SEENSORED: FACE ANONYMIZER USING YOLOV7 OBJECT DETECTION ALGORITHM

A Thesis
Presented to the Faculty of
Computer Studies Department
College of Science
Technological University of the Philippines
Ayala Boulevard, Manila

by

James Manuel M. Lim
Jerome A. Domanico
Macrainne Kenshee A. Ompad
Marvin Jay C. Tablizo
Paul Kramer C. Ontar

In Partial Fulfillment of the
Requirements for the Degree
Bachelor of Science in Computer Science

May 2024

	TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd., Ermita, Manila, 1000, Philippines | Tel No. +632-5301-3001 local 608 Fax No. +632-8521-4063 | Email: cos@tup.edu.ph | Website: www.tup.edu.ph	Index No.	TUPM-F-COS-16-TAU
		Revision No.	00
	THESIS APPROVAL SHEET FOR THE UNDERGRADUATE PROGRAMS OF THE COS	Date	07012022
		Page	1 / 1

This thesis hereto entitled:

SEENSORED: FACE ANONYMIZER USING YOLOV7 OBJECT DETECTION ALGORITHM

prepared and submitted by **JEROME A. DOMANICO, JAMES MANUEL M. LIM, MACRAINNE KENSHEE A. OMPAD, PAUL KRAMER C. ONTAR,** and **MARVIN JAY C. TABLIZO** in partial fulfillment of the requirements for the degree **BACHELOR OF SCIENCE IN COMPUTER SCIENCE** has been examined and is recommended for approval and acceptance.

DR. WELLANIE M. MOLINO
Adviser

Approved by the Committee on Oral Examination with a grade of **PASSED** on MAY 23, 2024.

PROF. MAY GARCIA
Chair

PROF. JAN EILBERT LEE
Member

PROF. ARIEL TOMAGAN
Member

Accepted in partial fulfillment of the requirements for the degree **BACHELOR OF SCIENCE IN COMPUTER SCIENCE**.

PROF. DOLORES L. MONTESINES
Department Head

Date: June 14, 2024

DR. JOSHUA T. SORIANO
Acting Dean

Transaction ID	TUPM-COS-TAU-ELS-07012022 -0258PM
Signature	

Transaction ID Legend: TUPX-AAA (Office Code)-BBB (Type of Transaction)-CCC (Initial of employee)-MMDDDYYYY (month day year)-HHMMAM/PM (hourminutesAM/PM)

ACKNOWLEDGEMENT

First and foremost, we would like to express our deepest gratitude to our advisor, Prof. May M. Garcia., for her guidance, encouragement, and patience throughout the course of our research. Her expertise and insights have been invaluable, not only to this work, but also to our growth as researchers. We are truly grateful for her support and mentorship.

We would also like to thank Prof. Jan Eilbert L. Lee, Prof. Ariel L. Tomagan, Dr. Wellanie M. Molino, Prof. Edward N. Cruz, Prof. Amador B. Peragrino Jr., and Mr. Julius Brian Calma, for their valuable feedback and suggestions. Their rigorous standards and insightful critiques have greatly improved our work. We are grateful for their willingness to help us make our research the best it can be.

Lastly, we wish to express our profound appreciation to our families and friends for their unwavering understanding and support throughout this arduous journey. Their enduring encouragement, patience, and belief in our abilities have been a bedrock upon which we have relied. The completion of this project would not have been possible without their steadfast support, which provided us with the resilience and motivation needed to persevere through challenges and setbacks. Furthermore, we extend our heartfelt gratitude to one another, acknowledging the remarkable collaboration, and dedication that each member of our team has demonstrated. The collective effort, mutual respect, and shared commitment to excellence have been instrumental in bringing this project to fruition. Each member's unique contributions, perspectives, and expertise have enriched our research, allowing us to achieve a higher standard of academic rigor and innovation.

The journey of conducting this research has been both enlightening and transformative. We have grown not only as researchers but also as individuals, learning valuable lessons in teamwork, problem-solving, and perseverance. We take immense pride in the work we have accomplished, recognizing the significance of our contributions to the field of object detection and privacy protection.

Our hope is that the SEENSORED: Face Anonymizer Using YOLOv7 Object Detection Algorithm will make a meaningful and positive impact on the lives of individuals who seek to protect their privacy in an increasingly digital world. We envision our work as a stepping stone towards more secure and anonymous data usage, contributing to the broader discourse on privacy and technology. We remain optimistic that our research will pave the way for further advancements and inspire future scholars to explore and innovate in this vital area of study.

- The Researchers

ABSTRACT

Since 2016, video editing techniques have become crucial for protecting privacy, especially for anonymizing people in public spaces. Effective anonymization is necessary due to privacy concerns in the mass media industry, particularly for quick news delivery. Popular face-blurring methods, while privacy-preserving, can be time-consuming and degrade media quality. Social media also faces privacy issues, with users at risk of identity theft and phishing. In order to overcome these obstacles, the researchers developed a desktop video face anonymizer application based on YOLOv7 object detection algorithm. By automating face blurring in videos, this solution improves productivity for mass media industries and content creators, especially in the news sector. Video importation, face detection, voice audio anonymity, cache file creation and integrity verification, configurable blur types, and video exports are some of the key features. The application uses CustomTkinter for a user-friendly interface, leveraging YOLOv7 for object detection, and frameworks like FFmpeg and OpenCV. It offers an efficient method to process videos, safeguarding privacy, especially for minors, while optimizing workflows to meet high quality and safety standards. Evaluated by 45 respondents using the ISO 25010 quality software model, the application is seen as a valuable tool for privacy protection in video content creation. Despite promising results, limitations include the inability to process videos in real time, desktop use only, no face-saving tools, and limited file export formats. The application achieved a grand weighted mean of 3.61 from the ISO 25010 quality software model, indicating it is Highly Acceptable, effectively completing its tasks while adhering to ethical and legal content creation standards

Keywords**:** YOLOv7, video editing, face anonymization, mass media, content creation

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

Chapter 1

THE PROBLEM AND ITS SETTING

This chapter presents the background of the study, the general and specific objectives, and the scope and limitations of the study.

Background of the Study

Video editing methods for the protection of privacy have become increasingly vital since 2016, stating the protection of individuals' privacy in public spaces. Techniques such as cropping, blurring, or pixelating human faces are commonly employed in commercially available software such as Adobe Premiere, although they can distort the original image structure, especially noticeable in temporal video data where each frame's anonymization introduces a flickering effect. Balaji, Blies, Gori, Mitsch, Wasserer, and Schon (2021) introduced a two-step method that involves identifying and masking faces in individual frames before filling in the missing content using a Generative Adversarial Network (GAN) to ensure temporal coherence and natural-looking faces across multiple frames. Their approach avoids using facial characteristics or landmarks, advancing the state of the art in landmark-free facial inpainting and introducing an identity invariance score to assess temporal coherence. However, limitations persist, as demonstrated by Balaji et al. (2021), which highlights the challenge of maintaining temporal consistency in video anonymization, particularly in generating coherent appearances across consecutive frames with existing generative models based on images.

Media that is broadcast without restrictions may cause privacy problems. Content creators and the people working in the media industry often use software to address this

concern, they use a software application to blur the faces of the people. This can be time-consuming on their part, and this is also crucial for the people working in the news industry wherein they need to deliver news and information as soon as possible and blurring people's faces to protect their privacy can cause some slowdowns on this process. One approach to anonymize is face-blurring, this promotes privacy while maintaining uninterrupted media broadcasts. Privacy will not be invaded when their faces are anonymized in a video or a photo. Widespread usage of online social networks (OSN) are most commonly for self-expression, communication, thought-sharing, and media done daily. Sharing information on online social networks (OSNs) can occasionally require disclosing private information, raising privacy issues like unintentional facial recognition, inference attacks, location exposure, identity theft, phishing, profiling risks, or the sale of counterfeit goods (Fire, Goldschmidt, & Elovici, 2014; Kumar, Gupta, Rai, & Sinha, 2013). Together, these problems add to cybersecurity difficulties that could lead to cybercrime.

Blurring involves suppressing content that is invading privacy, sensitive information, and forms of expression. It can also restrict any information in the media that authoritative organizations can consider as propaganda. Blurring can affect the influence of people who normally consume this form of media and people's behavior, beliefs, perspective and learnings.

