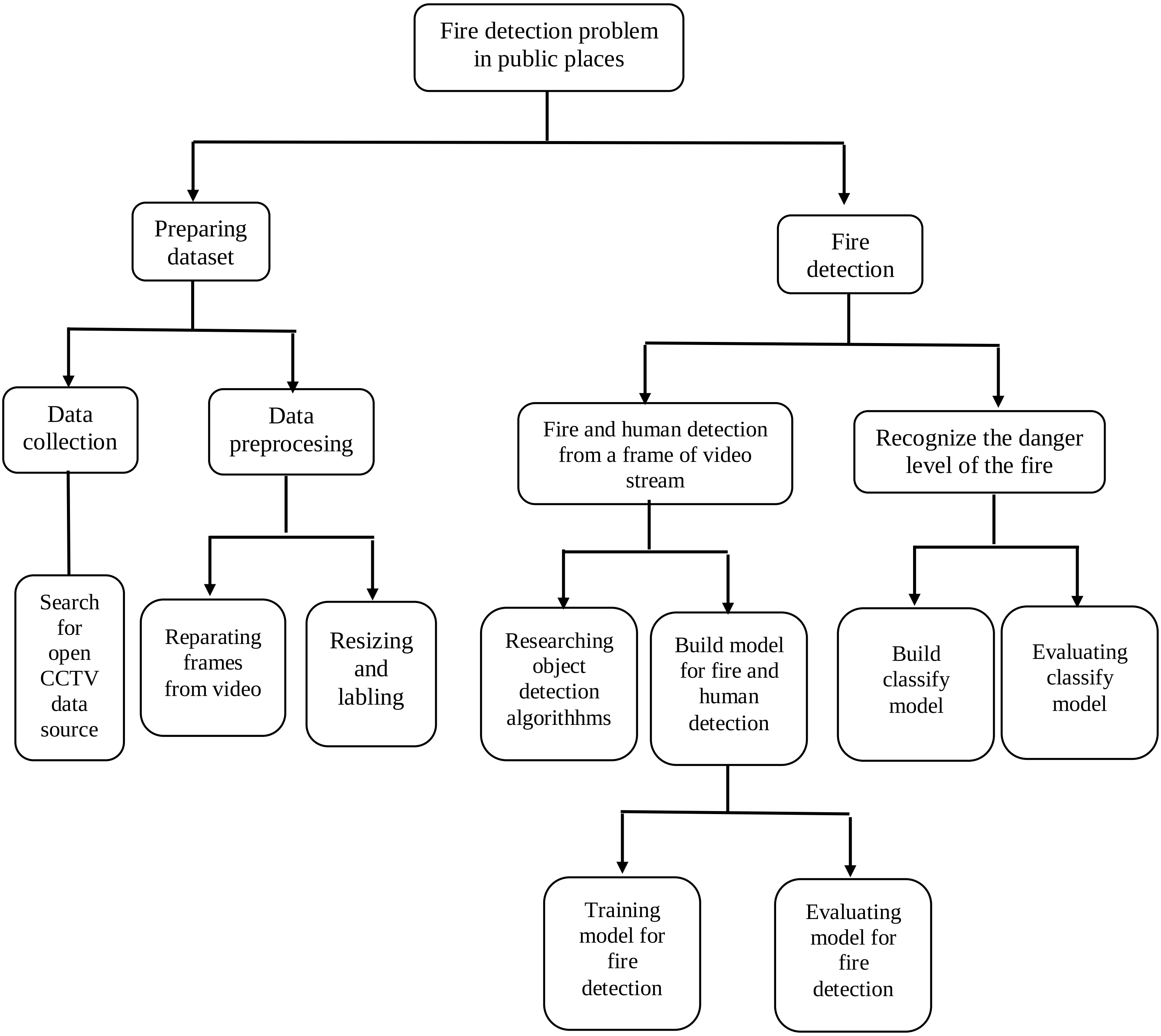

Figure 4: Problem decomposition process

a. How can a computer detect a picture containing a fire?

The input that we have is a video stream. However, in essence, video is a collection of frame images. And for the computer to read those images, its essence is a matrix of different numbers. So, we need to "teach" computers to know what are the specific values for images which have a fire scene and know what are the specific values for images which don't have a fire scene. To "teach" computers to learn these things, we need a "support tool", Machine Learning or Deep Learning are the most useful ones.

