VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", BELAGAVI-590018

A PROJECT REPORT ON

"STRATEGIES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS"

Submitted to Visvesvaraya Technological University, Belagavi

In partial fulfillment for the requirement for the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING

Submitted by

HARSHITHA K S	**4BW16CS405**
LAKSHMI R	**4BW16CS408**
NAVYA A S	**4BW16CS409**
NAVYA A U	**4BW16CS410**

Under the guidance of
Mr. Manu Y M
Asst. professor, Dept. of CS&E

H O D
Dept. of Computer Science & Engg.
B.G.S. Institute of Technology,
B.G. Nagar - 571 448.
Nagamangala Tq, Mandya Dist
Karnataka (INDIA)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
BGS INSTITUTE OF TECHNOLOGY, B G NAGAR
MANDYA-571448

2018-19

BGS INSTITUTE OF TECHNOLOGY

BG Nagar-571448

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled **"STRATEGIES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS"** is a bonafied work carried out by **HARSHITHA K S (4BW16CS405), LAKSHMI R (4BW16CS408), NAVYA A S (4BW16CS409) and NAVYA A U (4BW16CS410)** in partial fulfillment for the award of **Bachelor of Engineering in Computer Science & Engineering** of Visvesvaraya Technological University, Belagavi during the year 2018-2019. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements with respect to the project work prescribed by the University.

Signature of Guide:	Signature of HOD:	Signature of Principal:
Mr. Manu Y M **Asst.professor,** **Dept. of CS&E**	**Mrs. Shashikala S V** **Prof. & HOD,** **Dept. of CS&E**	**Dr. B K Narendra** **Principal** **BGSIT, BG NAGAR**

External Viva:

Name of Examiners	Signature of Examiners
1. H.J. PRASANNA KUMAR	....
2.	....

ACKNOWLEDGEMENT

The completion of the project work involves the effort of many people. We are lucky to have received a lot of help from all, during the course of this project work. And thus I take this opportunity to express my sincere thanks to all those who have guided, helped and encouraged me to emerge successfully.

We have immense pleasure in expressing our thanks to **Dr. B K Narendra, Principal,** for having supported us in academic endeavors and for providing all the facilities for the successful completion of the project.

With due respect we would like to thank **Mrs. Shashikala S V, Prof. & Head**, Department of Computer Science and Engineering, for her phenomenal support, keen interest which kept our spirit alive all through. We are thankful to our project co-ordinator **Mr. Prasanna kumar M J, Asst. Prof.,** Department of Computer Science and Engineering, for his encouragement and help throughout the duration of the project.

We express our sincere concern to our project guide **Mr. Manu Y M**, Asst. Professor, Department of Computer Science and Engineering, for his patience and valuable advice and support that has been highly instrumental in the success of the project work.

We also would like to express our gratitude towards all our teaching staff for the kind co-operation during the course of work. Finally we would like to thank all our friends, parents who have been with us with their valuable suggestion. We would like to thank all the non-teaching staff, for their kind co-operation.

HARSHITHA K S	(4BW16CS405)
LAKSHMI R	(4BW16CS408)
NAVYA A S	(4BW16CS409)
NAVYA A U	(4BW16CS410)

Shalini
H O D
Dept. of Computer Science & Engg.
B.G.S. Institute of Technology,
B.G. Nagar - 571 448
Nagamangala Tq. Mandya Dist
Karnataka (INDIA)

ABSTRACT

Recent research in video analytics promises the capability to automatically detect and extract information from video. Potential tasks include object and pedestrian detection, object and face recognition, motion detection, object tracking, as well as background subtraction and activity recognition. However, in many instances, the quality of the video from which information is to be extracted is not very high. This may be because of system constraints (like a bandwidth constraint or VHS recorder), environmental conditions (fog or low light), or a poor camera (wobbly/moving camera, limited FOV, or just a low-quality lens). In this proposed concept, we provide an overview of research on designing video analytics systems that use potentially low quality data. We consider a variety of analytics tasks, and present five categories of approaches to create quality-aware analytics: quantify the impact, predict the impact, create an analytics-aware encoder, enhance the input before analytics, and modify the analytics algorithms.

Table of Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Introduction

Vast amounts of video data render manual video analysis useless while recent automatic video analytics techniques suffer from insufficient performance. To alleviate these issues, we present a scalable and reliable approach exploiting the visual analytics methodology. This involves the user in the iterative process of exploration, hypotheses generation, and their verification. Scalability is achieved by interactive filter definitions on trajectory features extracted by the automatic computer vision stage.

We establish the interface between user and machine adopting the Video Perpetual Gram (VPG) for visualization and enable users to provide filter-based relevance feedback. Additionally, users are supported in deriving hypotheses by context-sensitive statistical graphics. To allow for reliable decision making, we gather uncertainties introduced by the computer vision step, communicate these information to users through uncertainty visualization, and grant fuzzy hypothesis formulation to interact with the machine.

The proliferation of movie and TV provides large amount of digital video data. This has led to the requirement of efficient and effective techniques for video content understanding and organization. Automatic video annotation is one of such key techniques. In this paper our focus is on annotating characters in the movie and TVs, which is called movie character identification.

The objective is to identify the faces of the characters in the video and label them with the corresponding names in the cast. The textual cues, like cast lists, scripts, subtitles and closed captions are usually exploited. In a movie, characters are the focus center of interests for the audience. Their occurrences provide lots of clues about the movie structure and content. Automatic character identification is essential for semantic movie index and retrieval, scene segmentation, summarization and other applications.

A typical processing pipeline for a video analytics system includes capture, compression, transmission, analysis, alerting, and storage. Our focus in this paper is to consider the implications of the degradations created during capture, compression, and transmission on the analytics tasks. Three core tasks are recognition, localization, and detection, each of which can be applied to a variety of objects, events, or activities. Specific objects of interest include people, faces, text, vehicles and license plates, while example actions of interest may include gestures, slip and fall, or leaving a bag. Additional tasks include object, image, semantic, and instance segmentation, scene understanding, 3D-scene reconstruction, summarization, tracking, and generic anomalous event detection.

In many cases, there are common computational steps that can be shared across a variety of analytics tasks. Foreground/background segmentation may be useful for object tracking and object detection, while image registration, object classification, tracking, and action recognition may all share the same key point extraction step. In addition, the result of a first task may determine whether a second task is performed or not. For example, if there has been nothing moving in the scene, there is little reason to perform pedestrian detection.

Regarding the fact that characters may show various appearances, the representation of character is often affected by the noise introduced by face tracking, face clustering and scene segmentation. Although extensive research efforts have been concentrated on character identification and many applications have been proposed, little work has focused on improving the robustness. We have observed in our investigations that some statistic properties are preserved in spite of these noises. Based on that, we propose a novel representation for character relationship and introduce a nameface matching method which can accommodate a certain noise.

The quality of video captured by the camera depends on the spatial and temporal resolution, any lens distortions (like fish-eye), rolling shutter, and blur. A poorly placed camera might impair the field of view of the object or action to be identified. In addition, camera motion may degrade visual quality, but may also provide valuable information for analytics. For example, a camera mounted on the body or the head may provide useful information to identify human behavior.

Sensitivity analysis is common in financial applications, risk analysis, signal processing and any area where models are developed. Good modeling practice requires that the modeler provides an evaluation of the confidence in the model, for example, assessing the uncertainties associated with the modeling process and with the outcome of the model itself. For movie character identification, sensitivity analysis offers valid tool for Characterizing the robustness has been no efforts directed character identification.

Finally, the camera's viewpoint and location may create obstructions, and an elevated camera may alter the perspective on objects within the field of view. Environmental viewing conditions can also impair the video during capture. In particular, lighting and illumination can create glare, reflections, and under-exposed video. Low light conditions can also introduce noise. Rain can cover the camera lens and create distortions, clouds may create time varying illumination issues, while fog and snow may decrease quality by reducing contrast.

Character identification, though very intuitive to humans, is a tremendously challenging task in computer vision. The reason is four-fold:

1)Weakly supervised textual cues:

There are ambiguity problem in establishing the correspondence between names and faces: ambiguity can arise from a reaction shot where the person speaking may not be shown in the frames 1; ambiguity can also arise in partially labeled frames when there are multiple speakers in the same scene 2.

Videos are often associated with additional information that could be valuable for interpretation of their content. This especially applies for the recognition of faces within video streams, where often cues such as transcripts and subtitles are available. However, this data is not completely reliable and might be ambiguously labeled. To overcome these limitations, we take advantage of semi-supervised (SSL) and multiple instance learning (MIL) and propose a new semi-supervised multiple instance learning (SSMIL) algorithm. Thus, during training we can weaken the prerequisite of knowing the label for each instance and can integrate unlabeled data, given only probabilistic information in form of priors. The benefits of the approach are demonstrated for face recognition in videos on a publicly available benchmark dataset. In fact, we show exploring new information sources can considerably improve the classification results.

2) Face identification in videos is more difficult than that in images:

Low resolution, occlusion, non rigid deformations, large motion, complex background and other uncontrolled conditions make the results of face detection and tracking unreliable. In movies, the situation is even worse. This brings inevitable noises to the character identification.

Driven by key law enforcement and commercial applications, research on face recognition from video sources has intensified in recent years. The ensuing results have demonstrated that videos possess unique properties that allow both humans and automated systems to perform recognition accurately in difficult viewing conditions. However, significant research challenges remain as most video-based applications do not allow for controlled recordings. In this survey, we categorize the research in this area and present a broad and deep review of recently proposed methods for overcoming the difficulties encountered in unconstrained settings. We also draw connections between the ways in which humans and current algorithms recognize faces. An overview of the most popular and difficult publicly available face video databases is provided to complement these discussions.

Finally, we cover key research challenges and opportunities that lie ahead for the field as a whole. Facial recognition used to recognize the faces of an individual's to identify or verify a person. Face recognition system should be able to automatically detect a face in an image. The challenges of facial recognition in the visible range include reducing the impact of variable lightening and detecting a cover or picture. Some facial recognition systems use a real time process to detect a person's head and locate the face automatically. Major benefits of facial recognitions are that it is non-intrusive, hand free, uninterrupted and accepted by most operators.

Face recognition system categorizes into two main works: verification and identification. Verification identifies a 1:1 match that compares a face image to that of template images whereas the identification identifies a 1: N match. Face recognition for a huge face database has a rate greater than 90% with well-controlled postured. Template Matching is a high straight forward machine vision technique that assign to recognize the ability of an image to that of given image pattern. It can be used in manufacturing, navigation, or to detect edges in image.

3) The same character appears quite differently during the movie:

There may be huge pose, expression and illumination variation, wearing, clothing, even makeup and hairstyle changes. Moreover, characters in some movies go through different age stages, e.g., from youth to the old age. Sometimes, there will even be different actors playing different ages of the same character.

Face track clustering serves as an important step in movie character identification. In most of the existing methods, some cues are utilized to determine the number of target clusters prior to face clustering, e.g., in, the number of clusters is the same as the number of distinct speakers appearing in the script. While this seems convinced at first glance, it is rigid and even deteriorating the clustering results sometimes. In this paper, we lose the restriction of one face cluster corresponding to one character name.

Automatic video understanding has become one of the most essential and demanding challenges and research directions. The problems that span from this field, such as activity recognition, saliency and scene analysis, comprise detecting events and extracting high level semantics in realistic video sequences. So far, the majority of the methods designed for these tasks deal with visual data ignoring the presence of other modalities, such as text and sound. Nonetheless, the exploitation of the information they provide can lead to better understanding of the underlying semantics. In addition, most of these techniques are fully supervised and are trained on diverse and usually large-scale datasets.

4) The determination for the number of identical faces is not trivial:

Due to the remarkable intra-class variance, the same character name will correspond to faces of huge variant appearances. It will be unreasonable to set the number of identical faces just according to the number of characters in the cast. Our study is motivated by these challenges and aims to find solutions for a robust framework for movie character identification.

Humans are the typical end recipients of images and video footage. Recently, however, various systems and applications started to rely on video analysis algorithms to automate their tasks. Such systems include video surveillance, autonomous vehicles, and applications running on small wireless mobile devices. For example, a video surveillance system can automatically analyze video, without alerting the human guard, until a suspicious event occurs; an autonomous aircraft or vehicle can detect, track, and follow a target based on results of the video analysis; a mobile phone with camera used for automated tagging of friends on a photo or identification of the current location based on analysis of a landmark captured in the picture.