For the context of media, blurring has been used to avoid leaking personal information like posting a picture on social media platforms. Uploading pictures and videos on social media is a common activity online but some are not permitted to be recorded or to post sensitive information from bystanders making anonymity an issue for people since there are over 350 million worth of photos are being uploaded daily in

Facebook. Ilia, Polakis, Athanasopoulos, Maggi, and Ioannidis (2015) stated that the issue is caused by the way online social networks are currently designed, which gives people associated with shared photos little control over how visible they are and frequently allows other users to override their privacy settings. Social media is not just about a single person, but it might affect the peers around that person and users are not aware about revealing the peer's privacy when doing activities in social media making the peer's information also getting compromised.

The researchers aimed to aid media content creation through the development of a YOLO-based video face anonymizer desktop application. Content creators, including major news organizations, had to comply with social media platforms' community guidelines to keep the online community safe. Even if the scope of the content did not violate the community guidelines, ethical issues regarding content creation, especially vlog creation, still existed. The YOLO-based video face anonymizer application was designed to be integral to content creation by automatically blurring the faces of sensitive entities in the video input. Thus, content creators did not need to manually track frame-by-frame using video editing software to blur the faces of sensitive entities and produce the content.

Objectives of the Study

The general objective of this study is to develop a YOLO-based video face blurring application**.**

Specifically, it aims to:

1. Design a desktop application that blur faces in video inputs with the following features:

a) File importation module that allows user to import a video
b) Identifies the faces throughout the video input
c) Selected faces will be processed for face blurring
d) Cache memory that stores initialized video data
e) Pitch lowering module for voice audio anonymity
f) Custom blur type (black box/white box/pixelate/blur)
g) Exportation module that produces the final video output (with blurred faces)

2. Create the YOLO-based video face blurring application as designed using the following applications:
    a) Machine learning algorithm for face detection:
        i) YOLOv7 object detection algorithm
    b) Desktop application as a user interface using the following tools:
        i) Python programming language
        ii) OpenCV
        iii) FFmpeg frameworks
        iv) CustomTkinter
3. The developed YOLO-based video face blurring application aims to provide the following:
        i) Convenient video processing for content creators and mass media
        ii) An integral tool as contribution for acquiring healthy online community in content creation

iii) An integral tool for mass media productions in terms of content creation efficiency

iv) Security toward vulnerable individuals such as minors from privacy risks

4. Test and implement the YOLO-based video face blurring application using performance efficiency, and security.
5. Evaluate the level of acceptability of the developed YOLO-based video face blurring application using the ISO 25010 quality software model.

Scope and Limitations of the Study

A video anonymization application is an important technology in today's era, where videos are extensively shared online. It can be used to protect the privacy of individuals who are featured in videos without leaking their identities. The application has used YOLOv7 object detection algorithms, which is an advanced object detection algorithm that exceeds all the other object detection algorithms in both speed and accuracy. The model has used a dataset of face images provided by WIDER FACE dataset that was trained and got the accuracy that is needed. The model was able to detect and blur faces. The application was tested using different video datasets.

The researchers have used development tools in developing the SEENSORED desktop application. In the implementation desktop application's computer vision, OpenCV framework that allows functions for video processing tasks such as face detection. FFmpeg framework has been implemented for its video manipulation and transcoding tasks. Lastly, in the designing of the GUI (graphical user interfaces), the researchers have

used CustomTkinter as it is one of the widely used frameworks in designing GUI applications in Python.

The study aimed to develop a YOLOv7-based face blurring application for video anonymization. The desktop application is able to blur faces in videos, which makes it perfect for use in editors of contents in social media and broadcast media, especially on television and the internet. This conducted study benefits a wide range of populations, including individuals using online platforms and shares video contents such as content creators, as well as media and broadcast organizations.

On social media platforms, the application can be used by content creators and video editors to anonymize the faces of unwanted people within the video to protect their privacy especially minor individuals. Also, media and broadcast organizations can benefit from the usage of the application by reduction of video editing time cost due to traditional video blurring techniques for their publishing of news reports, documentaries, and other media content.

The application has the following limitations: real-time processing, desktop only, person's face saving, and file format exportation. The application cannot detect, and blur faces in videos real time. It means users must process in a recorded video to blur faces. The application is only available on desktop computers, it means the user cannot blur the face in the video using his/her mobile device or tablet. The application cannot save blurred faces or specific people on the video. It means the application does not have the database for the user to retain the blurred faces. Lastly, in video exportation, the only available file extension format that the application produces is .mp4 as it is the most widely used video file format.

The application was evaluated by purposely selected 45 respondents. The evaluation instrument to determine the level of acceptability of the desktop application was based from the ISO 25010 Software Quality Models.

Chapter 2

CONCEPTUAL FRAMEWORK

This chapter presents a review of related literature, related studies, the conceptual model of the study, and the operational definition of terms relevant to this study.

Review of Related Literature

This section discusses key concepts and ideas on the subject matter of the study. It presents topics anonymization, social media and content creation, video anonymization, AI machine learning, object detection and algorithms, YOLOv7, WIDER FACE dataset, Python, PyTorch, OpenCV, CustomTkinter, and evaluation system.

Anonymity

Anonymity has been recognized as a factor that promotes increased self-disclosure, whether in offline or online contexts. One early illustration of the augmented self-disclosure associated with anonymity is observed in the "stranger on a train" phenomenon, as outlined by Zick Rubin. This phenomenon refers to individuals divulging personal information quite intimately to fellow passengers during train journeys. John Suler identified the disinhibition effect in online settings, where individuals tend to disclose more about themselves in various online environments compared to face-to-face interactions. In the realm of Human-Computer Interaction (HCI), Knuttila (2011) suggest that anonymity, when strategically incorporated into design, can be utilized to shape online interactions and norms. Notable examples include platforms like Reddit and 4chan (Knutilla, 2011). Some investigations highlight the favorable aspects of anonymity. According to Christopherson, anonymity represents a crucial mechanism for

safeguarding privacy and promoting psychological well-being. Christopherson emphasizes that concealing one's physical identity can liberate individuals from discriminatory treatment related to factors such as gender, race, age, ethnicity, physical disability, and attractiveness. Elliston (1982) has stated that in situations involving whistleblowing, where there exists a substantial power imbalance between the whistleblower and the accused, anonymity may act as an incentive for individuals to step forward and reveal the truth, thereby promoting public welfare. An American Association for the Advancement of Science (AAAS) conference revealed that participants regarded anonymous communication as a fundamental human right. The consensus was that the discretion to determine the extent of anonymity should rest with individuals and organizations, acknowledging its importance (Weisbuch, Deffuant, Amblard, & Nadal, 2002).

Importance of Anonymity

The right to privacy is closely connected to anonymity. Without the capacity to manage the disclosure and utilization of personal information, an individual cannot reasonably anticipate the protection of their privacy. Trytko (2015) advocated for the right to remain anonymous contend that it safeguards individuals' privacy, autonomy, and freedom of expression. Additionally, it enables immunity against the overwhelming influence of the government, the market, and, in certain instances, repressive common norms.

Social Media and Content Creation in the Philippines

Due to COVID-19 pandemic emerging since 2020, the world has been affected, and several restrictions have been imposed to control the spread of the disease, including going outdoors became limited. This resulted the socialization being taken into online and social media. After the first year of COVID-19 emergency, the Philippines remained as top social media consumer but with significantly higher usage of 22 minutes compared to 2020's average, and 3 minutes higher 2019's average (Chua, 2021). The rise of social media consumption per minute is due to the new normal transition. According to Dela Peña (2023), Filipinos topped with 23% of consumers in time spending on social media in the South/Southeast Asia. An article published by Caparas (2023) discussed relevant statistics and activities toward social media in the Philippines. As of 2022, there were approximately 76 million social media consumers from the Philippines and 82.4% of it are social media account owners. Facebook is the most consumed social media platform in the Philippines. In addition, Facebook has a market share of over 90%. Over 51 million users follow celebrities and influencers on the platform. Around 97% of Filipino consumers prefer the usage of smartphones for social media access. Lastly, there were over 56 million YouTube consumers and almost 36 million in TikTok.

Within the COVID-19 pandemic, there was also an observed rise in the numbers of content creators. According to Caparas (2023), user-generated contents has emerged as Filipinos shifted to entertainment scene due to new normal, the trend still arising, and people started to pursue content creation on TikTok, YouTube, and so on. Caparas (2023) added the emerging trends in social media scene in the Philippines. The utilization of Instagram/Facebook reels and how they are used in advertising. Influencer marketing

wherein the brand and influencer has partnership for advertisements as influencer can sell, promote their brand. Lastly, the trend of one-minute videos commonly portrait as it is found as direct and concise, and enjoyable for consumption without the audience getting bored. Primary example of one minute-videos is TikTok videos and Facebook and Instagram reels.