Our computer needs to "learn", so we need to give the computer a collection of images which have a fire scene and a collection of images which don't have a fire scene. This dataset goes through the preprocessing, feature extraction, training and fine - tuning the parameters to finally obtain the model to predict the other images.

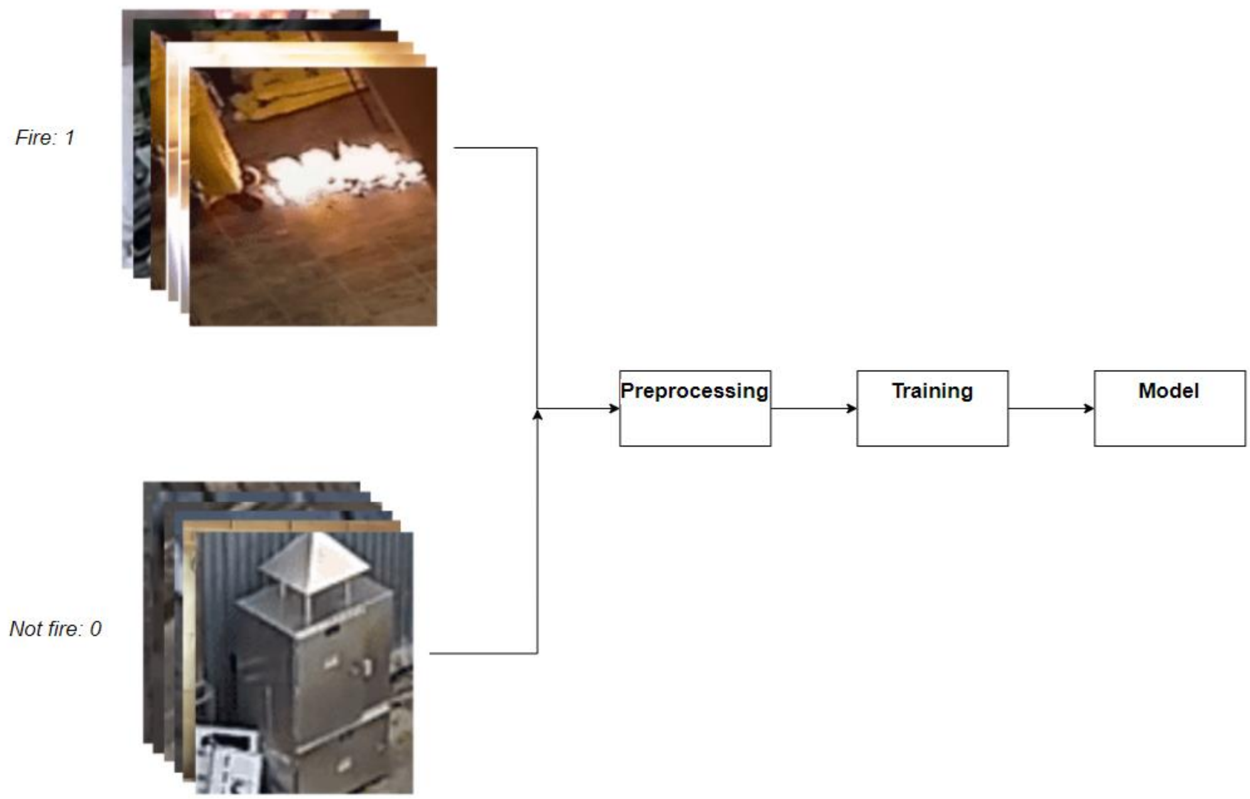

Figure 5: Traning fire detection model

b. The difference between images containing dangerous fire scenes and images containing not dangerous fire scenes ?

In the case of finding the difference between images containing dangerous fire scenes and images containing not dangerous fire scenes, we just need to consider that at other locations like: school campuses, dormitories, companies, the fires are being handled by humans.

Figure 6: Example of dangerous and non-dangerous fire

We can easily see that the difference between dangerous fires and not dangerous fires is the appearance of humans. If a place contains a fire scene that has the appearance of humans except fire forest, it can be considered a not dangerous fire and if those fires are detected, we don't need to give a warning. In contrast, if the fires are without human supervision, those are potential risks to be dangerous fires, more bigger if those are not found out early. So, we will stipulate that if an image contains the appearance of humans, it will be marked as a *non-fire image*. In the other hand, if an image does not contain the appearance of humans, instead, an image containing the fire scene, it will be marked as a *fire image*.