Face detection is defined as the process of extracting faces from scenes. So, the system positively identifies a certain image region as a face. There is a standard image input format, so there is no need for a detection step. An example of this could be a criminal data base. There, the law enforcement agency stores faces of people with a criminal report. If there is new subject and the police has his or her passport photograph, face detection is not necessary. However, the conventional input image of computer vision systems are not that suitable. They can contain many items or faces. In these cases face detection is mandatory. It's also unavoidable if we want to develop an automated face tracking system. For example, video surveillance systems try to include face detection, tracking and recognizing. So, it's reasonable to assume face detection as part of the more ample face recognition problem. Face detection must deal with several well known challenges. They are usually present in images captured in uncontrolled environments, such as surveillance video systems.

1.2 Objectives

A typical processing pipeline for a video analytics system includes capture, compression, transmission, analysis, alerting, and storage.

Video degradations are commonplace in deployed systems. Some degradation may happen at the moment of capture, i.e., in the camera, while others may be "self-inflicted", introduce prior to the analysis stage because of constraints within the system.

The proliferation of movie and TV provides large amount of digital video data. This has led to the requirement of efficient and effective techniques for video content understanding and organization. Automatic video annotation is one of such key techniques. In this paper our focus is on annotating characters in the movie and TVs, which is called movie character. The objective is to identify the faces of the label marks in the technical faces to compare to multiple detection of images in videos. In analysis to train the images in system database can be stored and retrieved.

Character identification, though very intuitive to humans, is a tremendously challenging task in computer vision.

The reason is four-fold:

1) Weakly supervised textual cues.

2) Face identification in videos is more difficult than that in images.

3) The same character appears quite differently during the movie.

4) The determination for the number of identical faces is not trivial.

Our study is motivated by these challenges and aims to find solutions for a robust framework for movie character identification the faces detected from the video and the names extracted from the movie script. These methods only utilize the case list textual resource. In the "cast list discovery" problem, faces are clustered by appearance and faces of a particular character are expected to be collected in a few pure clusters. Names for the clusters are then manually selected from the cast list.

The prior work on the topic of video analytics using degraded quality images or videos can be subdivided into 5 basic categories.

- Quantify the impact.
- Predict the impact.
- Create an analytics-aware encoder for video or features.
- Enhance the input before analytics.
- Modify the analytics algorithm.

1.3 Problem Statement

In this project is used to detect the face of movie characters and recognize the characters and the existing system is taking the too much time to detect the face. But this one we can do it in a minute process.

Face recognition is important for the interpretation of facial expressions in applications such as intelligent man-machine interface and communication, intelligent visual surveillance, teleconference and real-time animation from live motion images. There have been many computer models proposed for machine-based recognition of face images. Among them, the eigenface approach, initially introduced by Sirovich and Kirby for face image coding and then adopted by Turk and Pentland for classification, extracts principal local and global hidden "face stimuli" that are significant in recognition.

The distinctive features of this method are: the eigenvectors reflect the statistical properties of face images they represent; they capture more global "signatures" of the faces and therefore, more tolerant and immune to local variations. Because face recognition is commonly subject to a wide range of changes in viewing angle and facial hair as well as to partial occlusion and blurring, the eigenface method is computationally more robust and biologically more plausible than other template matching techniques that are based on the detection of visible local facial features and the representation of face models by geometric measures of such features, for example the location and size of eyes, nose and mouth as well as their distance. Eigenfaces also provide an attractive mechanism for the transmission of coded image sequences through networks.

Another challenge faced in the detection and tracking process is occlusion. There are many types of occlusion that can occur in a real-time scenario. Occlusion between pedestrians, occlusion between pedestrians and buildings, occlusion between pedestrians and vehicles are the common types of occlusions faced in a real-time scenario. In an overcrowded situation, all these occlusions may affect the accuracy of the algorithm.

In addition, the problem becomes more complex due to illumination changes in the scene. Different lighting conditions may affect the visibility of an object and even alter the appearance of the object. Hence the way lights are placed in a scene, may cause an object to look different. In our case, pedestrians may look different due to the lighting conditions in the scene.

Chapter 2

LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength.. Before building the system the above consideration are taken into account for developing the proposed system.

1. Real-time Video Analytics: The Killer app for Edge Computing

Authors: Ganesh Ananthanarayanan, ParamvirBahl, MatthaiPhilipose

Year: 2018

Description:

According to a 2015 report by the Information Handling Services on the installed base for video surveillance equipment, there is a camera installed for every 29 people on the planet, with mature markets having a camera for every 8 people. The report predicts that the number of cameras will grow by 20 percent year over year for the next 5 years.

Video analytics from these cameras are used for traffic control, surveillance, and security in both public and private venues, as well as consumer applications including digital assistants for real-time decisions. We propose that a geographically distributed architecture of public clouds, private clusters, and edges that extend down to the cameras is the only approach that can meet the strict real-time requirements of large-scale video analytics, which must address latency, bandwidth, and provisioning challenges.

Video analytics is central to a wide range of future and existing applications ranging from surveillance and self-driving cars, to personal digital assistants and even drone cameras.

Advantages:

- Real time requirements of large-scale video analytics.

Disadvantages:

- Low latency.
- Bandwidth.

2. Video Analysis of Traffic Accidents based on Projection Extreme Learning Machine

Authors: XinmanZhanga, Longbin Lua, XuebinXub, ShuanglingYuea

Year:2017

Description:

Recently, increasing attentions have been concentrated upon the judgment of video-based traffic accidents, which serves as a crucial technology of intelligent transportation system (ITS). This paper introduces a novel vehicle tracking method based on projection extreme learning machine (PELM) by taking consideration of poor performance in the dealing with high-dimensional and low-samples.

Firstly, samples near the target in the first frame of a video are employed as training samples to the PELM classifier. The algorithm uses projection vectors instead of random assignment as the weights of the input layer. Then the PELM classifier is trained with the samples near the target in the following frames to update parameters and obtain the target location with the maximum response. Finally, the fitting error and maximum-range trajectory are adopted to judge whether the vehicle is collided or not. Simulation experiments have shown the proposed method can achieve quite satisfying results.

It has two main challenges, high road and vehicle detection accuracy and real-time performance. To study the two problems, we developed a driving simulation platform in a virtual scene. In this paper, as the first step of final solution, the Extreme Learning Machine (ELM) has been used to detect the virtual roads and vehicles. The Support Vector Machine (SVM) and Back Propagation (BP) network have been used as benchmark. Our experimental results show that the ELM has the fastest performance on road segmentation and vehicle detection with the similar accuracy compared with other techniques.

Advantages:

- Easy to install.
- High-dimensional.

Disadvantages:

- Poor performance.
- Low samples.

3. Demo Abstract: EVAPS: Edge Video Analysis for Public Safety

Authors: Qingyang Zhang, Zhifeng YU, Weisong SHI

Year: 2016

Description:

Real time video analysis at the edge of the network is very promising to significantly improve public safety, e.g., dangerous accidents detection and find a missing person. Simply uploading the video stream to the cloud for analysis costs too much energy and network bandwidth to an energy-limited camera. Hence we propose EVAPS, which distributes the computing workload in both the edge nodes and the cloud in an optimized way.

EVPAS is able to eliminate unnecessary data transmission over and save energy for edge devices, i.e. cameras. Three demos are used to illustrate the energy efficiency and optimized solution for public safety using the proposed EVAPS framework.Edge video analysis is the key to solve the problems stated above. In edge computing, computing should happen at the proximity of data sources. One remark able feature of real time video analysis proposed here is that the processed video is much smaller than the source raw video in term of size yet without missing any significant object and event, especially after a face detection or other recognition processing.

Which distributes the computing workload in both the edge nodes and the cloud in an optimized way. EVPAS is able to eliminate unnecessary data transmission over and save energy for edge devices, i.e., cameras. Three demos are used to illustrate the energy efficiency and optimized solution for public safety using the proposed EVAPS framework.

Advantages:

- Real time performance.
- Object detection.
- Reduced cost.

Disadvantages:

- Network bandwidth to an energy-limited camera.

4. On a Relation between Graphs edit distance and maximum common Sub graph.

Authors: H. Bunke

Year:2016

Description:

In approximate, or error-correcting, graph matching one considers a set of graph edit operations, and defines the edit distance of two graphs g1 and g2 as the shortest (or least cost) sequence of edit operations that transform g1 into g2. A maximum common sub graph of two graphs g1 and g2 is a sub graph of both g1 and g2 such that there is no other sub graph of g1 and g2 with more nodes.

Graph edit distance and maximum common sub graph are well known concepts that have various applications in pattern recognition and machine vision. In this paper a particular cost function for graph edit distance is introduced, and it is shown that under this cost function graph edit distance computation is equivalent to the maximum common sub graph problem.

1)Latency: Applications require processing the video at very low latency because the output of the analytics is used to interact with humans or to actuate some other system.

2) Bandwidth: High-definition video requires large bandwidth (5Mbps or even 25Mbps for 4K video) and streaming large number of video feeds directly to the cloud might be infeasible. When cameras are connected wirelessly, such as inside a car, the available uplink bandwidth is very limited.

3) Provisioning: Using compute at the cameras allows for correspondingly lower provisioning in the cloud. Also, uninteresting parts of the video can be filtered out.

Advantages:

- Error-correction.
- Reduced cost.

Disadvantages:

- A maximum common sub graph of two graphs g1 and g2 is a sub graph of both g1 and g2 such that there is no other sub graph of g1 and g2 with more nodes.

5. Multiple instance learning for labeling faces in broadcasting news video

Authors: Jun Yang

Year:2015

Description:

Labeling faces in news video with their names is an interesting research problem which was previously solved using supervised methods that demand significant user efforts on labeling training data. In this paper, we investigate a more challenging setting of the problem where there is no complete information on data labels.

Specifically, by exploiting the uniqueness of a face's name, we formulate the problem as a special multi-instance learning (MIL) problem, namely exclusive MIL or Emil problem, so that it can be tackled by a model trained with partial labeling information as the anonymity judgment of faces, which requires less user effort to collect.

We propose two discriminative probabilistic learning methods named Exclusive Density (ED) and Iterative ED for Emil problems. Experiments on the face labeling problem shows that the performance of the proposed approaches are superior to the traditional MIL algorithms and close to the performance achieved by supervised methods trained with complete data labels.

Multiple Instance Learning (MIL) is a class of learning algorithms for handling problems with only partial label information expressed as the labels on bags of instances. In the MIL setting, unlabeled instances are grouped into a set of bags, and each bag is assigned a binary label which has a logical-or relationship with the instance labels.

Advantages:

- It labeling faces in news video with their names.

Disadvantages:

- There is no complete information on data labels.
- High performance.

6. Character-based Movie Summarization

Authors: J. Sang and C. Xu

Year:2014

Description:

A decent movie summary is helpful for movie producer to promote the movie as well as audience to capture the theme of the movie before watching the whole movie. Most exiting automatic movie summarization approaches heavily rely on video content only, which may not deliver ideal result due to the semantic gap between computers calculated low-level features and human used high-level understanding.

In this paper, we incorporate script into movie analysis and propose a novel character-based movie summarization approach, which is validated by modern film theory that what actually catches audiences' attention is the character. We first segment scenes in the movie by analysis and alignment of script and movie.

Then we conduct sub story discovery and content attention analysis based on the scent analysis and character interaction features. Given obtained movie structure and content attention value, we calculate movie attraction scores at both shot and scene levels and adopt this as criterion to generate movie summary. The promising experimental results demonstrate that character analysis is effective for movie summarization and movie content understanding.

According to modern film theory, "All films are about nothing - nothing but character", which reveals characters are important for movie summarization. From audiences' perspective, movie is attractive and catches their attention because they want to know about the story of characters.

Advantages:

- It is effective for movie summarization and movie content understanding.

Disadvantages:

- It takes more time because we first segment scenes in the movie by analysis and alignment of script and movie.Then we conduct sub story discovery based on the scent analysis and character interaction features.

