Fire and Smoke Detection

Submitted by:

Arslan Shafique	2017-Elect-118
Usman IlamDin	2017-Elect-119
Khawaja Hammad	2017-Elect-135

Supervised by: Mr. AliShafique

Department of Electrical Engineering

University of Engineering and Technology Lahore

Fire and Smoke Detection

Submitted to the faculty of the Electrical Engineering Department
of the University of Engineering and Technology Lahore
in partial fulfillment of the requirements for the Degree of

Bachelor of Science

in

Electrical Engineering.

Internal Examiner

External Examiner

Director
Undergraduate Studies

Department of Electrical Engineering

University of Engineering and Technology Lahore

Declaration

I declare that the work contained in this thesis is my own, except where explicitly stated otherwise. In addition this work has not been submitted to obtain another degree or professional qualification.

Signed: ____________________

Date: ____________________

Acknowledgments

This is acknowledg that this project is developed using Blog at Pyimagesearch

https://www.pyimagesearch.com/2019/11/18/fire-and-smoke-detection-with-keras-and-deep-learning/ by Adrian Rosebrock on November 18, 2019

Contents

List of Figures

List of Tables

Abbreviations

LAH	**L**ist **A**bbreviations **H**ere

Abstract

The efficiency of the current sensor-based fire warning device is constrained by the environmental conditions affecting the sensor. In order to address these issues, a variety of image-based fire detection systems have been developed. But such a program will produce a false alert for artifacts that are identical in nature to fire due to an algorithm that explicitly determines the characteristics of a blaze. Fire detection systems that use motion between video flames can not perform correctly as expected in an area in which the network is unstable.Throughout this report, we used CNN (Convolutional Neural Network) to introduce an image-based fire detection system. In this process, we first took an picture and then identified a fire using a qualified CNN. We also demonstrate that efficiency is dramatically enhanced relative to the detection rate observed in a video frame base test or some other application.

Chapter 1

Introduction

The increasing scale and complexity of the buildings has brought major challenges to fire safety with rapid economic development. Fast fire detection and warning with high sensitivity and precision is therefore necessary to raising fire losses. Nevertheless, conventional fire alarm systems, such as smoke and heat detectors, are not ideal for wide cities, complicated structures or spaces with multiple disruptions. Due to the limitations of the above detection technologies, missing detections, false alarms, delays in detection and other problems often occur, making early warnings even more difficult to achieve.

Image fire detection has recently become a hot research topic. The technology provides many advantages, including early fire detection, high precise deployment of the device and the ability in large space and complex building systems to effectively detect fire. In order for images to be present, they process image data from a camera through algorithms. The detection algorithm is therefore at the very heart of this technology that decide directly the fire detector output.

The framework of fire detection algorithms includes three main phases, including image preprocessing, extraction and detection of fires. The central part of algorithms is among the feature extraction. The conventional algorithm relies on the fire and machine learning classification manually picked. It is the algorithms weakness to rely on professional knowledge for manual selection of features.Although the researchers develop many studies on image characteristics of smoke and fire, only simple image characteristics, like colour. However, the extraction of small and medium-sized complex image features is difficult to differentiate between fire and fire, causing a lower accuracy and poor generalization efficiency, thanks to diverse fire forms and scenes as well as many interference events in the field.

Current neural network (CNN's) image recognition algorithms can learn and extract complex image functions automatically. This type of algorithm has caused serious concern with high visual search efficiency, automatic driving, medical diagnoses, etc. Some scientists therefore apply CNNs to fire detection and improve the self-learned algorithm into fire image array.

Although CNN-based fire detection algorithms are more prominent than traditional algorithms in complex situations in detection accuracy, there are still some issues. First, existing machine-learning algorithms were mostly categorized as image fire detection and the stage proposed for the area was ignored. The algorithms separate the entire picture into a single class. Yet smoke and flame only filled a small area of the picture at the early stage of fire. If smoke and inflammation are not apparent, the whole picture feature will decrease the accuracy of detection and delay fire detection and alarm activation without regional suggestions.Therefore, before image classification proposals regions should be defined to enhance the algorithm's capacity to detect early fire. Secondly, other scholars developed proposed regions through the manual selection of features and the classification of CNNs for pro-posal regions. This type of algorithm, which generates the proposed regions individually by computing, does not use CNNs as part of the global detection process , resulting in large numbers and a slow rate of detection. In this project we implement CNNs based fire detections model to detects the features.

Chapter 2

Motivations and Problem Statement

Appendix A

Introduction to Latex

The material provided in this appendix is taken from
http://www.sunilpatel.co.uk/thesistemplate.php

A.1 Learning LaTeX

LaTeX is not a WYSIWYG (What You See is What You Get) program, unlike word processors such as Microsoft Word or Corel WordPerfect. Instead, a document written for LaTeX is actually a simple, plain text file that contains *no formatting*. You tell LaTeX how you want the formatting in the finished document by writing in simple commands amongst the text, for example, if I want to use *italic text for emphasis*, I write the '`\emph{}`' command and put the text I want in italics in between the curly braces. This means that LaTeX is a "mark-up" language, very much like HTML.