The history of content creation in the Philippines started with the few Filipino content creators referred as vloggers. Lincoln Velasquez, also known as Cong TV, rose to fame due to his content creation on YouTube and started his career in 2011. As of 2023, he garnered over 11 million subscribers on YouTube, 5 million followers on Facebook, and almost 3 million followers on TikTok. A phenomenological qualitative inquiry was conducted by De Vera and Saludadez (2021) toward Cong TV. There are two out of thirteen sub-themes, Cong TV emphasized the lucrative opportunities toward what is his definition of content creation. The likes of Cong TV and other content creators are inspiration of Filipinos to pursue content creation in social media. Seeing social media content creation, also known as vlogging as an alternative way to find a living.

News outlets and journalists are also considered as content creators. Social media has been utilized by several news networks around the country as it is helpful for information dissemination and reach. According to Chua (2022), there was a significant growth in online news consumption in the Philippines across different social media platforms. Chua (2022) also stated that the GMA network is the Philippine's largest online news distributor in TikTok. While other news outlets such as ABS-CBN, News5, Rappler, Manila Bulletin, The Philippine Star, and DZRH gained significant following.

Ethical Issues toward Content Creation and Publishing

Social media contents are conveniently accessible due to the internet, easy platforms, and smartphones. Even content creation and publishing are convenient nowadays. due to user friendly features of the social media platforms. Every content creator must follow the community guidelines to prevent being banned from their platforms. According to CEOP Education (n.d.) community guidelines are set of rules often referred to as terms of use within the platform to maintain a healthy online community. Most of community guidelines cover behaviors that are prohibited online including: hate speech, harassment and bullying, nudity and sexual content, violence and graphic content, fake news and disinformation, and spam and frauds.

However, despite the existence of social media platforms' community guidelines, it still has its limitations as the online community is still not a sound environment. There will still be ethical issues revolving around social media particularly in content creation related to privacy. The most vulnerable individuals in this ethical issue are underaged children. According to Ahmed (2018), producing family vlog content is considered unethical at all angles. Several points emphasized that this kind of content is a form of child labor, compromises children's privacy, parent's accountability, psychological effects on children, and could cause harm to the bystander's privacy. According to Carrelo (2022), children are often experiencing breach of privacy in content creation and publishing due to lack of explicit consent making them very vulnerable.

Risks in the future of content creation

There are threats and risks in content creation that can affect the most vulnerable individuals such as children. According to Nottingham (2019), digital kidnapping refers to an act by an intruder uses, upload images of a child into their fake account. It is a type of identity theft that targets underaged individuals. Nottingham (2019) suggested that parents are responsible for the privacy and safety of their child. Another issue that poses risks is the rapid emergence of the utilization of artificial intelligence that developed deepfakes. According to Bussaca and Monaca (2023), deepfakes refer to the deceiving type of content produced by image synthesis with the aid of AI tools and commonly uploaded as misleading type of content. Children and youth are vulnerable to disinformation as deepfakes could be seen indistinguishable from real content. However, deepfake content is not also used to spread disinformation, entertainment for some individuals that consume contents, but it is also utilized to produce malicious content. A paper by Rizzica (2021) enumerates the risk of deepfake technology. The deepfake technology could be used to produce sexually explicit content that is AI-generated that utilized a person's physical identity without consent of the person rendered. Also, Rizzica (2021) included that the children are vulnerable to this kind of maliciousness due to deepfakes' ability to render anybody's face. With these kinds of threats, the predators are likely to consume these types of contents to feed their appetency towards having sexual interaction with underaged individuals. The spread and consumption of this AI-generated malicious content will be having detrimental effects for the individuals affected.

Methods of Video Anonymization

In 2016, the The European Union applied the General Data Protection Regulation (GDPR), with the aim of safeguarding individuals' personal rights through General Data Protection Regulation (GDPR). Due to this regulation, there is a necessity to anonymize individuals in photographs captured in public spaces. This process often involves actions such as cropping, blurring, or pixelating human faces (Neustaedter & Greenberg, 2003). While these methods guarantee anonymity, they have a significant drawback, as highlighted by Hudson and Smith. The original structure of the image is altered. In the case of temporal video data, this alteration becomes more pronounced, as each frame, when individually anonymized, introduces a flickering effect in the video sequence.

Balaji et al. (2021) introduced a two-step method, as outlined by Goodfellow et al. (2014). Initially, the approach involves identifying and masking out faces in individual frames. Subsequently, the missing content is filled using a newly generated artificial face through a temporally coherent Generative Adversarial Network. According to Heusel, Ramsauer, Unterthiner, Nessler, and Hochreiter, this method not only ensures consistent faces across multiple frames within a video sequence but also results in more natural-looking faces and higher FID scores, even in individual images compared to current state-of-the-art techniques. To achieve thorough anonymization without compromising original facial information, our approach avoids utilizing facial characteristics or features such as landmarks. The outcomes presented in Section VI-A illustrate that our proposed method advances the state of the art in landmark-free facial inpainting. To gauge and compare the temporal coherence of generated faces throughout a sequence of frames, we introduce an identity invariance score in Section VI-B.1. The evaluation of our results on single frames

is conducted on the FDF dataset (Hukelas, Mester & Lindseth, 2019), while assessing temporal coherence necessitates video sequences of faces.

Limitations of existing methods of Video Anonymization

Balaji et al. (2021) addressed the challenge of maintaining temporal consistency in anonymizing faces within natural video streams. They presented JaGAN, a two-step system that begins by identifying and concealing faces through black bar in each frame of the video. The second stage utilizes Video Generative Adversarial Network for privacy-preservation to fill in the obscured image patches with artificially generated faces. Initial experiments indicate that generative models based on images struggle to fill in patches with temporally coherent appearances across consecutive video frames.

Artificial Intelligence

Geetha and Bhanu (2018) stated that Artificial intelligence (AI) was defined by John McCarthy as "the science and engineering of making intelligent machines, especially intelligent computer programs." Artificial intelligence functions in the same ways as human intelligence, including learning, adapting, identifying, and correcting. The definition provided by the term's 1956 originator, John McCarthy, is "the science and engineering of making intelligent machines." It is a robotic system, or computer-enabled system, that is intended to process data and produce results in a manner similarly to how human workers inside the company use their capacity for learning, decision-making, and problem-solving to produce outcomes. AI is the science of utilizing computers to simulate human intellect.

Artificial Intelligence (AI) is a burgeoning trend and technology that offers a multitude of advantages to businesses. AI and ML save a corporation's money through optimizing the total cost of business operations. AI offers cognitive engagement, process automation, and cognitive insights to the business. It also helps the firms make better decisions, lower costs, and enhance customer satisfaction. Businesses can also use AI to solve problems and make smarter decisions in their operations. To enhance their overall business operations and processes, many organizations in the worldwide market are implementing and demanding the use of AI and ML technologies. The market size for AI and ML across several industries is also covered, along with the percentage and kind of business (Sriram et al., 2021).

Discriminative AI

According to Beijbom (2023), to make a choice or discern amongst several data kinds is the aim of discriminative AI. From the training data, these models identify the boundaries between several classes or categories and use that information to inform their predictions. Training a discriminative function is frequently faster when you need to finish a discriminative task. They also scale considerably better because it is simpler to add more data and outputs and to monitor how well the function works. Model performance can be enhanced by including a human reviewer in the loop to moderate decisions with low confidence and retrain the model using fresh annotated data.

According to Gales, Watanabe and Fosler-Lussier (2012), even while state-of-the-art systems today produce acceptable identification rates in certain areas, overall performance is insufficient for speech applications to proliferate. The given insights toward posterior probability of the classes (sentences) are precisely modeled in discriminative

models. Compared to generative models, this kind of model can perform better since it can leverage a larger variety of information from the observation and word sequences for inference.