7. TV parser: An automatic TV video parsing method

Authors: C. Liang, C. Xu, J. Cheng, and H.Lu

Year:2013

Description:

In this paper, we propose an automatic approach to simultaneously name faces and discover scenes in TV shows. We follow the multi-modal idea of utilizing script to assist video content understanding, but without using timestamp (provided by script-subtitles alignment) as the connection. Instead, the temporal relation between faces in the video and names in the script is investigated in our approach, and an global optimal video-script alignment is inferred according to the character correspondence. The flourishing TV industries have contributed to an explosive growth of video content. However, such increasing quantity has not yet been accompanied by an improvement in its accessibility. Huge amount of TV videos have become a burden of storage and management.

The contribution of this paper is two-fold:

(1) we propose a generative model, named TV Parser, to depict the temporal character correspondence between video and script, from which face-name relationship can be automatically learned as a model parameter, and meanwhile, video scene structure can be effectively inferred as a hidden state sequence;

(2) we find fast algorithms to accelerate both model parameter learning and state inference, resulting in an efficient and global optimal alignment. We conduct extensive comparative experiments on popular TV series and report comparable and even superior performance over existing methods.

Advantages:

- It is simultaneously display the name of the face and discover scenes in TV shows.

Disadvantages:

- It is complex.

8. Taking the bite out of Automated Naming of Characters in TV Video

Authors: M. Everingham, J. Sivic, and A. Zissserman

Year:2012

Description:

We investigate the problem of automatically labeling appearances of characters in TV or film material with their names. This is tremendously challenging due to the huge variation in imaged appearance of each character and the weakness and ambiguity of available annotation. However, we demonstrate that high precision can be achieved by combining multiple sources of information, both visual and textual.

The principal novelties that we introduce are:

(i) Automatic generation of time stamped character annotation by aligning subtitles and transcripts;

(ii) Strengthening the supervisory information by identifying when characters are speaking.

In addition, we incorporate complementary cues of face matching and clothing matching to propose common annotations for face tracks, and consider choices of classifier which can potentially correct errors made in the automatic extraction of training data from the weak textual annotation. Results are presented on episodes of the TV series "Buffy the Vampire Slayer".

Advantages:

- It is automatically labeling appearances of characters in TV or film material with their names.

Disadvantages:

- It is difficult because huge variation in imaged appearance of each character and the weakness and ambiguity of available annotation.

9. Character Identification in feature-length films using Global Face-name Matching

Authors: Y. Zhang, C. Xu, H. Lu, and Y. Huang

Year:2012

Description:

Identification of characters in films, although very intuitive to humans, still poses a significant challenge to computer methods. Different from the state-of-the-art methods on naming faces in the videos, most of which used the local matching between a visible face and one of the names extracted from the temporally local video transcript, we attempt to do a global matching between names and clustered face tracks under the circumstances that there are not enough local name cues that can be found. The objective of this work is to label television or movie footage with the names of the people present in each frame of the video. As has previously been noted such material is extremely challenging visually as characters exhibit significant variation in their imaged appearance due to changes in scale, pose, lighting, expressions, hair style, etc. There are additional problems of poor image quality and motion blur.

The contributions of our work include:

1) A graph matching method is utilized to build face-name association between a face affinity network and a name affinity network which are, respectively, derived from their own domains (video and script).

2) An effective measure of face track distance is presented for face track clustering.

3) As an application, the relationship between characters is mined using social network analysis. The proposed framework is able to create a new experience on character-centered film browsing. Experiments are conducted on ten feature-length films and give encouraging results.

Advantages:

- An effective measure of face track distance is presented

Disadvantages:

- It is difficult to identifying characters in feature-length films using video and film script.

10. Robust Movie Character Identification and The Sensitivity Analysis

Authors:J. Sang, C. Liang, C. Xu, and J. Cheng

Year: 2011

Description:

Automatic face identification of characters in movies has drawn significant research interests and led to various applications. It is a challenging problem due to the huge variation in the appearance of each character.

In this paper we present a robust character identification approach by incorporating a noise insensitive relationship representation and a graph matching algorithm. Beyond existing character identification approaches, we further perform explicit sensitivity analysis on character identification.

Automatic character identification in movies is essential for semantic movie analysis such as movie indexing, summarization and retrieval. Character identification, though very intuitive to humans, is a tremendously challenging task in computer vision. This is due to the introduced noises of huge variation in appearance of characters, such as scale, pose, illumination, expression and wearing. The objective of this work is to explicitly consider the unavoidable noises and analyze the noise impact.

Advantages:

- We present a robust character identification approach by incorporating a noise insensitive relationship representation.
- Experiments validate.

Disadvantages:

- It is limited in complex movie scenes due to the noises generated during the face tracking and face clustering process.

Chapter 3

ANALYSIS

3.1 System Analysis

3.1.1 System

A system is an orderly group of interdependent components linked together according to a plan to achieve a specific objective. Its main characteristics are organization, interaction, interdependence, integration and a central objective.

3.1.2 System Analysis

System analysis and design are the application of the system approach to problem solving generally using computers. To reconstruct a system the analyst must consider its elements output and inputs, processors, controls feedback and environment.

3.1.3Analysis

Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. One aspect of analysis is defining the boundaries of the system and determining whether or not a candidate system should consider other related systems. During analysis data are collected on the available files decision points and transactions handled by the present system. This involves gathering information and using structured tools for analysis.

3.2 Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- Economical feasibility
- Technical feasibility
- Social feasibility

3.2.1 Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

3.2.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

3.2.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.3 Existing System

In this project is used to detect the face of movie characters and recognize the characters and theexisting system is taking the too much time to detect the face. But this one we can do it in a minute process.

Disadvantages:

1. In the previous process the time taken for detecting face is too long in windows processed.

3.4 Proposed System

In this Robust Face-Name Graph Matching for Movie Character Identification is used to detect the face of movie characters and the Proposed system is taking the minimum time to detect the face. In this one we can do it in a minute process.

Advantages:

1. In the proposed process the time taken for detecting face in minimum(min) time only in windows processed.

3.5 Applications

1. Banks and Credit unions-Suspicious person detection :

There are some ways to detect suspicious person are:

- Monitor Access to Sensitive Areas with Real-Time Alerts

Using video analytics, financial institutions can monitor specific areas of interests in their branches and trigger an alert when a physical event – like a safe opening – takes place. Security staff can set up automated reports detailing each time the safe was opened and check the corresponding video to see what took place.

- Safeguard Surveillance Cameras with Active Tampering Alarms

Fraudsters often try to cover their tracks by dismantling or blocking the view of a security camera in advance of a crime. With intelligent video, banks can be immediately notified if something is obstructing their camera's field of view. This could be as innocent as an employee stacking inventory too high and inadvertently blocking the camera, or in outdoor situations, a tree branch that's blown over in front of the camera. Analytics can help ensure your video is protected by generating real-time alerts if your cameras have been blocked or moved.

- Detect the Installation of ATM Skimming Devices

Video-based business intelligence solutions can help banks combat the installation of skimming devices by detecting when someone is standing at an ATM for an extended period of time, but not making a transaction. Banks can receive alerts in real-time, so they can quickly investigate, remove a skimming device if one has been installed, and proactively alert customers who may have been impacted.

- Find Check & Identity Fraud

A solution that integrates clear surveillance video with transaction data and highly-accurate analytics can also help uncover identity fraud. Video can be used to identify imposters, but it can also help ensure banking staff are following protocol by always asking customers for proper identification.

2. Stadiums- Suspicious person detection:

Video analytics helps stadium security in spotting suspicious person and also people who are wearing the wrong credential badge for the areas and sending staff to detain or remove them. In fact, some stadiums used their video systems' sophisticated search algorithms to locate missing children so that be able to reunite them with their parents. In addition, camera operators sweep the area continuously for any activity outside the norm, therefore, search algorithms can also help them spot lost objects and suspicious packages then trace them back to whom left them.

3. Railway /Metro stations:

Platform surveillance monitoring system using image processing technology for passenger safety in railway station. The system monitors almost entire length of the track line in the platform by using multiple cameras, and determines in real-time whether a human or dangerous obstacle is in the preset monitoring area by using image processing technology. According to the experimental results, we verity system performance in real condition. Detection of train state and object is conducted robustly by using proposed image processing algorithm. Moreover, to deal with the accident immediately, the system provides local station, central control room and train with the video information and alarm message.

4. Parking management:

Parking has been a real issue in the urban areas as it can be difficult for many to find parking spots in public places, especially malls and retail stores. This also leads to higher fuel consumption and recurring problems with traffic congestion. video analytics features are implemented in the smart parking system, firstly it enables constant monitoring of all the vehicles present at a parking facility. Any suspicious activity can be tracked and appropriate actions can be taken before the person enters the premises of the store.

5. Vehicle monitoring on roads:

Today most of the vehicles are equipped with onboard automotive driver assistance system aiming to alert drivers about driving environments, and possible collision with other vehicles has attracted a lot of attention lately. To avoid this we presents a review of recent vision based on-road vehicle detection systems. Our focus is on systems where the camera is mounted on the vehicle and also being fixed such as in traffic/drive way monitoring systems. First, we discuss the problem of on-road vehicle detection using optical sensors followed by a brief review of intelligent vehicle research worldwide. Then, we discuss active and passive sensors to set the stage for vision based vehicle detection. Methods aiming to quickly hypothesize the location of vehicles in an image as well as to verify the hypothesized locations are reviewed next. Tracking of vehicle is also reviewed to illustrate the benefits of exploiting temporal continuity for vehicle detection.

3.6 System Requirement Specification

A System Requirements Specification is a finished depiction of the conduct of the structure to be made. It intertwines a strategy of utilization cases that portray every one of the joint endeavors the client will have with the product. In spite of utilization cases, the SRS in like way contains nonfunctional necessities.

3.6.1 Hardware Requirements:

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 512 Mb.

3.6.2 Software Requirements:

- Operating system : Windows XP.
- Coding Language : C#.net
- Data Base : SQL Server 2005

Chapter 4

SYSTEM DESIGN

4.1 Features of . Net

Microsoft .NET is a set of Microsoft software technologies for rapidly building and integrating XML Web services, Microsoft Windows-based applications, and Web solutions. The .NET Framework is a language-neutral platform for writing programs that can easily and securely interoperate. There's no language barrier with .NET: there are numerous languages available to the developer including Managed C++, C#, Visual Basic and Java Script.

The .NET framework provides the foundation for components to interact seamlessly, whether locally or remotely on different platforms. It standardizes common data types and communications protocols so that components created in different languages can easily interoperate.

The .NET is a software framework. It is developed by Microsoft. It includes a large library and also provides language inter-operability across some programming languages. Language inter-operability refers the capability of two different languages to interact and operate on the same kind of data structures.

".NET" is also the collective name given to various software components built upon the .NET platform. These will be both products (Visual Studio.NET and Windows.NET Server, for instance) and services (like Passport, .NET My Services, and so on).

The programs written for .NET execute in a software environment. The name of the software environment is Common Language Runtime (CLR). It is the virtual machine component. The compiled code is converted into machine code at first. Then it is executed by computer's CPU. The CLR provides additional services like exception handling, memory management, type safety, garbage collection, thread management etc.

The .NET Framework's Base Class Library offers user interface, database connectivity, data access, cryptography, web application development, numeric algorithms, network communications etc. Programmers produce software by combining their own source code with the .NET Framework and other libraries. Microsoft developed the .Net FRAMEWORK, which runs predominantly on Microsoft windows.

The .NET Framework is projected to be used by most new applications created for the Windows platform. Microsoft also produces an integrated largely for .NET software called Visual Studio.

4.2 The .Net Framework

The .NET Framework has two main parts:

1. The Common Language Runtime (CLR).
2. A hierarchical set of class libraries.

The CLR is described as the "execution engine" of .NET. It provides the environment within which programs run. The most important features are

- Interoperability
- Common Language Runtime engine (CLR)
- Language independence
- Base Class Library
- Simplified deployment
- Security
- Portability
- Conversion from a low-level assembler-style language, called Intermediate Language (IL), into code native to the platform being executed on.
- Memory management, notably including garbage collection.
- Checking and enforcing security restrictions on the running code.
- Loading and executing programs, with version control and other such features.
- The following features of the .NET framework are also worth description:

4.2.1 Managed Code

The code that targets .NET, and which contains certain extraInformation - "metadata" - to describe itself. Whilst both managed and unmanaged code can run in the runtime, only managed code contains the information that allows the CLR to guarantee, for instance, safe execution and interoperability.