A.1.1 A (not so short) Introduction to LaTeX

If you are new to LaTeX, there is a very good eBook – freely available online as a PDF file – called, "The Not So Short Introduction to LaTeX". The book's title is typically shortened to just "lshort". You can download the latest version (as it is occasionally updated) from here:
http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf

It is also available in several other languages. Find yours from the list on this page:
http://www.ctan.org/tex-archive/info/lshort/

It is recommended to take a little time out to learn how to use LaTeX by creating several, small 'test' documents. Making the effort now means you're not stuck learning the system when what you *really* need to be doing is writing your thesis.

A.1.2 A Short Math Guide for LaTeX

If you are writing a technical or mathematical thesis, then you may want to read the document by the AMS (American Mathematical Society) called, "A Short Math Guide for LaTeX". It can be found online here:

http://www.ams.org/tex/amslatex.html
under the "Additional Documentation" section towards the bottom of the page.

A.1.3 Common LaTeX Math Symbols

There are a multitude of mathematical symbols available for LaTeX and it would take a great effort to learn the commands for them all. The most common ones you are likely to use are shown on this page:
http://www.sunilpatel.co.uk/latexsymbols.html

You can use this page as a reference or crib sheet, the symbols are rendered as large, high quality images so you can quickly find the LaTeX command for the symbol you need.

A.1.4 Figures

There will hopefully be many figures in your thesis (that should be placed in the 'Figures' folder). The way to insert figures into your thesis is to use a code template like this:

```
\begin{figure}[htbp]
  \centering
    \includegraphics[width = 1.5in]{./Figures/uet_logo.pdf}
    \rule{35em}{0.5pt}
  \caption{The UET Laore logo.}
  \label{fig:uet_logo}
\end{figure}
```

Also look in the source file. Putting this code into the source file produces the picture of the UET logo that you can see in the figure below.

FIGURE A.1: The UET Laore logo.

Sometimes figures don't always appear where you write them in the source. The placement depends on how much space there is on the page for the figure. Sometimes there is not enough room to fit a figure directly where it should go (in relation to the text) and so LaTeX puts it at the top of the next page. Positioning figures is the job of LaTeX and so you should only worry about making them look good!

Figures usually should have labels just in case you need to refer to them (such as in figure A.1). The '\caption' command contains two parts, the first part, inside the square brackets is the title that will appear in the 'List of Figures', and so should be short. The second part in the curly brackets should contain the longer and more descriptive caption text.

The '\rule' command is optional and simply puts an aesthetic horizontal line below the image. If you do this for one image, do it for all of them.

The LaTeX Thesis Template is able to use figures that are either in the PDF or JPEG file format. It is recommended that you read this short guide on how to get the best out of figures in LaTeX, available here:
http://www.sunilpatel.co.uk/texhelp5.html

Though it is geared more towards users of Mac and OS X systems, much of the advice applies to creating and using figures in general. It also explains why the PDF file format is preferred in figures over JPEG.

A.1.5 Typesetting mathematics

If your thesis is going to contain heavy mathematical content, be sure that LaTeX will make it look beautiful, even though it won't be able to solve the equations for you.

The "Not So Short Introduction to LaTeX" (available here) should tell you everything you need to know for most cases of typesetting mathematics. If you need more information, a much more thorough mathematical guide is available from the AMS called, "A Short Math Guide to LaTeX" and can be downloaded from:
ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf

There are many different LaTeX symbols to remember, luckily you can find the most common symbols here. You can use the web page as a quick reference or crib sheet and because the symbols are grouped and rendered as high quality images (each with a downloadable PDF), finding the symbol you need is quick and easy.

You can write an equation, which is automatically given an equation number by LaTeX like this:

```
\begin{equation}
E = mc^{2}
  \label{eqn:Einstein}
\end{equation}
```

This will produce Einstein's famous energy-matter equivalence equation:

$$E = mc^2 \tag{A.1}$$

All equations you write (which are not in the middle of paragraph text) are automatically given equation numbers by LaTeX. If you don't want a particular equation numbered, just put the command, '`\nonumber`' immediately after the equation.

A.2 Sectioning and Subsectioning

You should break your thesis up into nice, bite-sized sections and subsections. LaTeX automatically builds a table of Contents by looking at all the '`\chapter{}`', '`\section{}`' and '`\subsection{}`' commands you write in the source.

The table of Contents should only list the sections to three (3) levels. A '`\chapter{}`' is level one (1). A '`\section{}`' is level two (2) and so a '`\subsection{}`' is level three (3). In your thesis it is likely that you will even use a '`\subsubsection{}`', which is level four (4). Adding all these will create an unnecessarily cluttered table of Contents and so you should use the '`\subsubsection*{}`' command instead (note the asterisk). The asterisk (*) tells LaTeX to omit listing the subsubsection in the Contents, keeping it clean and tidy.

References