Figure 7: Dangerouse fire without human

Figure 8: Non-dangerous fire with humans

Then, the problem will generate a smaller subproblem. It is human body detection in the image. Similar to the fire detection problem in the image, we also will “teach” computers to know if an image has the appearance of humans or not. To do that, we still need a dataset that is a collection of the human body and the other image without the human body. Then, those will be labeled as the *human body* or *others*. This dataset goes through the preprocessing, feature extraction, training and fine - tuning the parameters to finally obtain the model to predict the other image

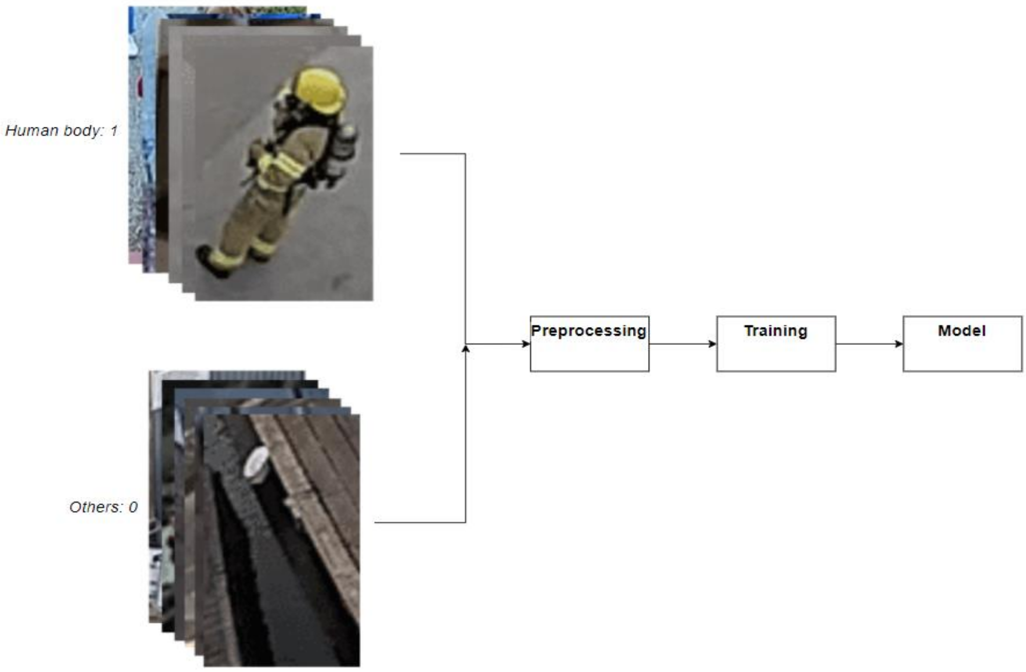

Figure 9: Traning human detection model

c. Model combination.

Summary, we have two machine learning models, one to predict the image containing the fire scene with the others and one to predict the image containing the human body with the others. So, to save operating time, increase quality, we can combine two models into a single model and build classfication model to predict three different labels: the fire, the human body and the others.