Any language that is written in .NET Framework is managed code. . The code, which is developed outside .NET, Framework is known as unmanaged code.

4.2.2 Managed Data

With Managed Code comes Managed Data. CLR provides memory allocation and Deal location facilities, and garbage collection. Some .NET languages use Managed Data by default, such as C#, Visual Basic.NET and JScript.NET, whereas others, namely C++, do not. Targeting CLR can, depending on the language you're using, impose certain constraints on the features available. As with managed and unmanaged code, one can have both managed and unmanaged data in .NET applications - data that doesn't get garbage collected but instead is looked after by unmanaged code.

4.2.3 Common Type System

The CLR uses something called the Common Type System (CTS) to strictly enforce type-safety. This ensures that all classes are compatible with each other, by describing types in a common way.

CTS define how types work within the runtime, which enables types in one language to interoperate with types in another language, including cross-language exception handling. As well as ensuring that types are only used in appropriate ways, the runtime also ensures that code doesn't attempt to access memory that hasn't been allocated to it.

4.2.4 Common Language Specification

The CLR provides built-in support for language interoperability. To ensure that you can develop managed code that can be fully used by developers using any programming language, a set of language features and rules for using them called the Common Language Specification (CLS) has been defined. Components that follow these rules and expose only CLS features are considered CLS-compliant.

Common Language Specification. A Common Language Specification (CLS) is a document that says how computer programs can be turned into Common Intermediate Language (CIL) code. When several languages use the same byte code, different parts of a program can be written in different languages.

4.3 The Class Library

.NET provides a single-rooted hierarchy of classes, containing over 7000 types. The root of the namespace is called System; this contains basic types like Byte, Double, Boolean, and String, as well as Object. All objects derive from System.

Object. As well as objects, there are value types. Value types can be allocated on the stack, which can provide useful flexibility. There are also efficient means of converting value types to object types if and when necessary.

The set of classes is pretty comprehensive, providing collections, file, screen, and network I/O, threading, and so on, as well as XML and database connectivity. The class library is subdivided into a number of sets (or namespaces), each providing distinct areas of functionality, with dependencies between the namespaces kept to a minimum.

4.4 Languages Supported By .Net

The multi-language capability of the .NET Framework and Visual Studio .NET enables developers to use their existing programming skills to build all types of applications and XML Web services. The .NET framework supports new versions of Microsoft's old favorites Visual Basic and C++ (as VB.NET and Managed C++), but there are also a number of new additions to the family.

Visual Basic .NET has been updated to include many new and improved language features that make it a powerful object-oriented programming language. These features include inheritance, interfaces, and overloading, among others. Visual Basic also now supports structured exception handling, custom attributes and also supports multi-threading.

Visual Basic .NET is also CLS compliant, which means that any CLS-compliant language can use the classes, objects, and components you create in Visual Basic .NET.

Managed Extensions for C++ and attributed programming are just some of the enhancements made to the C++ language. Managed Extensions simplify the task of migrating existing C++ applications to the new .NET Framework.

C# is Microsoft's new language. It's a C-style language that is essentially "C++ for Rapid Application Development". Unlike other languages, its specification is just the grammar of the language. It has no standard library of its own, and instead has been designed with the intention of using the .NET libraries as its own.

Microsoft Visual J# .NET provides the easiest transition for Java-language developers into the world of XML Web Services and dramatically improves the interoperability of Java-language programs with existing software written in a variety of other programming languages.

Active State has created Visual Perl and Visual Python, which enable .NET-aware applications to be built in either Perl or Python. Both products can be integrated into the Visual Studio .NET environment. Visual Perl includes support for Active State's Perl Dev Kit.

The .NET Framework was developed so that it could support a theoretically infinite number of development languages. Currently, more than 20 development languages work with the .NET Framework.

C# is the programming language specifically designed for the .NET platform, but C++ and Visual Basic have also been upgraded to fully support the .NET framework.

Other languages for which .NET compilers are available include

- Fortran
- Cobol
- Eiffel

ASP.NET XMLWEB SERVICES	Windows Forms
Base Class Libraries	
Common Language Runtime	
Operating System	

Fig. 4.1 .Net Framework

C#.NET is also compliant with CLS (Common Language Specification) and supports structured exception handling. CLS is set of rules and constructs that are supported by the CLR (Common Language Runtime). CLR is the runtime environment provided by the .NET Framework; it manages the execution of the code and also makes the development process easier by providing services.

C#.NET is a CLS-compliant language. Any objects, classes, or components that created in C#.NET can be used in any other CLS-compliant language. In addition, we can use objects, classes, and components created in other CLS-compliant languages in C#.NET .The use of CLS ensures complete interoperability among applications, regardless of the languages used to create the application.

4.5 The .Net Framework

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

The .NET Framework was developed so that it could support a theoretically infinite number of development languages. Currently, more than 20 development languages work with the .NET Framework.

4.5.1 Objectives Of. Net Framework

1. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.

2. To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.

3. Eliminates the performance problems.

There are different types of application, such as Windows-based applications and Web-based applications.

- Platform Independent.
- Language Independent.
- Language Integration.
- OOPs Concepts.
- Supports to develop GUI based applications.
- Supports to develop automated Background Processes with the help of Windows Services.
- Supports to develop N-Tier architecture with the help of Distributed programming

4.5.2 Emgu cv

Emgu CV is a cross platform .Net wrapper to the OpenCV image processing library. Allowing OpenCV functions to be called from .NET compatible languages such as C#, VB, VC++, IronPython etc. The wrapper can be compiled in Mono and run on Windows, Linux, Mac OS X, iPhone, iPad and Android devices.

4.5.3 Overview of the .NET Framework

The .NET Framework is a technology that supports building and running the next generation of apps and XML Web services. The .NET Framework consists of the common language runtime (CLR) and the .NET Framework class library. The common language runtime is the foundation of the .NET Framework.

The .NET Framework is designed to fulfill the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.
- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.

- To make the developer experience consistent across widely varying types of apps, such as Windows-based apps and Web-based apps.
- To build all communication on industry standards to ensure that code based on the .NET Framework integrates with any other code.

4.6 System Architecture

The goal of video analysis is to gain insight into the data provided by the capturing device. According to VA, insight is gained by a user-driven iterative process of exploration, hypothesis formulation, and hypothesis verification. The users are supported in this task by features automatically extracted from the input videos. We use trajectory features since they are suitable in many cases for visual surveillance and scientific video analysis.

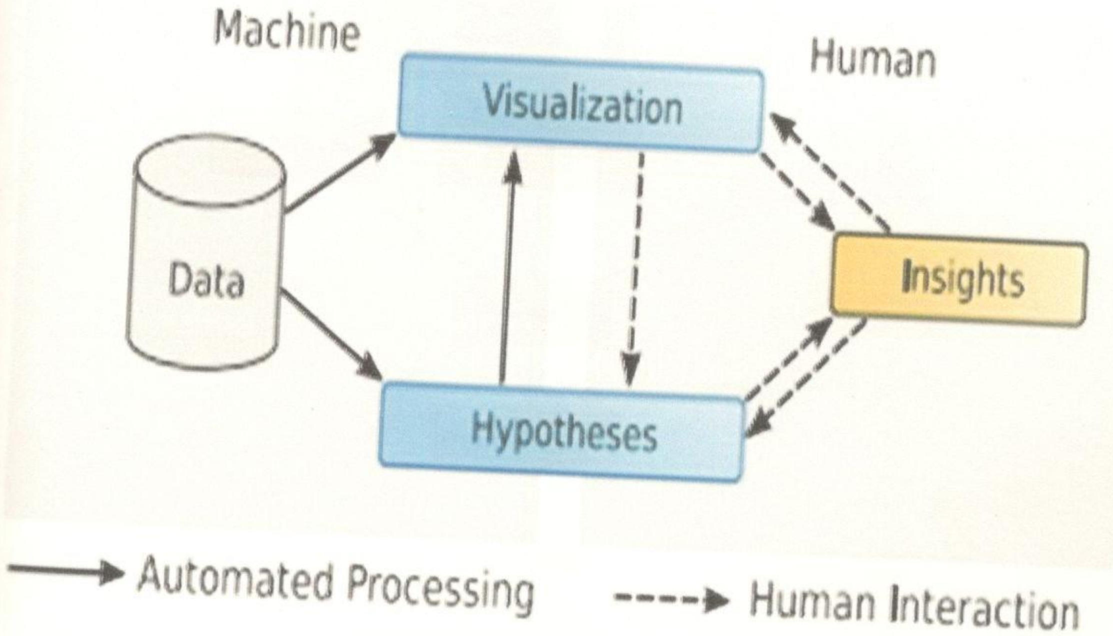

Fig. 4.2 Simplified visual analytics process

Fig. 3.1 shows the simplified visual analytics process. The interfaces between machine and human (i.e., visualization and human-computer interaction) are fundamental parts to combine automatic low-level feature extraction and high-level pattern recognition by the user hypotheses, assumptions, and finally, insights on possibly relevant events by means of the video data and the provided features. A real-life system that employs image processing, such as a surveillance system, will need to do much more than image processing.

Such a system should collect data from video feeds, apply video processing, apply business logic, manage business logic, let the user see the interesting data through a dashboard, generate and manage alerts, and manage equipment providing video feeds. Figure depicts a reference architecture that shows how to build such a system.

4.7 Data Flow Diagram

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

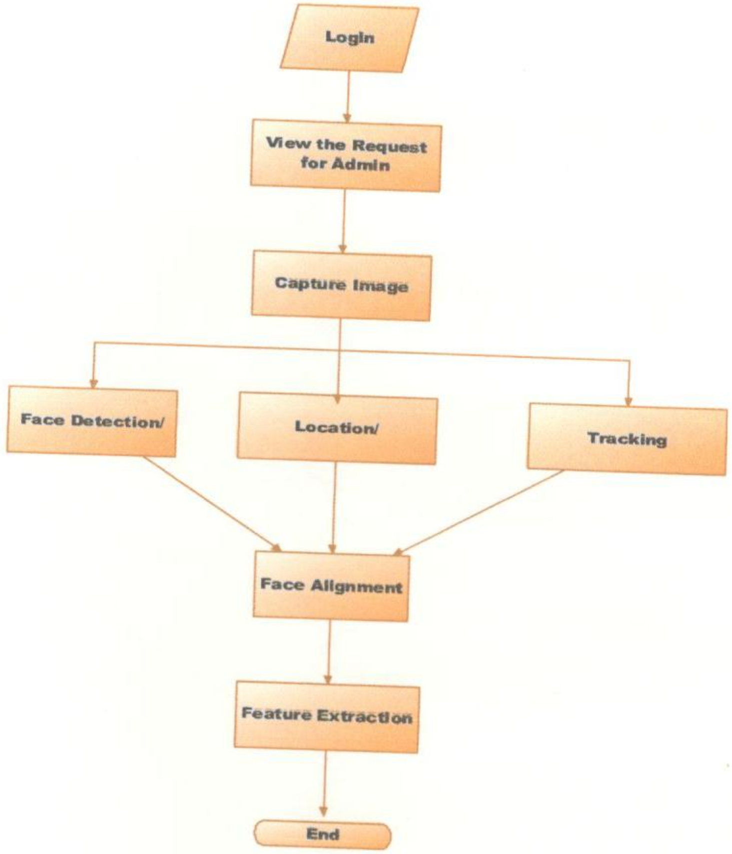

Fig. 4.3: Data Flow Diagram

Fig. 4.3 shows the data flow diagram of video analytics. A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually "say" things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That's why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.

4.8 Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site.