Machine Learning

Machine learning is the process of developing computer systems that can learn and adapt on their own by examining patterns in data and drawing conclusions from statistical models and algorithms. New learning theories and algorithms have been developed recently, and this discipline has advanced due to the availability of large amounts of internet data and reasonably priced processing power. Due to such, science, technology, and business are incorporating data-centric machine learning techniques more and more into their operations. This is encouraging a greater reliance on evidence-based decision-making in a number of industries such as manufacturing, healthcare, marketing, education, finance, and policing. The development of computer systems that can learn and adapt on their own by examining data patterns and making inferences using statistical models and algorithms is known as machine learning. The development of learning algorithms, along with the concurrent advancements of internet data and cost-effective computers, have all contributed to the recent machine learning advancements. A greater use of evidence in decision-making is the outcome of science, technology, and industry embracing data-intensive machine-learning techniques. These industries include manufacturing, health care, marketing, finance, education, and law enforcement. Established at the convergence of computer science, statistics, and data science, it is considered to be one of the most well-known technical topics of today (Jordan & Mitchell, 2015).

According to Harrington (2012), Machine learning has always been common in people's daily lives and the amount of information that was received won't recede, making it be able to cohere for those employed in data-driven industries so being able to interpret all this data will be crucial. (Machine learning analyzes data instances, each of which has several properties. At the junction of statistics, computer science, engineering, and machine learning converge, often taking the form of many academic disciplines. Because of its adaptability, machine learning can be used in a variety of fields, including geosciences and politics. Machine learning techniques are a versatile tool that are useful in any industry that requires data interpretation and well-informed decision-making.) Data instances are investigated in machine learning. Each data instance consists of multiple features: machine learning, computer science, and engineering all overlap with statistics and are often used in different academic disciplines. It can be applied in many fields, such as the geosciences and politics. It is a multipurpose tool that may be applied to various problems. Various industries that requires data processing and interpretation can be aided with machine learning techniques.

According to Ayodele (2010), machine learning includes various types, such as reinforcement learning, transduction, unsupervised, semi-supervised, and learning to learn. A method that maps inputs to desired outputs—typically presented as a classification task—is produced during supervised learning. Unsupervised learning uses inputs as models without the need for labeled samples. To develop a suitable function or classifier, semi-supervised learning integrates labeled and unlabeled examples. The algorithm learns a policy through reinforcement learning, which is based on feedback from its activities in the environment. While transduction and supervised learning are similar, transduction

concentrates on immediately predicting novel inputs rather than explicitly creating a function. By leveraging historical data, the program can create its own inductive bias through learning to learn; supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, transduction, and learning to learn are the various forms of machine learning. In supervised learning, the algorithm creates a functioniality that translates inputs into expected outputs. A popular way to structure a supervised learning task is the classification problem, in which the learner has to examine several input-output instances of a function in order to comprehend (or estimate) its behavior. A vector is translated into one of several classes using the function. In unsupervised learning, a set of inputs is modelled instead of labelled instances. In semi-supervised learning, labelled and unlabeled cases are combined to build an appropriate function or classifier. The process by which an algorithm learns a policy about how to act in response to a world observation is known as reinforcement learning. Every action has an impact on the environment, and the environment provides feedback to the learning process. Similar to supervised learning, transduction generates predictions about new inputs, training outputs, and new inputs rather than explicitly developing a function. Learning to learn is the process by which the program creates its own inductive bias based on historical facts.

Computer Vision

Researchers in computer vision have been developing mathematical techniques to replicate the appearance and three-dimensional form of things in pictures. With the help of thousands of partially overlapping photos, we can now reliably create a partial 3D representation of an environment (Szeliski, 2022).

According to Fernandes, Dórea and Rosa (2020), computer vision systems can also be referred to as image systems, machine vision systems, or visual image systems. Consequently, computer vision uses image processing and analysis techniques because it is basically the construction of artificial systems to tackle relevant visual difficulties. Computer Vision is strongly related to Machine Learning and Pattern Recognition in addition to picture processing and analysis. Machine learning is a key component in computer vision that helps to extract important information from images. CV makes a significant contribution to several fields, including robotics, optical character recognition, surveillance systems, suspect detection, and many more (Khan, Laghari, & Awan, 2021).

A survey on publicly accessible data sets for computer vision tasks related to precision agriculture was conducted by Lu and Young in 2020. The absence of publicly available image datasets is a major barrier to the rapid development and evaluation of computer vision and machine learning algorithms for the intended purposes. Since 2015, a number of image databases have been developed and made available to the public in an effort to reduce this backlog. Publicly available image datasets are helpful for precision agriculture because they reduce the amount of work required to collect and prepare data and enable the development and evaluation of more efficient algorithms for various vision-related tasks.

Object Detection

According to Xuan et al. (2020), it is crucial to recognize the significance of object detection in fields like Computer Vision, Deep Learning, Artificial Intelligence, and other related areas of study. Their research illustrates how identifying objects is essential for

computer vision applications such as Target Tracking, Event Identification, Behavioral Analysis and Semantic Scheme Understanding (Xuan et al., 2020). Additionally, Yang Fu (2020) asserts that object detection is crucial in a number of domains, including as image processing, robotics, autonomous cars, satellite technology, and e-commerce.

In the study conducted by Abraham et al. (2023), the main goal of object detection is to recognize and categorize objects, in images or video frames. A lot of methods from artificial intelligence, machine learning, pattern recognition, and image processing are used in this work. Additionally, Abraham and colleagues noticed that object detection find use in a variety of domains, including the development of sophisticated human-computer interfaces, the monitoring of materials in industrial settings, traffic accident mitigation, and military surveillance on prohibited areas. The use of methods from artificial intelligence, machine learning, pattern recognition, and image processing is crucial to this effort. Abraham and associates further stated that object detection finds use in industries such as military surveillance, building sophisticated human-computer interfaces, reducing traffic accidents, and monitoring materials in industrial settings.

During his investigation, Rahaman (2023), argued that object detection is a type of machine learning. It incorporates recognizing and differentiating between item categories in visual media, such pictures and movies. "Localization of objects" requires precisely finding their spatial coordinates by enclosing them in bounding boxes, whereas "detection of instances" refers to identifying and labeling items in an image. Basically, object detection is important to computer vision because it makes it possible to locate and identify items in visual media. This has uses in many areas, including preventing traffic accidents, watching industrial materials, conducting military surveillance, and creating modern

human-computer interfaces. It entails recognising and differentiating between item categories in visual media, such as pictures and videos. "Localization of objects" entails precisely finding their spatial coordinates by enclosing them in bounding boxes, whereas "detection of instances" refers to identifying and labelling items in an image. In essence, object detection is a component of computer vision that makes it possible to locate and identify things in visual material, wherein there are practical uses for this.

Different Object Detection Algorithms

There are proficient algorithms in the field of object identification that we may use to advance technology, attain precision and efficiency, and produce the best outcomes possible given the particular application at hand. These object identification techniques are useful in almost every sector that uses vision, such as robots, driving, security cameras, and any virtual field that may need and rely on vision.

An object detection model called EfficientDet has been used. To get accuracy, it blends innovative detection techniques with a backbone network. Researchers have found that EfficientDet outperforms models in object detection and semantic segmentation by using parameters and computational operations (Tan et al., 2020).

A feature pyramid network and a focus loss function are used by the object identification learning system RetinaNet. The goal of this design was to directly address the disparity between foreground and background instances in object detection, hence increasing the process's accuracy. It's worth mentioning that this model is highly efficient in terms of resources making it a popular choice, for real time object detection tasks (Lin et al., 2018).

One type of network architecture created especially with object detection in mind is the Faster R CNN model. It locates items in an image by using a region proposal network (RPN). Subsequently, an additional network is utilised to classify these suggested areas and improve their coordinates. The precision of this model is greatly valued. It is widely used in the field of object detection, for both static images and videos (Ren et al., 2017).

A deep learning model called Mask R CNN was created especially for object detection. It incorporates object mask prediction, expanding upon the Faster R CNN architecture. In order to perform this, Mask R CNN integrates a network that creates pixel-level masks for every object that is recognized. It's worth noting that Mask R CNN has gained recognition for its accuracy in object detection and its practicality, for instance segmentation (He et al. 2018).

An efficient technique for detecting objects in several categories is the SSD (shot object detector). One key feature of this model is its ability to use bounding box outputs at scales and, across various feature maps located at the top of the network (Liu et al. 2016).

For real-time object detection, the You Only Look Once (YOLO) algorithm is regarded as state-of-the-art. It has shown its superiority over CNN based detection approaches by surpassing speed limitations while maintaining a balance between efficiency and accuracy (Handalage & Kuganandamurthy, 2021).