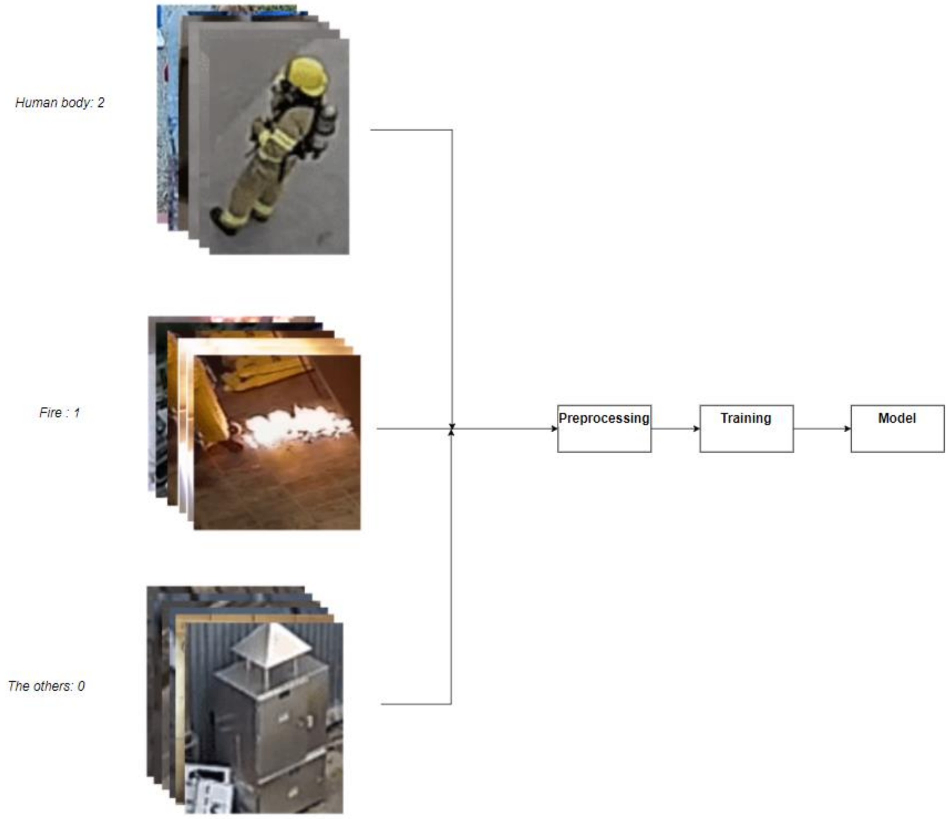

Figure 10: Build classification model

2. Pattern recognition:

- **Object detection problem:** we recongnize that this is the object detection problem, when we detect the location of the fire and peoples, so we research some deep learning algorithms about object detection.
- **Classification problem:** not only detect the fire but also we need to classify it into warning class and no warning class.

3. Abstraction:

In reality, cameras are usually placed in fixed positions and don't change position for a long time. So, the background that the camera captures doesn't seem to change without the presence of external agents such as: vehicles, animals, humans, fires and so on. For that reason, we just need to care about the appearance of external objects and don't care about the background.

To be able to separate the appearance of external objects from every frame image in video, we just need to subtract each value in the previous frame from each value in the current frame and take its absolute value. This process can be represented as follows:

Figure 11: Find the difference between two frames

From the difference between two frames, we will take each new object and proceed to launch the model to predict what they are and don't care about background.