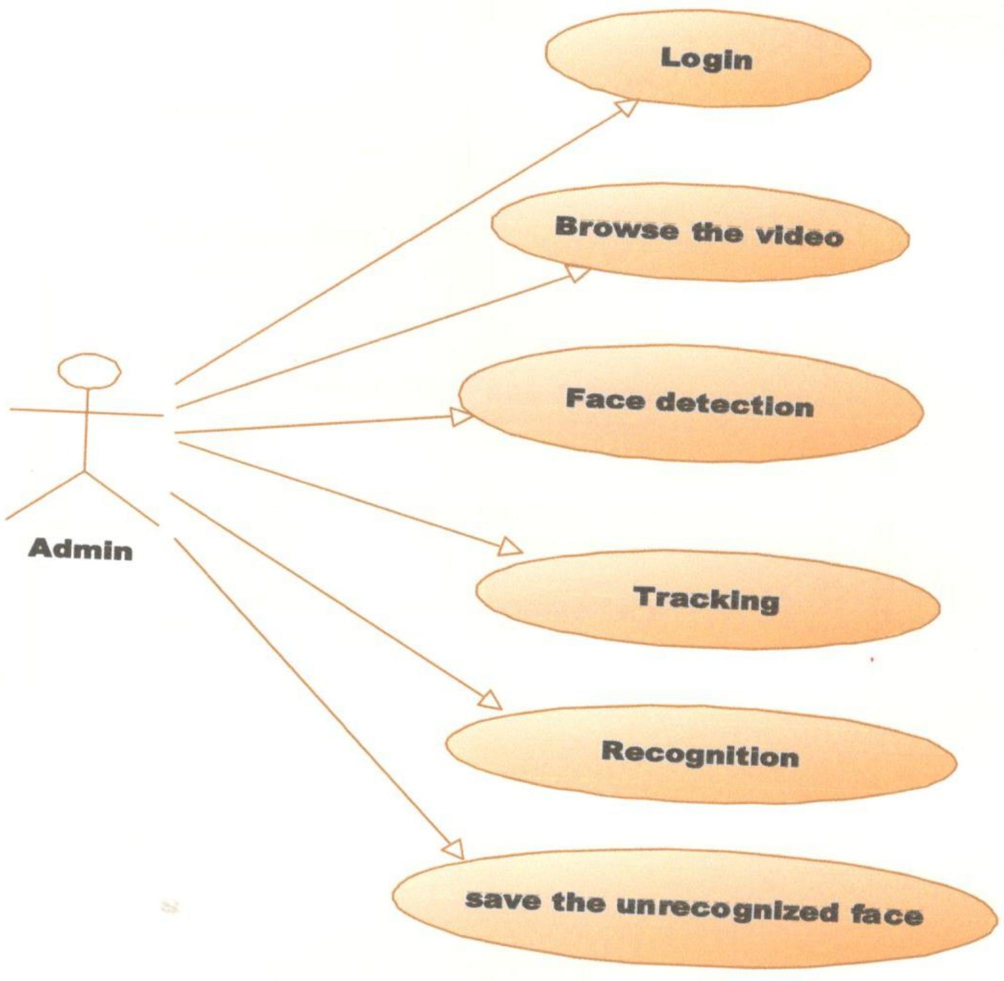

Fig. 4.4: Use Case Diagram

Fig. 4.4 shows the use case diagram of strategies for quality aware video analytics. In this the admin can login, browse the video, detect the face, tracking, face recognition and save the unrecognized face. Object recognition is an extended version of object detection that uses object detection in the initial stage, then maps the detected image into a known related sample dataset to match the features and try to recognize a unique object. Object recognition is widely used for face recognition, number plate recognition, and handwriting recognition. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements.

4.9 Sequence Diagram

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram.

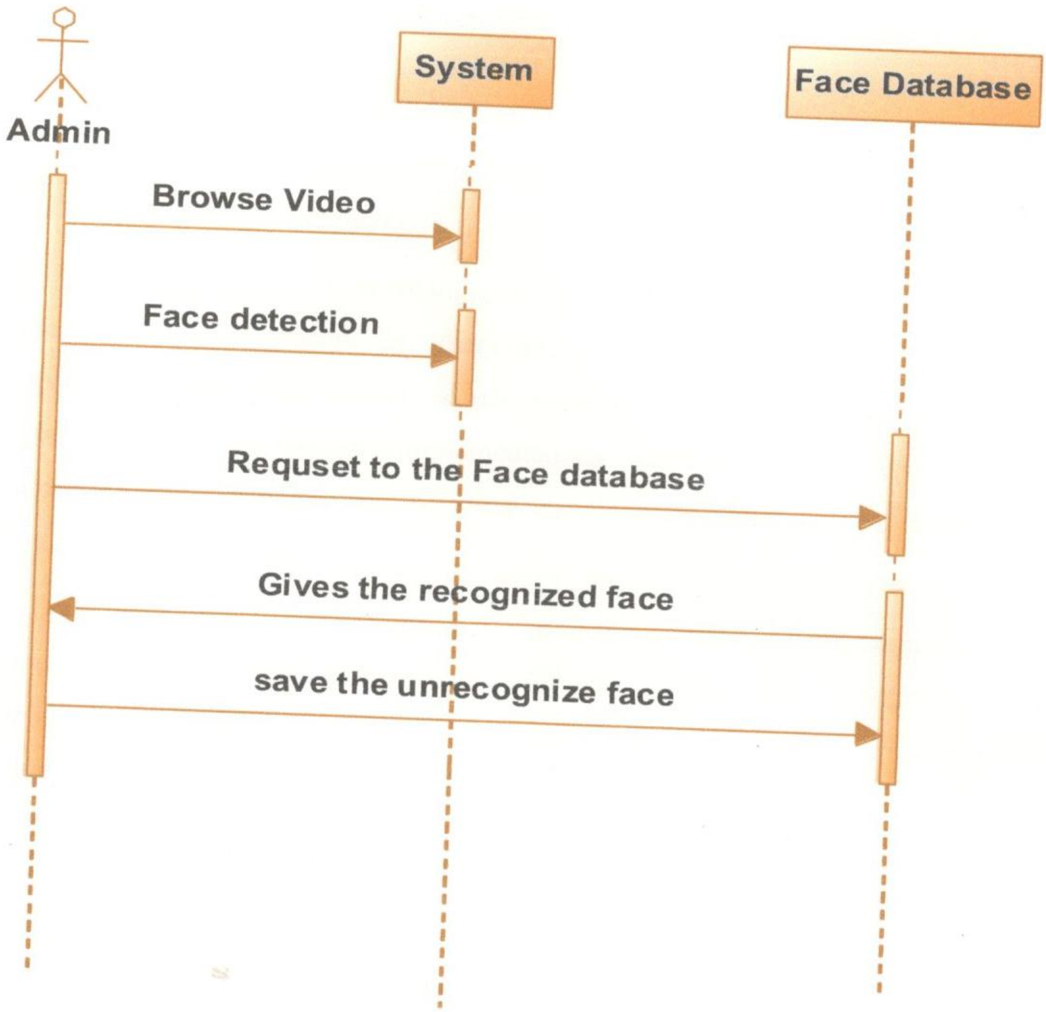

Fig. 4.5: Sequence Diagram

Fig. 4.5 shows the sequence diagram of strategies for quality aware video analytics. In this the admin can login, browse the video, detect the face, tracking, face recognition and save the unrecognized face.

Sequence diagrams describe how and in what order the objects in a system function. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.

Chapter 5

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

An implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through computer programming and deployment. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium-recommended specifications, and software development tools contain implementations of programming languages.

A special case occurs in object-oriented programming, when a concrete class implements an interface, in this case the concrete class is an implementation of the interface and it includes methods which are implementations of those methods specified by the interface.

5.1 Project Module Description

5.1.1 Design & Authentication Module

In this module is going to explain the Robust Face-Name Graph Matching for Movie Character Identification designing and how we did the face detection and recognition in this project. The images will explain about the facial fetching details. After that admin going to login with the details which needed for the login page.

Video surveillance system (VSS) is increasingly becoming important part in daily life. For traditional VSS, each camera stores streaming data to a centralized server. It will create a great volume of video data for a large VSS and may raise some issues in keeping daily video data to centralized server, such as limited bandwidth and storage insufficiency of server and lower reliability and scalability. In order to solve the problem, we have proposed an architecture for VSS based on well-developed peer-to-peer technique and emerging cloud computing.

5.1.2 Detection

In this module we are going to detect the face of the movie characters. In this module we are using the emgu cv library we must install the emgu cv library. After installing the emgu cv lib in our project we need to add reference with the name emgu.cv, emgu.cv.util, emgu.cv.ui.

When you will complete the references you will get the emgu controls in the toolbox. Video content analysis (also video content analytics, VCA) is the capability of automatically analyzing video to detect and determine temporal and spatial events. Video Motion Detection is one of the simpler forms where motion is detected with regard to a fixed background scene.

Face detection is conducted occasionally to adjust locations of tracked objects and to discover new objects in video as well. A multi-view face detector is developed based on the Viola-Jones method. Three classifiers are trained and applied for frontal, half-profile and profile faces, respectively. And with each classifier, it works at multiple directions.

5.1.3 Training Module

In this module, we train the faces which are detected in the earlier module. The user can train the system by adding the names of the user. The name of the training data set in stored in image format with the graph name.

5.1.4 Recognition

In this module we are going to recognize the face of the movie characters which is we previously stored on the face database. We just found that the give the real name of it. This is going to be done here. Here we are using the With the help of these Eigen Object Recognizer we are going to recognize the face.

Once the face image has been normalized, the feature extraction and recognition of the face can take place. In feature extraction, a mathematical representation called a biometric template or biometric reference is generated, which is stored in the database and will form the basis of any recognition task. It is important for successful recognition that maximal information is retained in this transformation process so that the biometric template is sufficiently distinctive.

Chapter 6

TESTING

6.1 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTS

6.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input	: identified classes of valid input must be accepted.
Invalid Input	: identified classes of invalid input must be rejected.
Functions	: identified functions must be exercised.
Output	: identified classes of application outputs must be exercised.
Systems/Procedures	: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

6.2.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

- **White Box Testing**

  White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

- **Black Box Testing**

  Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

6.3 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.4 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

6.5 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Chapter 7

RESULTS& SNAPSHOTS

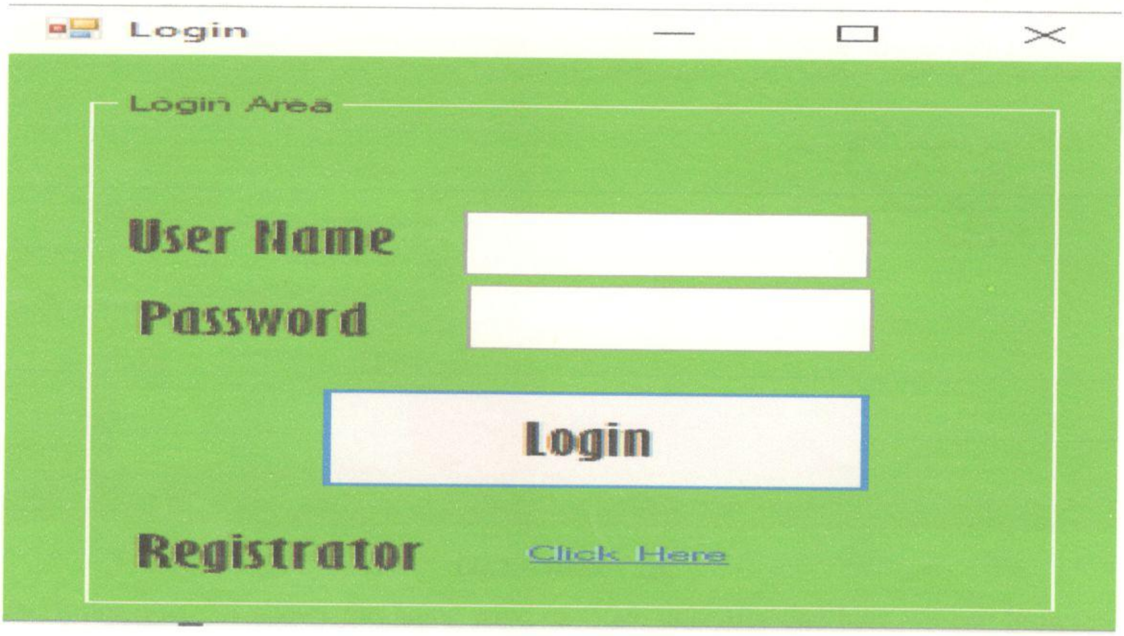

Fig.7.1: Login Page

Description: Fig. 7.1 shows the Login page with username and password, before login user have to get register. Here we see click here to get register.