Therefore, from all the algorithms above, the most suitable algorithm to use for the application is the YOLO algorithm, since, As asserted by Handalage and Kuganandamurthy (2021), their scholarly work illuminates the manifold advantages of YOLO in practical application. Serving as a cohesive model for object detection, YOLO

proves to be effortlessly crafted and trained, aligning seamlessly with its straightforward loss function. This distinctive feature enables YOLO to concurrently train the entire model, emphasizing its efficiency in parallel processing. When it comes to object detection, the various YOLO versions provide the best possible balance between speed and accuracy. Moreover, YOLO outperforms object identification models in the creation of object representations. Given that it is the state-of-the-art algorithm for real-time object detection, it comes highly recommended.

YOLOv7 Object Detection Algorithm

The seventh version of the YOLO (You Only Look Once) object detection method is called YOLOv7. Wong Kin-Yiu and Alexey Bochkovsky, who also worked on YOLOv4 before creating YOLOv7 (Solawetz, 2022). In the YOLOv7 study examined by Wang, Bochkovsky, and Liao (2022), it is stated that YOLOv7 performs better than other object identification algorithms due to its superior speed, accuracy, and highest average percentage (AP). There are different sub-versions of YOLOv7 such as YOLOv7-Tiny, YOLOv7-X, YOLOv7-W6, YOLOv7-E6, and YOLOv7-D6 (Boesch, 2023).

Features of YOLOv7

The notable advancements in object detection and feature extraction of the YOLOv7 object detection algorithm have led to several projects and studies in various sectors. YOLOv7 is a fantastic choice for object recognition tasks since it successfully recovers object features from deep layers and has demonstrated strong performance in terms of mean of average precision (mAP). These enhance the YOLOv7's improved inference speed and detection accuracy through its new features

Wang, Bochkovskiy, and Liao (2022) introduced E-ELAN and Compound Model Scaling as two of the advanced architectural features within YOLOv7 object detection algorithm.

E-ELAN (Extended Efficient Layer Aggregation Network) is the computing block that forms the backbone of YOLOv7 that was developed based on the original ELAN. Group convolution is used by the E-ELAN to increase integrated feature cardinality. allowing the integration of the merge and shuffle cardinality techniques without modifying the gradient transmission channel.

The term "model scaling" for concatenation-based models describes a method used in YOLOv7 to modify the models' depth (number of layers), width (number of channels), resolution (size of the input image), and stage (number of pyramids). This makes it possible to create models that work with various pieces of hardware and software. Scaling the depth of calculation blocks and the breadth of the transition layers are both possible with compound model scaling. This method minimises the parameters and computation while maintaining the model's ideal structure and attributes.

YOLOv7 also has trainable Bag of Freebies (BoF) techniques. According to Shah (2020), BoF are techniques that increase performance while the training cost remained the same. Wang, Bochkovskiy, and Liao (2022) introduced two trainable Bag of Frebies (BoF) techniques: Re-Parameterized Convolution and Coarse-to-Fine Lead Guided Label Assignment.

A method known as Re-Parameterized Convolution (RevConv) seeks to enhance the model while keeping the inference cost low. Model level and module level re-

parameterization are the two approaches. It is possible to apply model level re-parameterization by using different training sets with identical settings. The average weight of each model training is the final model. The model training procedure is broken down into a number of modules at the module level. The final model is identified by the output of each module that has been gathered.

During training, a method known as Coarse-to-Fine Lead Guided Label Assignment attempts to assign labels to the outputs of various network branches. Key components of the method are the lead and auxiliary heads. The head in charge of the finished product is called the lead head. The head that supports training in the intermediate levels is called an auxiliary head. There are two procedures in the technique for assigning labels: coarse label assignment and fine label assignment. Reducing constraints results in a larger number of grids that are considered affirmative targets, which generates coarse labels. However, soft label produced during the training phase and utilised for output finalisation is also referred to as fine label.

The video anonymizer application relies heavily on the E-ELAN, compound model scaling, re-parameterized convolution, and course-to-fine lead guided label assignment features that were introduced by the YOLOv7 object identification method. This will help the suggested application's computational performance so that object identification and feature extraction may be completed with the appropriate level of accuracy and precision.

Comparison of YOLOv7 with previous versions of YOLO

YOLOv7 object detection is similar to its predecessors in features and has considerable enhancements. Several comparative studies have been conducted to test YOLOv7 and its predecessors to determine which YOLO algorithm has performed better.

A literature survey was authored by Nadhum and Nadhum (2023) toward YOLO object detection algorithms and its implementation on face recognition in COVID-19 pandemic. The proponents evaluated YOLOv1-v5 and YOLOv7 object detection algorithms. In the paper, it was stated that YOLOv7 can perform from 5 frames per second up to 160 frames per second, obtaining the highest AP of 56.8% compared to the other YOLO versions. The YOLOv7 trained on scratch COCO dataset. The proponents stated that YOLOv7 has perceived adaptive and can be implemented on lightweight models. The proponents commend the training process of YOLOv7 due to its convenience, in comparison to the other YOLO object detection algorithms. They suggested that the utilization of YOLOv7 to the deployment for COVID-19 face recognition is applicable.

K and Kanmani (2023) conducted a comparative study of different versions of YOLO object detection algorithm in detecting drones. All YOLO versions from v1-v7 are implemented with their own methodologies, utilized dataset, and image sizes. YOLOv7 utilized PyTorch framework, trained on Kaggle and COCO dataset with 640x640 image size and achieved 56.8% AP. While YOLOv6 achieved lower 43.3% AP, trained with COCO dataset with the same image size of 640x640. Results showed that YOLOv7 has higher precision/recall percentage and higher frames per second among the six previous versions of YOLO object detection algorithm. The proponents also commended YOLOv7's

better resolution scope compared to YOLOv3's 416x416. The proponents suggested that the implementation of YOLOv7 for drone detection is recommendable due to its performance in object detection.

Applications and uses of YOLOv7

The YOLOv7 object detection algorithm's speed and precision allow for a wide range of applications and purposes. applications in the fields of robotics, medical imaging, aviation, construction, and video analytics, among others.

Because YOLOv7 can analyse video in real-time, it can be applied to video analytics in fields like traffic monitoring, sports science, and surveillance. YOLOv7 was used as the object detection algorithm in a study by Rouf et al. (2023) on a real-time vehicle detection tracking and counting system. Important stages of the study's implementation included object detection, tracking, categorization, and counting. The improved YOLOv7, according to its proponents, has high precision tracking, detection, and counting capabilities. Another application of YOLOv7 in sports science video analytics. A study conducted by Martinez, Olivera, Aliaga, and Herrera (2023) implemented YOLOv7 and YOLOv7-Tiny object detection algorithms, and DeepSORT on their soccer ball detection and tracking system. The proponents used DeepSORT to accurately track the trajectory of balls by conducting two types of dataset training, namely supervised and semi-supervised training. The study emphasized the 95% ball detection accuracy that was attained. The algorithm's implementers claimed it performs well in tasks involving detection and classification. One of the integral functions in robotics is object detection. For enabling and adding functionalities, YOLOv7 is also integrated. A study conducted by Liu et al. (2023)

on *Camellia oleifera* fruit harvesting robot through trunk detection using improved YOLOv7 object detection algorithm. In order to compare their enhanced YOLOv7 with both the standard YOLOv7 and earlier YOLO versions, the proponents also assessed it. The enhanced YOLOv7 obtained mAP of 89.2%, 0.94 recall rate, 0.87 F1 score, and an average detection speed of 0.018 s/pic, according to the results. The enhanced YOLOv7, according to its proponents, demonstrated remarkable results in trunk detection and was quite accurate. They also claimed that the technology has the potential to be crucial to advanced forestry applications.

One important method for evaluating and diagnosing a person's interior bodily systems, organs, or tissues is medical imaging. Because of its accuracy and performance, the YOLOv7 object detection method is also essential for medical imaging applications. A study conducted by Abdusalomov, Mukhiddinov, and Whangbo (2023) integrated deep learning with the aid of their improved YOLOv7 and magnetic resonance imaging for brain tumor detection. Large picture datasets combining meningioma, pituitary, glioma, and non-tumor images had been used by the proponents. The findings demonstrated that their enhanced YOLOv7 obtained a 99.5% accuracy rate in brain tumour detection. The integration of their enhanced YOLOv7 with magnetic resonance imaging for brain tumour identification, according to the proponents, may help with brain tumour diagnosis and treatment.