4. Algorithms:

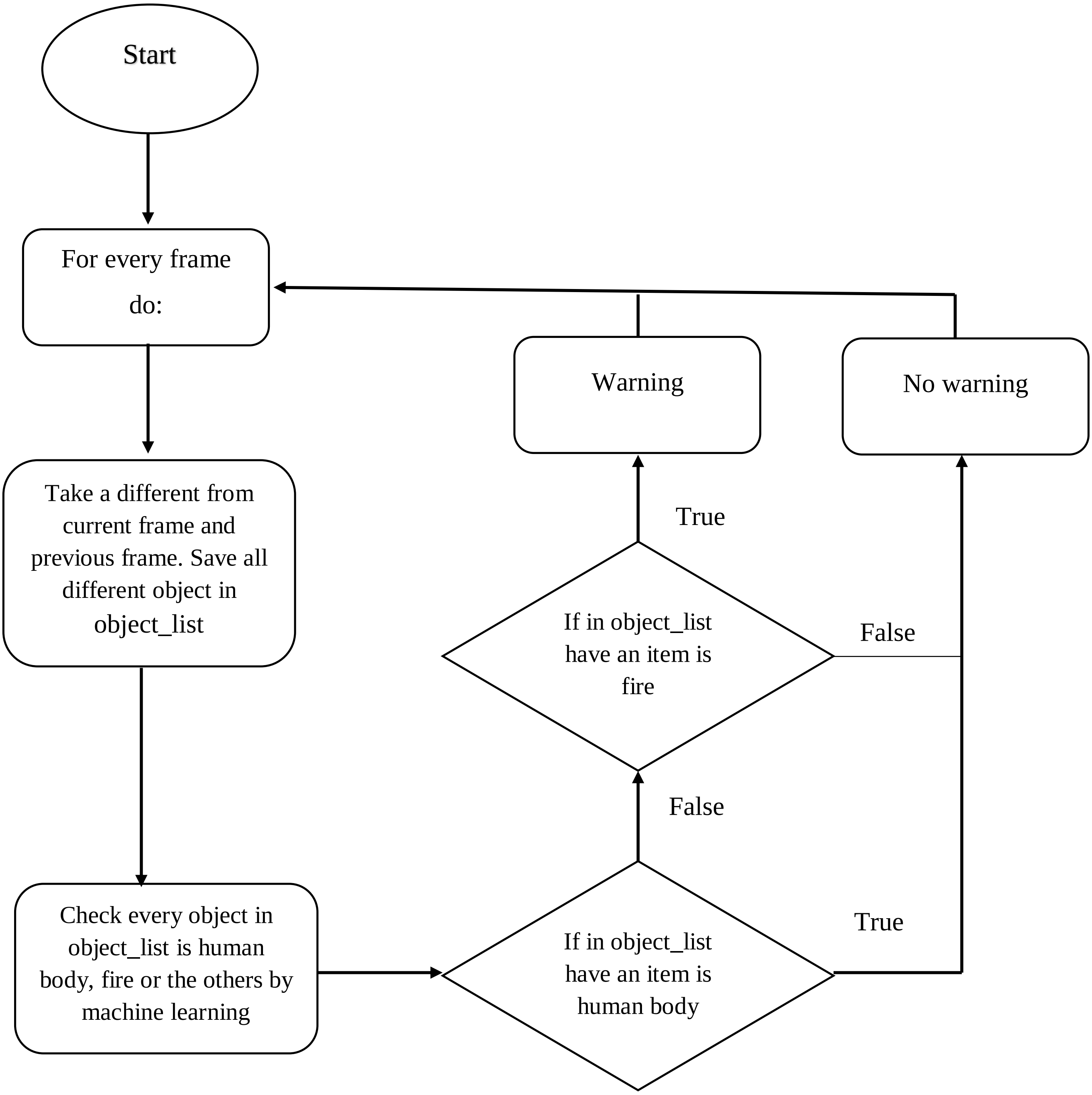

Figure 12: Algorithm Flowchart

IV. Demo and evaluation:

Link github demo: https://github.com/VinhPhamBG/CS117.L21

1. Evaluation demo:

- There are some objects that the model has not detected yet and some object that the model has detected wrong. This may cause the devices to not function properly, due to the devices being capable of false warning.
- The cause of this condition may be due to overfitting, missing data to training make the model unable to detect some objects in the frame.
- In addition, model is unable to recognize moving object in the frame and mark them as background in exchange for running time.

2. Compare with other solution:

- There are some other solutions to solve the fire detection problem, for example, fire detection based on the YOLOv2 CNN model below (https://link.springer.com/article/10.1007/s11554-020-01044-0).
- In this model, the results are very good. However, because the CNN is quite deep and the algorithm goes through all the pixels of the frame instead of focusing on the specific object, the running time can be slower. Moreover, the model still can't distinguish between dangerous fires and not dangerous fires in the frame but only focusses on detecting fires in the frame.
- Although the current model has not good results when compared to other models like model based on YOLOv2. However, the running time of the current model can be faster because it only focuses on specific object in the frame. In addition, the current model is able to distinguish between dangerous fires and not dangerous fire instead of focusing on fire detection and giving immediate warnings.

3. Discussion and development directions:

- To fix the situation that the model machine learning has bad results due to overfitting. Firstly, we need to collect more data for training. In addition, we need to adjust hyperparameter reasonably or use another model that is more optimal.
- Moreover, instead of finding objects then predicting by model machine learning, we can directly apply the YOLO algorithm to detect object better. However, the running time can be increased.
- About the practical application, when we have a good model, it can be embedded in devices that have connected to surveillance cameras, giving warning for dangerous fires that haven't been detected, thereby minimizing damage caused by fires.

V. Tasks table:

	Task	Member
1	Write introduction, problem identification, decomposition, pattern recognition, complete the report.	Vinh
2	Write decomposition, abtraction, algorithm, build demo and write evaluation	Linh

VI. References:

[1] Python cv2.absdiff() Examples

https://www.programcreek.com/python/example/89428/cv2.absdiff?fbclid=IwAR2gTHsRpEYKPoyX-nsBDn5RO61sG0Ps24B8XWUXw5RXJfKPYA4Qtdo9vxE

[2] Steveston Harbour Wharf Fire Security Camera Footage

https://www.youtube.com/watch?v=Oe573lToJQE&t=784s

[3] Electrical Fire Caught on Surveillance Video with Fire Sprinkler Activation

https://www.youtube.com/watch?v=TNEXZVKl5Vs

[4] Problem Solving Using Computational Thinking – Michagan uni

https://www.coursera.org/learn/compthinking/home/welcome