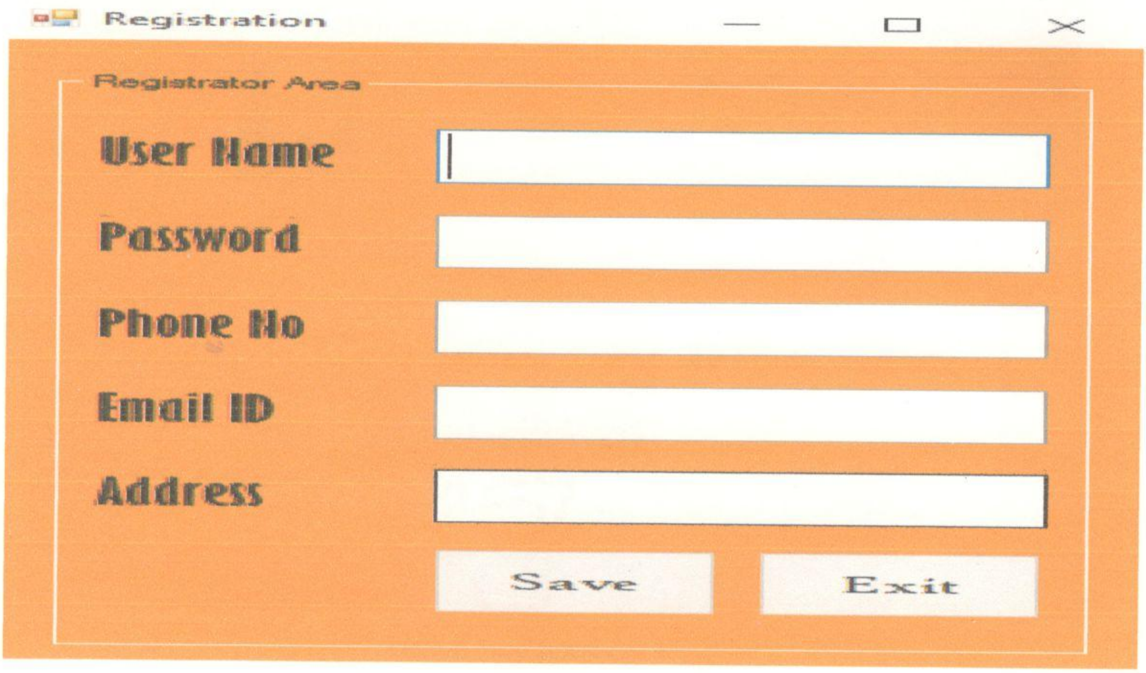

Fig.7.2: Registration Page

Description: Fig. 7.2 shows the Registration page to get register with the following fields and the registrator have to get register first and then have to get login.

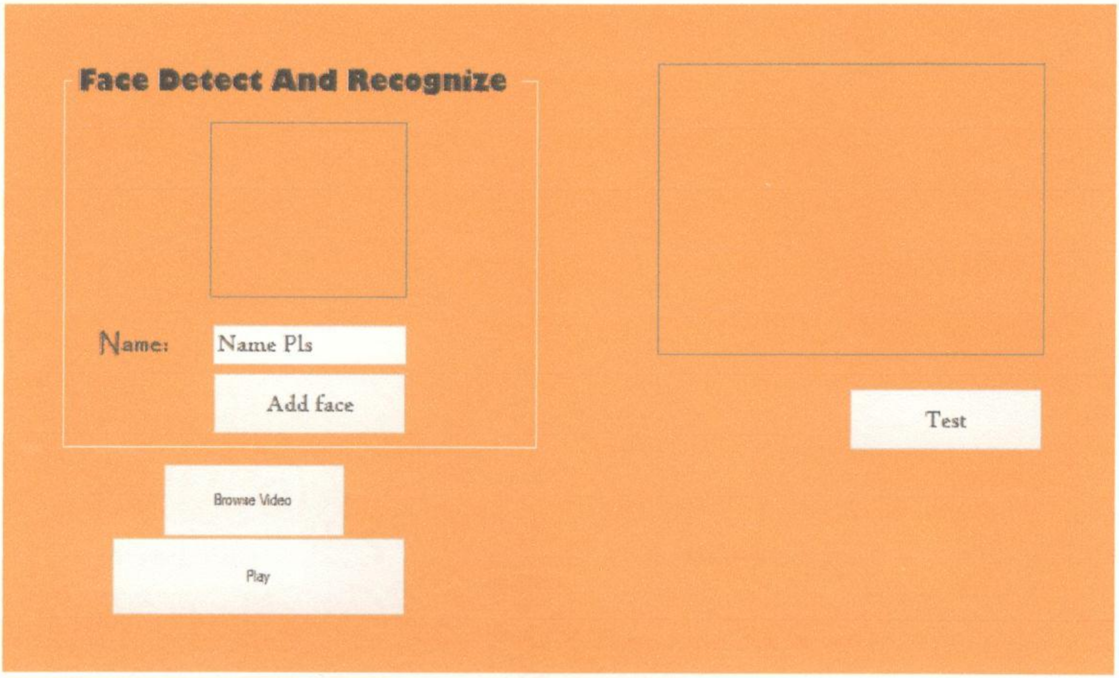

Fig.7.3: Home Page

Description: Fig. 7.3 shows the home page of video analytics and here we can browse the video and name can be given to the detected face.

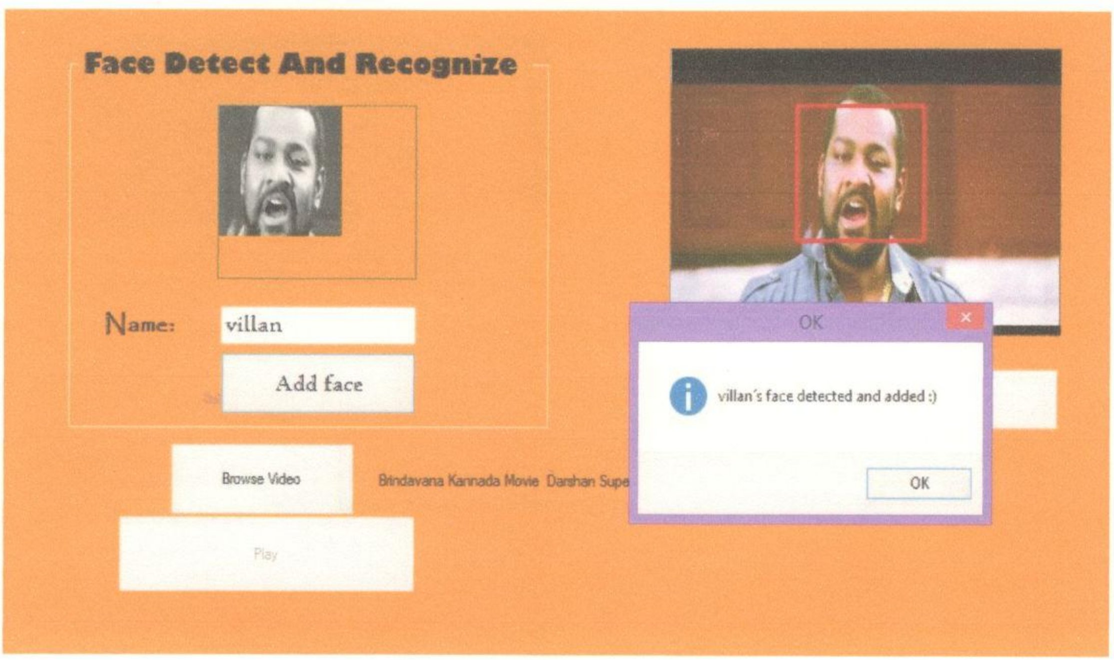

Fig.7.4: Face detect and recognize

Description: Fig. 7.4 shows the Face detect and recognize tool, here the video is browsed first and name given to the detected face.

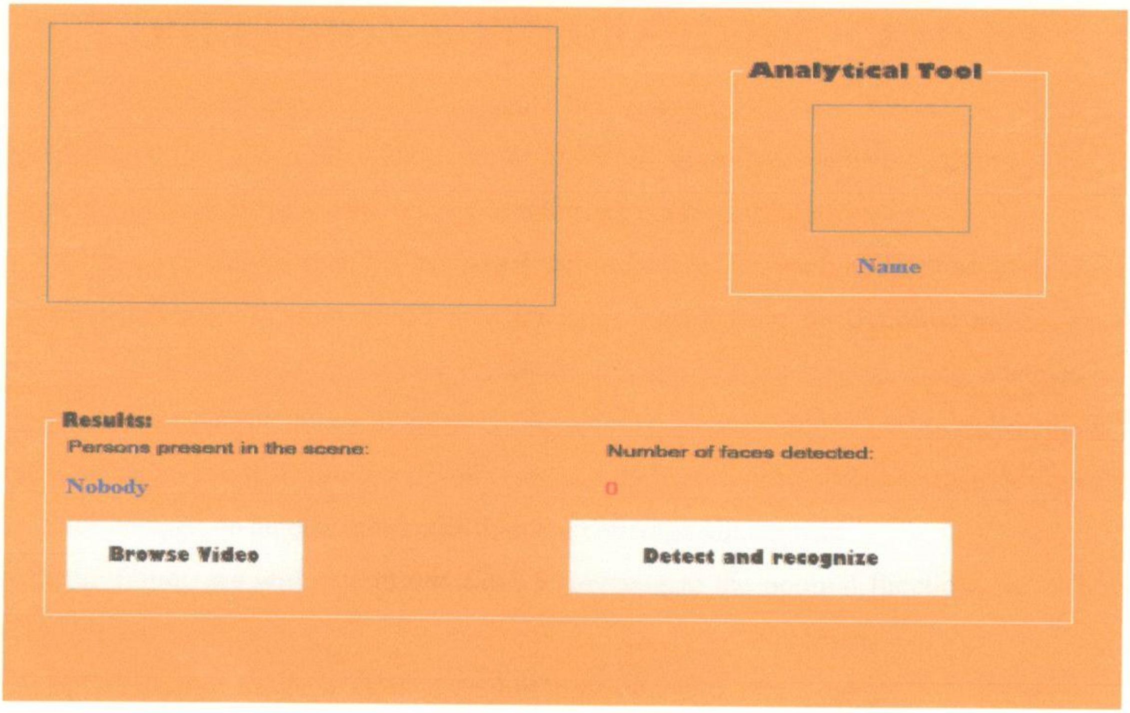

Fig.7.5: Before adding video for testing

Description: Fig.7.5 shows the before adding video for testing. In this it just shows the detect and recognize faces in the video and counts the number of faces detected.

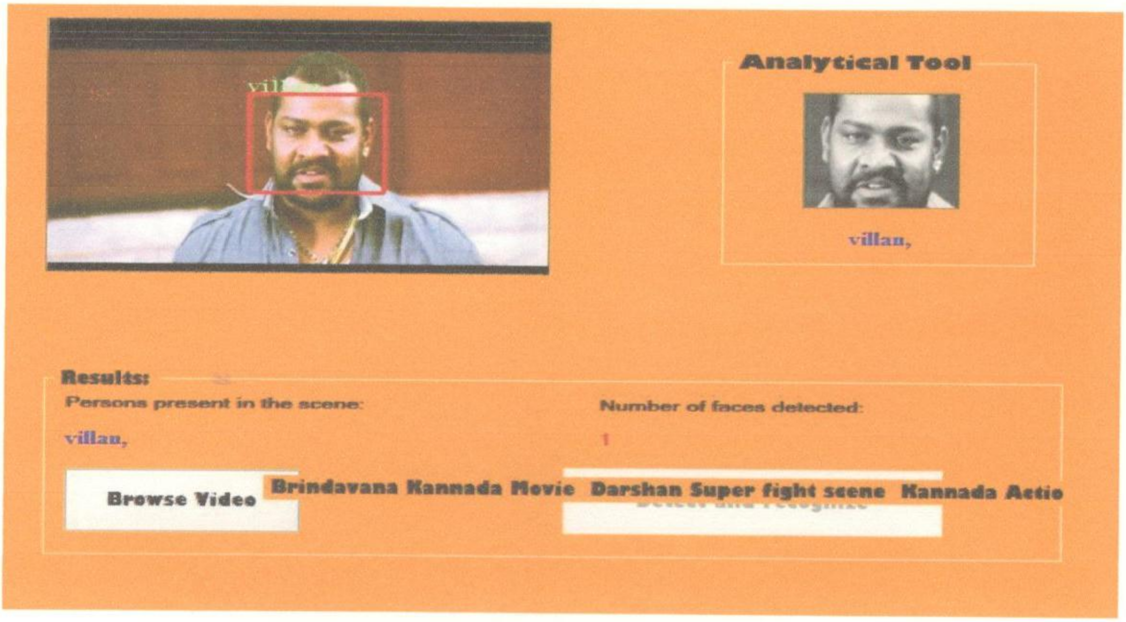

Fig. 7.6: Adding Person face

Description: Fig.7.6 shows how todetect the person face and Shows the number of faces detected. The detected face is displayed with the name given and it also shows the path where video is browsed.