Unmanned aerial vehicles (UAVs), in particular, when combined with computer vision and object recognition have a unique function in aviation that helps solve issues without the need for human intervention, particularly in high-altitude regions. A study conducted by Chang et al. (2023) toward the application of YOLOv7 object detection

algorithm and utilize it to multiple UAVs that aims to detect transmission line defects. Images of regular insulators, fractured insulators, insulator self-blast, and bird nests are included in the datasets' coverage. In order to compare YOLOv7 against alternative object detection algorithms, the proponents assessed it. YOLOv7 has an 87.6% F1 score, an 86.8% recall rate, and an 88.6% mAP, according to the results. YOLOv7 detects transmission line flaws better than the other object detection systems. The incorporation of YOLOv7 into several UAVs, according to its supporters, was a useful tool for detecting gearbox line problems and may have future uses.

Computer vision and object detection are tools that can be used in the construction sector to improve the infrastructure's quality and safety and avert future mishaps. A study was conducted by Wei et al. (2023) toward the implementation of a modified YOLOv7 utilized for detecting building façade defects. For the study, the proponents created their own enhanced YOLOv7 object detection system, called BFD-YOLO, which employs datasets of photos of building facades with three different kinds of flaws. Additional algorithms including YOLOv5-L, Faster R-CNN (region based Convolutional Neural Network), and RetinaNet were used to assess BFD-YOLO. According to the results, there was a 2.2% and 2.9% increase in accuracy and mAP@5 compared to the normal YOLOv7. The proponents claimed that their recommended algorithm can carry out real-time detection tasks and obtained excellent detection accuracy.

WIDER FACE Dataset

The Web Image Dataset for Event Recognition (WIDER) FACE dataset, comprising 32,203 images with 393,703 labelled faces, is a well-known dataset for face

detection. Yang et al. (2016) developed this dataset to address the limitations of current face detection datasets, which do not provide enough diversity in terms of pose, scale, facial expression, occlusion, and background clutters.

The WIDER FACE dataset is organized based on 61 event classes of the original WIDER Dataset (Xiong et al., 2015). 40% of the data for each event class is used for testing, 50% for training, and 10% for validation. This section enables a thorough assessment of face detection models in different contexts.

Use cases of WIDER FACE Dataset

The WIDER FACE dataset is distinguished by its size, being approximately ten times larger than other available datasets (Yang et al., 2016). Rich annotations, poses, event categories, occlusions, and face bounding boxes are also provided by this dataset, which is very useful for face detection algorithms.

The WIDER FACE dataset has been used by several studies to train their models, particularly on the YOLO architecture. For example, Qi et al. (2022) developed a YOLOv5 Face Detector that is trained using WIDER FACE dataset and compared it with other face detection algorithms. Another study of Khalili and Shakiba (2022) implemented a real-time face detection ensemble model by integrating the results of YOLO v1 to v4, using the WIDER FACE benchmark for training. The YOLOFace project also of Redmon (2019) employed the WIDER FACE dataset to train their pre-trained YOLOv3 weights for face detection.

Visual Studio Code

Microsoft produced Visual Studio Code (VSCode), a code editor available on multiple platforms with a proprietary license. It includes just minor modifications to the open-source, unrestricted Code – OSS code base. Johnson (2019), has stated that there are research studies focused on developing face recognition and anonymization techniques, keeping in mind the ethical and privacy concerns. Researchers have proposed algorithms and methodologies that can accurately detect and anonymize faces in images and videos. These techniques can be leveraged by a desktop application to provide an effective solution to anonymize faces in videos. Visual Studio Code has a large ecosystem of extensions and libraries that enable developers to enhance the functionality of their applications effectively. Various libraries related to image and video processing, facial recognition, and anonymization can be easily integrated into a VS Code project. These extensions and libraries can simplify the implementation of advanced functionalities required for face anonymization in videos.

Visual Studio Code is a great tool for developing a desktop application that can anonymize faces in videos. It is powerful and flexible, and it has a large ecosystem of extensions and libraries that can help developers build a robust solution quickly and easily. Visual Studio Code also makes it easy to emphasize the privacy and ethical considerations associated with face recognition technology.

Python

Python is a high-level programming language intended for more appropriate applications, including automation, data analytics, and software and web development.

Rouse (2022) stated that Python is a flexible programming language and can be utilized to program various types of functions. According to Python (n.d.), Because of its dynamic semantics, the programming language is suggested for use with the Rapid Application Development (RAD) methodology. Additionally, scripting and glue code that binds software components together are done using the Python programming language. Finally, because of its clear syntax, Python programmers are inexpensive to maintain. Python is a highly regarded programming language for use in artificial intelligence and machine learning applications. In the study conducted by Garg and Raheja (2022) toward assessment and selection of programming languages, Python is highly preferred compared to other object-oriented programming languages.

Lateef (2023) listed several factors on why Python is utilized as programming language for artificial intelligence (AI) projects: shorter code writings, existing libraries to implement, simple syntaxes, platform independence as Python can run on Windows, MacOS, Linux, etc., and huge community support. For artificial intelligence and machine learning projects, Python comes with pre-built object detection libraries. Python libraries were used by Krishna et al. (2022) in their work to analyze and alter data for object detection. Additionally, object detection benefits from Python's parallel processing capability. Python libraries were employed by Aziz et al. (2021) in their study on object detection by multiprocessing and parallel computing techniques.

The Python programming language is taken into consideration for the creation of desktop applications for video anonymizers due to its great versatility. Because this programming language includes pre-existing frameworks and libraries, developing a

desktop application takes less time when using it. Python is essential to front-end and back-end development because it makes applications easy to use.

PyTorch

PyTorch is an open-source framework for machine learning that is integral for computer vision applications and natural language processing. PyTorch has several techniques that can be applied to different tasks including transfer learning. Transfer learning refers to a technique that enables leveraging pre-trained model for the related task (Ramesh, 2019). There are two processes in transfer learning: fine-tuning and feature extraction. Fine-tuning refers to the removal of random initialization of the network due as the pre-trained model has initialized. And feature extraction refers to the freezing of layers' weight then the last layer will be utilized for training.

Multiple studies have utilized PyTorch machine learning framework for the development of their computer vision projects. In the study of Zaatouri and Ezzedine (2018), they utilized the transfer learning technique in PyTorch to implement YOLO object detection algorithm to their self-adaptive light control system. Another study conducted by Santos, Aguiar, Welfer, and Belloni (2022) utilized transfer learning to implement their YOLOv5 for fundus lesions detection. Gallo et al. (2023) also accomplished with transfer learning to implement YOLOv7 object detection algorithm for crop weeds detection.

The transfer learning technique of PyTorch framework is utilizable to these projects particularly in computer vision. The utilization of the framework will be integral to the implementation and training of the YOLOv7 object detection algorithm to the video anonymizer desktop application.

FFmpeg

Comprising software libraries for managing multimedia files, FFmpeg is an adaptable multimedia framework. Its main purpose is to process audio and video files. According to Kranzler, Herglotz and Kaup (2019) this multimedia framework is capable of doing multimedia tasks for video and audio such as processing, encoding, decoding, and transcoding, and streaming. Because of its many features, FFmpeg is best used for projects and research. Its extensive format conversion capabilities are one of its primary characteristics. The primary reason of utilization of FFmpeg is its capability on converting files to another format (Subhash, 2023). FFmpeg's benefits and drawbacks were enumerated by Cloudinary (n.d.). Being free and open source, compatible with many operating systems, and packed with functionality are just a few of FFmpeg's benefits. The sporadic issues that result in crashes and instability are a drawback.

Several FFmpeg framework implementations can be found in development projects. A study conducted by Zeng and Fang (2013) integrated FFmpeg to their MFC-based application that enables users to upload video and convert it into stream media server supported formats. Another implementation of FFmpeg framework developed by Cheng et al. (2012), an MFC-based application media player. Since FFmpeg is useful for managing media in a variety of ways, including importing, processing, and creating the final video output, it was thought to be an appropriate framework to use in the construction of the desktop video anonymizer application. It will help to shorten the time required to construct the desktop application by using this framework.

OpenCV

OpenCV is an open-source computer vision framework mostly used for object detection, feature extraction, and real-time image processing. According to Cristina (2023),s OpenCV is also commonly utilized in face recognition, object tracking, image registration and stitching, and augmented reality applications. Typical OpenCV algorithms include K-nearest Neighbours, Support Vector Machines, Decision Trees, and others. The OpenCV framework also supports PyTorch and Tensorflow. Twenty percent (20%) of OpenCV users are from the information and services sector, fourteen percent (14%) are from the computer software industry, and nine percent (9%) are from higher education, according to Enlyft (n.d.). The framework has been utilized by world-known companies such as Google, Yahoo, Microsoft, IBM, Sony, Honda, and Toyota (OpenCV, n.d.).