CONCLUSION & FUTURE ENHANCEMENT

We reviewed approaches to design video analytics systems, for videos which have degraded quality. Due to the wide range of analytics algorithms, potential degradations, and design approaches, there are many opportunities for continued improvement.

We have shown that the proposed two schemes are useful to improve results for clustering and identification of the face tracks extracted from uncontrolled movie videos. From the sensitivity analysis, we have also shown that to some degree, such schemes have better robustness to the noises in constructing affinity graphs than the traditional methods. A third conclusion is a principle for developing robust character identification method: intensity a like noises must be emphasized more than the coverage alike noises.

In the future, we will extend our work to investigate the optimal functions for different movie genres.

Another goal of future work is to exploit more character relationships, e.g., the sequential statistics for the speakers, to build affinity graphs and improve the robustness.

REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," ACM Computing Surveys (CSUR), vol. 35, no. 4, pp. 399–458, 2003.

[2] S.-R. Ke, Hoang Le UyenThuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, "A review on video-based human activity recognition," Computers, vol. 2, no. 2, pp. 88–131, 2013.

[3] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage CNNs," in IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1049–1058.

[4] H. Pirsiavash, C. Vondrick, and A. Torralba, "Assessing the quality of actions," in European Conference on Computer Vision, 2014, pp. 556–571. [5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in International Conference on Learning Representations, 2015.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., "Imagenet large scale visual recognition challenge," International Journal of Computer Vision, vol. 115, no. 3, pp. 211–252, 2015.

[8] P. Korshunov et al., "Critical video quality for distributed automated video surveillance," in ACM International Conference on Multimedia, 2005, pp. 151–160. [9] K. Tahboub, A. R. Reibman, and E. J. Delp, "Accuracy prediction for pedestrian detection," in IEEE International Conference on Image Processing (ICIP), Sept 2017, pp. 4192–4196.

[10] K. Tahboub, D. Guera, A. R. Reibman, and E. J. Delp, "Quality- ¨ adaptive deep learning for pedestrian detection," in IEEE International Conference on Image Processing (ICIP), Sept 2017, pp. 4187–4191.

[11] B. Lucas, T. Kanade, "An iterative image registration technique with an application to stereo vision," Intl'JointConf.onArtificial Intelligence,p.674-679, 1981.

[12] D. Comaniciu, V. Ramesh, P. Meer, "Real-time tracking of non-rigid objects using mean shift," IEEE Conf. onCVPR, p.142-149, 2000.

[13] M. Arulampalam, S. Maskell, et al. "A tutorial on particle filters for onlinenonlinear/non-Gaussian Bayesian tracking,"IEEE Trans. on Signal Processing,Vol.50, No.2, Feb. 2002.

[14] T. D. Ngo, D.-D. Le, et al. "Robust face track finding in video using tracked points," IEEE Intl' Conf. on Signal ImageTechnology and Internet Based Systems, 2008.

[15] X. Wu, L. Li, et al. "A framework of face tracking with classification using CAMShift-C and LBP," Fifth Intl'Conf.on Image and Graphics, p.217- 222, 2009.

[16] M. Kim, S. Kumar, V. Pavlovic, et al. "Face tracking and recognition with visual constraints in real-world videos," IEEE Conf. on CVPR, 2008.

[17] V. Belagiannis, F. Schubert, et al. "Segmentation based particle filtering for real-time 2d object tracking," Euro.Conf.on Computer Vision, Vol. IV,p.842-855, 2012.

[18] Z. Kalal, K. Mikolajczyk, et al. "Face- TLD:tracking-learningdetection applied to faces," IEEE Intl' Conf.onImage Processing, 2010.

[19] E. Maggio, E. Piccardo, et al. "Particle PHD filter for multi-target visual tracking," IEEE Conf. on Acoustics,SpeechandSignal Processing, p.15-20,2007.

[20] R. G. Cinbis, J. Verbeek, et al. "Unsupervised metric learning for face identification in TV video," IEEE Intl' Conf. onComputer Vision, p.1559– 1566, 2011.

INTERNATIONAL JOURNAL FOR TECHNOLOGICAL RESEARCH IN ENGINEERING

ISSN (Online): 2347 - 4718

Certificate of Publication

This certificate is awarded to "Harshitha K S" in recognition of publication of the paper entitled "APPROACHES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS" published in IJTRE (International Journal For Technological Research in Engineering) Volume 6, Issue 9, May-2019.

Publisher|Editor In Chief

Mr. Saumil C Patel

INTERNATIONAL JOURNAL FOR TECHNOLOGICAL RESEARCH IN ENGINEERING

ISSN (Online): 2347 - 4718

Certificate of Publication

This certificate is awarded to "Lakshmi R" in recognition of publication of the paper entitled "APPROACHES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS" published in IJTRE (International Journal For Technological Research in Engineering) Volume 6, Issue 9, May-2019.

Publisher|Editor In Chief

Mr. Saumil C Patel

INTERNATIONAL JOURNAL FOR TECHNOLOGICAL RESEARCH IN ENGINEERING

ISSN (Online): 2347 - 4718

Certificate of Publication

This certificate is awarded to "Navya A S" in recognition of publication of the paper entitled "APPROACHES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS" published in IJTRE (International Journal For Technological Research in Engineering) Volume 6, Issue 9, May-2019.

Publisher|Editor In Chief

Mr. Saumil C Patel

Issuing Authority: IJTRE

INTERNATIONAL JOURNAL FOR TECHNOLOGICAL RESEARCH IN ENGINEERING

ISSN (Online): 2347 - 4718

Certificate of Publication

This certificate is awarded to "Navya A U" in recognition of publication of the paper entitled "APPROACHES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS" published in IJTRE (International Journal For Technological Research in Engineering) Volume 6, Issue 9, May-2019.

Publisher|Editor In Chief

Mr. Saumil C Patel

STRATEGIES FOR QUALITY-AWARE VIDEO CONTENT ANALYTICS

Amy R. Reibman

School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana, USA

ABSTRACT

Recent research in video analytics promises the capability to automatically detect and extract information from video. Potential tasks include object and pedestrian detection, object and face recognition, motion detection, object tracking, as well as background subtraction and activity recognition. However, in many instances, the quality of the video from which information is to be extracted is not very high. This may be because of system constraints (like a bandwidth constraint or VHS recorder), environmental conditions (fog or low light), or a poor camera (wobbly/moving camera, limited FOV, or just a low-quality lens).

In this paper, we provide an overview of research on designing video analytics systems that use potentially low-quality data. We consider a variety of analytics tasks, and present five categories of approaches to create quality-aware analytics: quantify the impact, predict the impact, create an analytics-aware encoder, enhance the input before analytics, and modify the analytics algorithms.

Index Terms— Video Analytics, Video Quality

1. INTRODUCTION

The goal of video content analytics is to use a machine, not a person, to extract information from video. The extracted information can answer questions of who [1], what [2], where, when [3], and "how well" [4]. Video analytics can be *retrospective* (looking back at the past for forensic evidence), *real-time* (operating "in the now" and generating alerts), *predictive* (identifying what is likely to happen in the future, perhaps because of some identified anomaly), or *prescriptive* (deciding what action should be taken based on the data).

The ability of automated methods to extract information from video has been increasing dramatically in recent year, in part due to recent successes of training deep neural networks using extensive datasets [5, 6]. In many cases, the systems are trained on clean data: not just data that have been trimmed, cropped, painstakingly labeled, but also data that are mostly free from the types of degradations that pervade real systems [7]. For example, images from the ImageNet classification dataset typically contain one relatively-centered clearly-focused object that fills most of the image frame.

When a system is deployed in real-world scenarios, performance can suffer. Many quality degradations occur at the moment of image or video capture (for example, lighting or illumination, or low-light noise or motion blur), while others are "self-inflicted" because of constraints within the system (for example, transcoding, downsampling, or frame dropping). Low-quality inputs degrade analytics performance [8–11]. To study the confluence of video quality and video analytics, it is necessary to consider at least 3 distinct topics: the degradations that can be present, the types of problems that video analytics systems can be designed to address, and the methodologies of solutions for those video analytics systems. Each methodology might have its own vulnerabilities, and hence require its own remediation method.

In this paper, we begin in Sections 2 and 3 with a short summary of degradations and video analytics, respectively, and then in Section 4 describe five categories of approaches that have been considered in the literature to overcome these challenges. Due to space restrictions, we choose representative references only.

2. VIDEO ANALYTICS OVERVIEW

A typical processing pipeline for a video analytics system includes capture, compression, transmission, analysis, alerting, and storage. Our focus in this paper is to consider the implications of the degradations created during capture, compression, and transmission on the analytics tasks. Three core tasks are recognition, localization, and detection, each of which can be applied to a variety of objects, events, or activities. Specific objects of interest include people, faces, text, vehicles, and license plates, while example actions of interest may include gestures, slip and fall, or leaving a bag. Additional tasks include object, image, semantic, and instance segmentation, scene understanding, 3D-scene reconstruction, summarization, tracking, and generic anomalous event detection.

In many cases, there are common computational steps that can be shared across a variety of analytics tasks. Foreground/background segmentation may be useful for object tracking and object detection, while image registration, object classification, tracking, and action recognition may all share the same keypoint extraction step. In addition, the result of a first task may determine whether a second task is performed or not. For example, if there has been nothing moving in the scene, there is little reason to perform pedestrian detection.

Three basic types of features are typically extracted from video to perform video analytics tasks: hand-crafted features

*This work was supported by the Cisco University Research Program Fund CG-#594368 through the Silicon Valley Community Foundation

[12–15], kernel-based descriptors [16], and features learned using deep convolutional neural networks [5, 6]. It is likely that each feature will have its own robustness to impairments in the source video.

3. VIDEO QUALITY DEGRADATIONS

Video degradations are commonplace in deployed systems. Some degradations may happen at the moment of capture, i.e., in the camera, while others may be "self-inflicted", introduced prior to the analysis stage because of constraints within the system. For example, transcoding may happen in the network if bandwidth is limited and there is adaptive bit-rate video streaming. After all, it is often considered better to obtain a low-rate version of the video than no video at all.

The quality of video captured by the camera depends on the spatial and temporal resolution, any lens distortions (like fish-eye), rolling shutter, and blur. A poorly placed camera might impair the field of view of the object or action to be identified. In addition, camera motion may degrade visual quality, but may also provide valuable information for analytics. For example, a camera mounted on the body or the head may provide useful information to identify human behavior. Finally, the camera's viewpoint and location may create obstructions, and an elevated camera may alter the perspective on objects within the field of view.

Environmental viewing conditions can also impair the video during capture. In particular, lighting and illumination can create glare, reflections, and under-exposed video. Low light conditions can also introduce noise. Rain can cover the camera lens and create distortions, clouds may create time-varying illumination issues, while fog and snow may decrease quality by reducing contrast.

Finally, video compression and transmission-induced packet losses or RF interference can all decrease video quality between the camera and the analysis stage. Video compression depends on the available bandwidth for communication, and it may be unavoidable when it is imposed by system constraints. Each of the camera, compression, and transmission may introduce blur or noise.

Each analytics task may have its own specific quality requirements. Recognition may require sharper, higher-quality inputs either than detection or than analytics on larger objects using simple motion or flow. Object recognition performance depends on resolution, lighting and illumination, blur, occlusion angle, field of view. The performance of tracking depends on object speed, shape, and deformations. Tracking may need higher frame rate than classification, but classification may need a higher spatial resolution. Moreover, the frame-rate during tracking should be adequate to capture an object's speed and motion.