OpenCV is a framework that offers benefits and is being investigated for the creation of a desktop program for video anonymization. The YOLOv7 object detection technique used by video anonymizer will be implemented using OpenCV as the framework. This framework will help with the desktop application's development in terms of object detection capabilities.

CustomTkinter

CustomTkinter is a user interface library on Python based on Tkinter that enables creation of modernistic user interfaces with highly customizable widgets. It extends Tkinter by providing additional UI elements, such as the CTkButton, Schimansky (2024), has mentioned that it functions like a standard TKinter. Button but includes customizable features like borders and rounded edges. CustomTkinter offers fully customizable widgets

that can be created and used in the same manner as standard Tkinter widgets and can be seamlessly integrated with them.

Software Quality Evaluation

The ISO/IEC 25010, also known as "Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models", designate models comprising attributes and sub-aspects pertaining to the quality of software utilized as part with the software product. It also provides practical guidance on utilizing these quality models (Britton, 2021). This standard superseded the ISO/IEC 9126, which categorized software quality into six attributes (Rebeś, 2020). The ISO 25010 is more comprehensive than its predecessor, incorporating additional characteristics like security and compatibility that were not detailed in ISO 9126 (França & Soares, 2015).

According to ISO/IEC 25000:2005, a quality model (QM) is a framework consisting of a defined set of characteristics and their interrelationships, which is used to specify quality requirements and evaluate quality. This quality model forms the basis of a system for assessing product quality and determines the quality characteristics to be considered when evaluating a software product's qualities (Polillo, 2012). The product quality model, which encompasses both internal and external system qualities, consists of eight characteristics and thirty-one sub-characteristics.

The definitions of each characteristic are briefly outlined as follows.

Functional Suitability is the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions. This characteristic is composed of the following sub-characteristics:

- **Functional Completeness.** The degree to which the set of functions covers all the specified tasks and user objectives.
- **Functional Correctness.** The degree to which a product or system provides correct results with the needed degree of precision.
- **Functional Appropriateness.** The degree to which the functions facilitate the accomplishment of specified tasks and objectives.

Performance efficiency represents the performance relative to the number of resources used under stated conditions. This characteristic is composed of the following sub-characteristics:

- **Time Behavior.** The degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements.

Compatibility represents the degree to which a product, system or component can exchange information with other products, systems or components, and/or perform its required functions, while sharing the same hardware or software environment. This characteristic is composed of the following sub-characteristics:

- **Co-existence.** The degree to which a product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product.

Usability is the degree to which a product or system can be used by specified users to achieve specific goals with effectiveness, efficiency, and satisfaction in a specified context of use. This characteristic is composed of the following sub-characteristics:

- **Appropriateness Recognizability.** The degree to which users can recognize whether a product or system is appropriate for their needs.
- **Learnability.** The degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.
- **Operability.** The degree to which a product or system has attributes that make it easy to operate and control.
- **User Error Protection.** The degree to which a system protects users against making errors.
- **User Interface Aesthetics.** The degree to which a user interface enables pleasing and satisfying interaction for the user.

Security is the degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization. This characteristic is composed of the following sub-characteristics:

- **Integrity.** The degree to which a system, product or component prevents unauthorized access to, or modification of, computer programs or data.

Portability is the degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another. This characteristic is composed of the following sub-characteristics:

- **Installability.** The degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.

In this study, each component's acceptability is assessed in accordance with the ISO 25010 criteria. This encompasses functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, and portability. All these characteristics are expected to be relevant for this study.

Related Studies

The studies below were found relevant to the present study. Object detection in video streams is a crucial aspect of modern computer vision systems, finding applications in various domains. The process involves identifying and locating objects within individual frames of a video sequence. This entails detecting color and intensity variations that represent distinct physical objects. Additionally, the coordinates, dimensions, and other characteristics of these non-uniformities can be determined, enabling further tasks like object identification.

Puyda and Stoian’s study in 2017 delved into three algorithms for object detection, each employing different approaches: color non-uniformity detection, frame difference, and feature detection. Video streams serve as the input data, sourced from either a video camera or an MP4 video file. The algorithms were simulated and tested on a universal

computer powered by the Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit SoC processor operating at 1.5 GHz. The software was created on a universal PC running Linux (Raspbian Buster OS) for open-source hardware and using OpenCV 4 in Visual Studio 2019 on Windows 10.

Hukkelas, Mester, and Lindseth (2019) developed a novel architecture capable of automatically anonymizing faces in images while preserving the original data distribution. Their model generates images based only on privacy-safe information, achieving total anonymization of all faces in an image. The model, which is based on a conditional GAN, generates images while taking the initial position and image background into account. The development of extremely realistic faces with a smooth transition between the created face and the background is made possible by this conditional information.

The researchers trained their model on a wide variety of human faces, including faces in unusual positions, faces with obscured features, and faces against a variety of backgrounds. Experimental results show that the algorithm can anonymize images while maintaining the data distribution, making the anonymized data suitable for further training of deep learning models. Notably, this work stands out as the first solution to guarantee the anonymization of faces while generating realistic images.

Face anonymization is one of the key privacy protections for biologically identifiable human information that has garnered more attention in recent years. Ma, Li, Wang, and Dong developed a novel approach in 2021 that protects the facial image identity information from leakage with even the smallest alteration. In particular, the researchers use the power of generative adversarial networks trained on a conditional multi-scale reconstruction (CMR) loss and an identity loss to separate identity representation from

other visual features. In order to achieve face anonymization while preserving as much resemblance to the original image as feasible, the researchers assessed the model's detangle ability and created Anonymous Identity Generation (AIG), an efficient identity anonymization technique. The method's superiority over the SOTAs in terms of anonymization success rate and visual quality is demonstrated by both quantitative and qualitative data. The developed method is based on the idea of disentangling identity representation from other facial attributes. This is achieved by training a generative adversarial network (GAN) on a conditional multi-scale reconstruction (CMR) loss and an identity loss. The CMR loss forces the GAN to reconstruct the input image as accurately as possible, while the identity loss forces the GAN to disentangle identity representation from other facial attributes. This method has several advantages over the existing methods. First, the developed method can protect the identity information of facial images from being exposed with the slightest modification. Second, the developed method can maintain similarity to the original image as much as possible. Third, the developed method can achieve high visual quality.

The developed method has several potential applications. First, the developed method can be used to protect the privacy of individuals in social media applications. Second, the developed method can be used to protect the privacy of individuals in surveillance systems. Third, the developed method can be used to protect the privacy of individuals in access control systems.

Synthesis of Related Literature and Studies

According to the researchers' findings, anonymity is significant in protecting privacy in the media. The researchers proposed the development of a desktop application for anonymizing faces in videos using the YOLOv7 object detection algorithm. The study emphasized the importance of anonymity in protecting privacy and promoting self-disclosure, particularly in the context of social media applications, surveillance systems, and access control systems.

It also delved into the ethical concerns surrounding content creation, particularly as it relates to privacy issues affecting underaged children. The study underscored the necessity of anonymization methods and introduces the YOLOv7 object detection algorithm as a promising solution for video anonymization. This algorithm is renowned for its accuracy and efficiency in object detection tasks and has diverse applications in fields such as robotics, surveillance systems, and optical character recognition.

The study outlined the essential knowledge requirements and software and hardware components involved in developing the SEENSORED desktop application. It identified relevant technologies for GUI development, machine learning, video processing, and image recognition, including Visual Studio Code, Python, PyTorch, FFmpeg, OpenCV, Tkinter, and CustomTkinter. The study emphasizes the importance of evaluation to gauge the effectiveness and dependability of the developed desktop application.

The YOLOv7 object detection algorithm, employed in the SEENSORED application, holds significance in various fields beyond video anonymization. Its potential applications include autonomous vehicles, medical imaging analysis, and security

surveillance systems. The study underscores the ethical implications of using these technologies and the need for responsible implementation to protect individual privacy.

The development of the SEENSORED desktop application aims to anonymize faces in videos using the YOLOv7 object detection algorithm. The study highlights the significance of anonymity in protecting privacy and promoting self-disclosure. It explores the ethical issues surrounding content creation and the vulnerability of underaged children. The study also examines the applications of the YOLOv7 algorithm in various fields and highlights the necessary knowledge requirements and technologies for developing the desktop application. Evaluation is identified as a critical phase to assess the efficiency and reliability of the developed application.