As mentioned above, in some cases the degraded quality may actually be informative for the analytics system. For example, camera motion may either be considered to be a degradation (when it detracts from the ability to identify an object), or actually be the information to be extracted (when the camera is mounted on an object).

4. QUALITY-AWARE VIDEO ANALYTICS

The prior work on the topic of video analytics using degraded-quality images or videos can be subdivided into 5 basic categories.

- Quantify the impact.
- Predict the impact.
- Create an analytics-aware encoder for video or features.
- Enhance the input before analytics.
- Modify the analytics algorithm.

We will consider each of these topics below.

4.1. Quantify the impact

Quantifying the impact of a degradation is straightforward: apply the analytics algorithm to a variety of degraded inputs, at increasing levels of severity, for each type of degradation of interest, and characterize the resulting accuracy. The impact of compression on face detection, tracking, pedestrian detection, and activity recognition have been studied in [8–11, 17–20], and [21] considers the impact of object size, occlusion, and aspect ratio. It is unnecessary to distribute databases containing synthetically-introduced distortions. Instead, a methodology that repeatably introduces synthetic distortions at the requested level [22] is more flexible.

The result of this type of study can allow a system designer to understand the required video quality to achieve a given prediction accuracy, and to determine what resources are necessary to achieve that video quality. For example, it may be possible to characterize the minimum necessary bandwidth in aggregate, and then design the system with lower resource requirements. However, if insufficient resources are available, then it allows performance expectations to be adjusted based on system constraints. In addition, this knowledge can be used to dynamically allocate resources in a camera network [23].

However, it is important to note that the impact of quality on accuracy may be content dependent. It was shown in [11] that to obtain sufficient accuracy for activity recognition, a different minimum quality may be required for each activity. Also, the frame-rate required for accurate object tracking was shown to depend on image content in [24].

4.2. Predict the impact

A variety of approaches have been explored, for *predicting* the impact of quality degradation on the accuracy of a video analytics algorithm. One basic approach is to consider this prediction as a quality estimation, where quality to be estimated is the analytics accuracy. This differs from the usual no-reference quality estimation [25], in that the goal is not to characterize how humans will perceive the video. This prediction can happen at the edge of the network, prior to compression, or near the video analytics engine.

A quality model to compare the expected accuracy of ee moving object detection algorithms before applying npression was proposed in [26]. In [27], a quality model predict object tracking performance of a specific tracker s designed, when the frame rate and spatial resolution var-l. Pedestrian detection was considered in [9], where the edicted accuracy was proposed to be used in two ways: first the encoder to choose the level of compression that lowers bit-rate most without sacrificing detection accuracy, and cond at the decoder to choose the lightest-weight method mputationally) that will still satisfy the desired accuracy. destrian detection was also considered in [10], with the al to predict at the decoder, for a given degraded input leo, which of several quality-aware analytics algorithms e section 4.5) will provide the best performance. A sim-r idea was applied for activity recognition in [11], which edicts the level of compression necessary at the encoder each input video, with the goal of achieving as much npression as possible for each individual video, without ecting the system's ability to recognize activities.

. Create an analytics-aware encoder

eating an analytics-aware encoder is distinct from the above ategies that choose the bandwidth at which to compress the leo. Instead, the goal here is to modify the video encod-algorithm to provide the best-for-analytics bitstream at a en bandwidth, **or** to design an encoder that operates di-tly on features already extracted during video analytics.

Algorithms that modify the video encoding parameters, create a standards-compliant bitstream knowing that it will used solely for analytics, were designed in [28–31]. This ows the encoding parameters to be specifically chosen to nimize the impact of compression on analytics.

In the Analyze-Then-Compress strategy (ATC) [32], fea-es are extracted prior to video compression, and a feature-ecific encoder is created. This avoids the problem of ex-cting useful features from degraded compressed video, requires sufficient processing power at the network edge feature computation. This approach was used for object ognition [32,33], tracking [34], and a hybrid approach that codes both features and video was proposed for pedestrian ection in [35].

. Enhance the input prior to analytics

other straightforward approach is to enhance the video ut prior to running the analytics algorithm. Demosaicing, ite balancing, and denoising are all applicable here, as dehazing or contrast enhancement [36, 37]. The ability deblurring, image interpolation, and single-image super-olution algorithms to improve object recognition is con-ered in [38].

. Modify the analytics algorithm

wide variety of directions have been explored to create ality-aware video analytics algorithms. Each can be applied to individual or multiple degradations, and can be generic for all levels of quality or targeted to a specific quality range [10].

Each stage of the analytics algorithm can be modified to account for lower-quality inputs. Blur-robust descriptors for face recognition are designed in [39, 40], and [41] modifies gradient descriptors to accommodate the distortion caused by a wide-angle lens.

Other approaches include domain adaptation and transfer learning [42–46], using data augmentation during training to include additional synthetically created distortions, or simply training the system on low-quality data [10]. In addition, performance of a system that will use only low-resolution data can be significantly improved by incorporating high-resolution inputs during training [46, 47].

There are ample opportunities to integrate concepts of No-Reference Quality Estimation algorithms [25] into various video analytics algorithms. For example, quality-based features were shown to significantly improve performance of a face detection algorithm given low-quality inputs in [20]. A systems-based approach was designed in [10], where a quality-estimation algorithm was created to select among a collection of pedestrian detection modules, each of which had been optimized to work as well as possible on some range of quality.

5. CONCLUSIONS

We reviewed approaches to design video analytics systems, for videos which have degraded quality. Due to the wide range of analytics algorithms, potential degradations, and design approaches, there are many opportunities for continued improvement.

6. REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[2] S.-R. Ke, Hoang Le Uyen Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi, "A review on video-based human activity recognition," *Computers*, vol. 2, no. 2, pp. 88–131, 2013.

[3] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage CNNs," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1049–1058.

[4] H. Pirsiavash, C. Vondrick, and A. Torralba, "Assessing the quality of actions," in *European Conference on Computer Vision*, 2014, pp. 556–571.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[8] P. Korshunov et al., "Critical video quality for distributed automated video surveillance," in *ACM International Conference on Multimedia*, 2005, pp. 151–160.

[9] K. Tahboub, A. R. Reibman, and E. J. Delp, "Accuracy prediction for pedestrian detection," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2017, pp. 4192–4196.

[10] K. Tahboub, D. Güera, A. R. Reibman, and E. J. Delp, "Quality-adaptive deep learning for pedestrian detection," in *IEEE International Conference on Image Processing (ICIP)*, Sept 2017, pp. 4187–4191.

[11] C. Zhong and A. R. Reibman, "Prediction system for activity recognition with compressed video," in *Electronic Imaging, Visual Information Processing and Communcation*, 2018.

[12] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 2, no. 60, pp. 91–110, 2004.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2005, pp. 886 –893.

[14] Ivan Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[15] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal on Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[16] L. Bo, Z. Ren, and D. Fox, "Kernel descriptors for visual recognition," in *Advances in Neural Info. Processing Systems*, 2010, pp. 244–252.

[17] P. Korshunov, "Rate-accuracy tradeoff in automated, distributed video surveillance systems," in *ACM International Conference on Multimedia*, 2006, pp. 887–889.

[18] P. Korshunov et al., "Reducing frame rate for object tracking," in *Advances in Multimedia Modeling*, pp. 454–464. Springer, 2010.

[19] P. Korshunov and W. T. Ooi, "Video quality for face detection, recognition, and tracking," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 7, no. 3, pp. 14, 2011.

[20] S. Gunasekar, J. Ghosh, and A. Bovik, "Face detection on distorted images augmented by perceptual quality aware features," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, December 2014.

[21] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors," in *European Conference on Computer Vision (ECCV)*, 2012, pp. 340–353.

[22] H. Liu and A. R. Reibman, "Software to stress test image quality estimators," in *IEEE International Conference on Quality of Multimedia Experience (QoMEX)*, 2016, pp. 1–6.

[23] A. S. Kaseb, B. Fu, A. Mohan, Y.-H. Lu, A. Reibman, and G. K. Thiruvathukal, "Analyzing real-time multimedia content from network cameras using CPUs and GPUs in the cloud," in *to appear*, 2018.

[24] A. Mohan, A. S. Kaseb, K. Gauen, Y.-H. Lu, A. R. Reibman, and T. J. Thacker, "Determining the necessary frame rate of video data for object tracking under accuracy constraints," in *to appear*, 2018.

[25] S. S. Hemami and A. R. Reibman, "No-reference image and video quality estimation: Applications and human-motivated design," *Signal Processing: Image Communication*, Aug. 2010.

[26] L. Kong, R. Dai, and Y. Zhang, "A new quality model for object detection using compressed videos," in *IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3797–3801.

[27] J. M. Irvine and R. J. Wood, "Real-time video image quality estimation supports enhanced tracker performance," in *Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications X*, 2013, p. 87130Z.

[28] L.-T. Cheok and N. Gagvani, "Analytics-modulated coding of surveillance video," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2010, pp. 127–132.

[29] E. Soyak, S. A. Tsaftaris, and A. K. Katsaggelos, "Low-complexity tracking-aware H.264 video compression for transportation surveillance," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 10, pp. 1378–1389, 2011.

[30] J. Chao and E. Steinbach, "SIFT feature-preserving bit allocation for H.264/AVC video compression," in *IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 709–712.

[31] J. Chao and E. Steinbach, "Keypoint encoding for improved feature extraction from compressed video at low bitrates," *IEEE Transactions on Multimedia*, vol. 18, no. 1, pp. 25–39, 2016.

[32] L. Baroffio, M. Cesana, A. Redondi, M. Tagliasacchi, and S. Tubaro, "Coding visual features extracted from video sequences," in *IEEE Transactions on Image Processing*, 2014.

[33] L. Baroffio, M. Cesana, A. Redondi, and M. Tagliasacchi, "Performance evaluation of object recognition tasks in visual sensor networks," in *International Teletraffic Congress (ITC)*, 2014, pp. 1–9.

[34] B. Panti, P. Monteiro, F. Pereira, and J. Ascenso, "Descriptor-based adaptive tracking-by-detection for visual sensor networks," in *IEEE International Conference on Multimedia & Expo Workshops*, 2015.

[35] S. D. Chen and P. Moulin, "A two-part predictive coder for multitask signal compression," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 2035–2039.

[36] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, 2011.

[37] A. Loza, D. R. Bull, P. R. Hill, and A. M. Achim, "Automatic contrast enhancement of low-light images based on local statistics of wavelet coefficients," *Digital Signal Processing*, vol. 23, pp. 1856–1866, 2013.

[38] R. G. Vidal, S. Banerjee, K. Grm, V. Struc, and W. J. Scheirer, "UG^2: a video benchmark for assessing the impact of image restoration andenhancement on automatic visual recognition," *arXiv preprint arXiv:1710.02909*, 2017.

[39] R. Gopalan, S. Taheri, P. Turaga, and R. Chellappa, "A blur-robust descriptor with applications to face recognition," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1220–1226, 2012.

[40] N. Anantrasirichai, J. Burn, and D. R. Bull, "Robust texture features for blurred images using undecimated dual-tree complex wavelets," in *IEEE Int. Conf. on Image Processing (ICIP)*, 2014, pp. 5696–5700.

[41] A. Furnari, G. M. Farinella, A. R. Bruna, and S. Battiato, "Distortion adaptive descriptors: extending gradient-based descriptors to wide angle images," in *International Conference on Image Analysis and Processing*, 2015, pp. 205–215.

[42] M. Saerens, P. Latinne, and C. Decaestecker, "Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure," *Neural Computation*, vol. 14, pp. 21–41, 2002.

[43] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Scholkopf, "Correcting sample selection bias by unlabeled data," in *Neural Information Processing Systems*, 2007.

[44] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[45] J. Xu, S. Ramos, D. Vazquez, and A. M. Lopez, "Incremental domain adaptation of deformable part-based models," in *British Machine Vision Conference (BMVC)*, 2014.

[46] X. Peng, J. Hoffman, X. Y. Stella, and K. Saenko, "Fine-to-coarse knowledge transfer for low-res image classification," in *IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3683–3687.

[47] Z. Wang, S. Chang, Y. Yang, D. Liu, and T. S. Huang, "Studying very low resolution recognition using deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4792–4800.