Conceptual Model of the Study

The conceptual model of the study is depicted in Figure 1 using the Input-Process-Output diagram. This diagram gives an overview of the flow and the whole concept of the study.

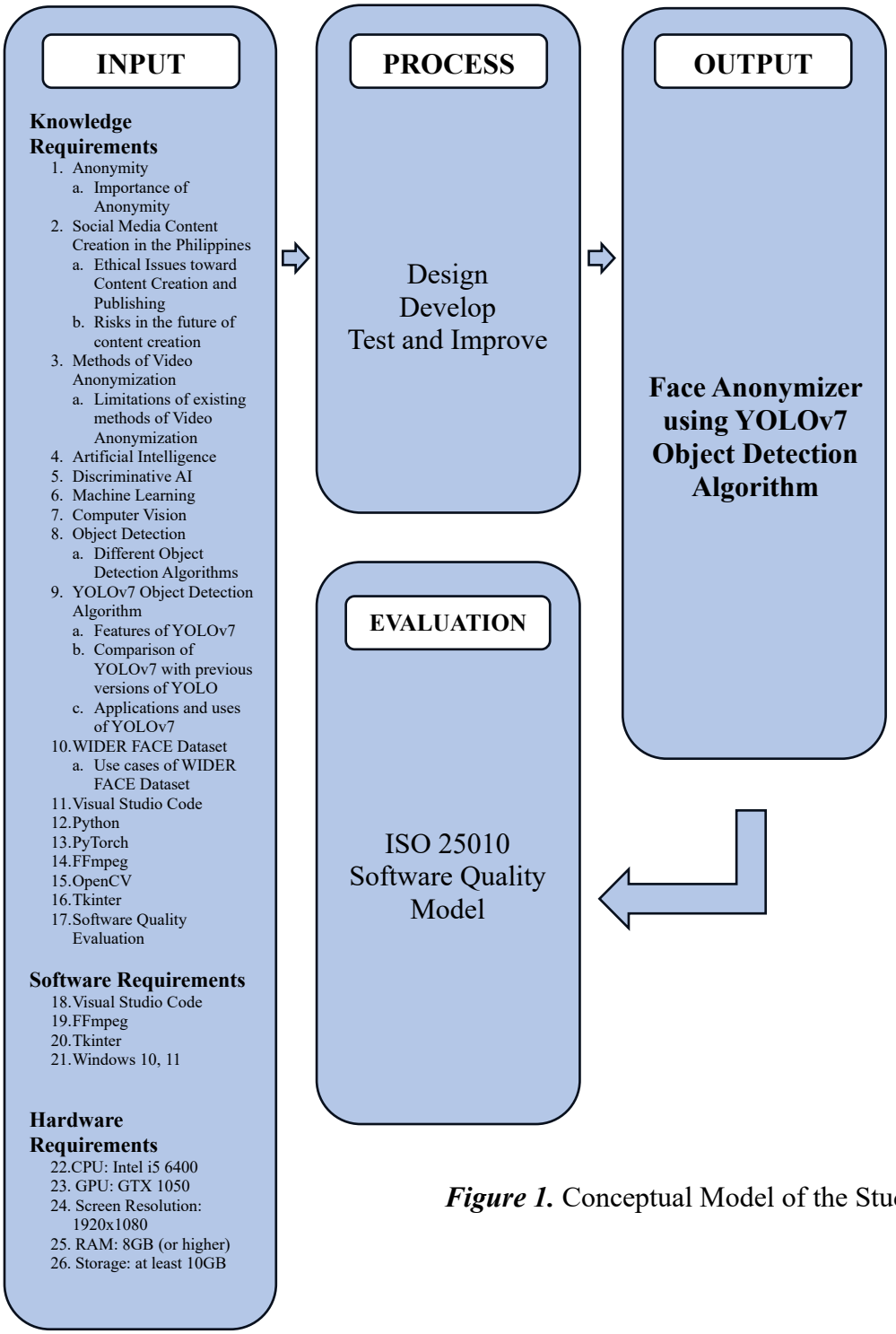


Figure 1. Conceptual Model of the Study

Input

The requisite knowledge, and required software and hardware are integral for the development of a desktop application for anonymizing faces in an input video. The knowledge requirements are the facts and concepts that the proponents needed to synthesize to formulate a concrete foundation toward the study. The following requisite knowledges are all about anonymity, content creation and its drawbacks, anonymization methods, artificial intelligence, discriminative AI, machine learning, computer vision, object detection, YOLOv7 object detection algorithm, WIDER face dataset, Visual Studio Code, Python language, frameworks such as PyTorch, FFmpeg, OpenCV, Tkinter, CustomTkinter, and ISO 25010 Software Quality Evaluation to meet the user requirements. Then, the software requirements for the development of front-end and back-end of the video face anonymizer desktop application including the utilization of a code editor, programming language, essential frameworks, and system platform. And the hardware requirements are the tangible items that are integral for the development and implementation such as computers and graphics cards.

Process

The process block contains the designing, developing, and testing tasks toward the development of the desktop application for anonymizing faces in an input video.

Design. In the designing phase, User Case Diagram is utilized to define the desktop application's expected behavior, functionalities, and software requirements. The utilization of flowchart is also considered for more efficient comprehension of the process flow of the desktop application.

Develop. In the development phase, the training, and the front-end and back-end development of the desktop application will take place. Visual Studio Code IDE (Integrated Development Environment) is the code editor for front-end and back-end development tasks. For graphical user interface, Tkinter, and CustomTkinter framework will be utilized. For the training, requisite knowledge is required to meet the desired precision and seamless object detection. YOLOv7 object detection algorithm will be utilizing PyTorch framework for transfer learning. OpenCV will be utilized for the object detection tasks of the desktop application.

Test. As the development of the desktop application has finished, tests will be undergoing to evaluate the accuracy in detecting faces in input video. Ensuring that the faces are anonymized evenly without errors and gaps. Standardization of the functionalities of the desktop application before subjecting it to evaluation.

Output

After the implementation of requisite inputs and processes depicted in the diagram, the output block shows the developed desktop application for video face anonymizer. The developed desktop application was subjected to evaluation to determine its efficiency and reliability.

Evaluation

The final block contains the evaluation phase wherein the desktop application will be evaluated by the stakeholders. The ISO 25010 assessment tool was utilized to test the desktop application’s software quality.

Operational Definition of Terms

The following terminologies are defined for a better understanding the study:

Anonymity refers to a state where people tend to hide or prevent any personal information from being exposed in either offline or online environments.

Anonymize refers to the process to hide the person's information.

Anonymization refers to the process of making a person to be anonymous.

Cache Integrity Checking refers to the stage before pre-processing wherein cache files are assessed if they are existent or tampered

Content Creation refers to an act of creating a subject or topic and making it into a digital presentation that is going to be posted on social media platforms.

Content Creator a refers to a person that makes the ideas for the subject or topic and makes it presentable for the audience in social media platforms.

Dataset refers to a set of images and annotations that can be used for training models.

Desktop Application is a software application that is commonly used by users utilizing a computer to do various duties.

Epoch is the circulation of training datasets.

Export is the process of outputting the video with anonymized faces that the user previously imported.

Import refers to the process of attaching a video file that a user wants to anonymize faces in.

Mass Media refers to the field of professional broadcasting and media dissemination

Object Detection refers to the process of detection of faces within the imported video file

Post-processing refers to the stage of application of video effects/face-blurring toward the selected faces on the imported video file

Pre-processing refers to the stage wherein face detection and cache creation undergo

Processed video refers to the final output with applied face-blurring effects

SEENSORED is an application where its main function is to detect possible faces in the environment and anonymize them.

Transfer learning refers to the method for the pre-trained model to be trained with custom datasets.

YOLOv7 is a model used for object detection that is well-known for doing computer vision task

Chapter 3

METHODOLOGY

This chapter entails the research methodology of the study with the following sections: project design, project development, operation and testing procedure, and evaluation procedure.

Project Design

The study aimed to develop a desktop application that aided content creation, designed to assist content creators in producing and publishing content that is ethical through the anonymization of unwanted individuals. The study utilized the YOLOv7 object detection algorithm for aiding face detection and the processing of anonymization in the desktop application.

Software Design

The scope of the desktop application software is represented through Use Case Diagram, illustrated in Figure 2. The diagram reveals the external entity which is the user that represents content creators and mass media producers which are expected to utilize the desktop application. Moreover, the diagram defined the expected behavior, functionalities, and requirements of the software.

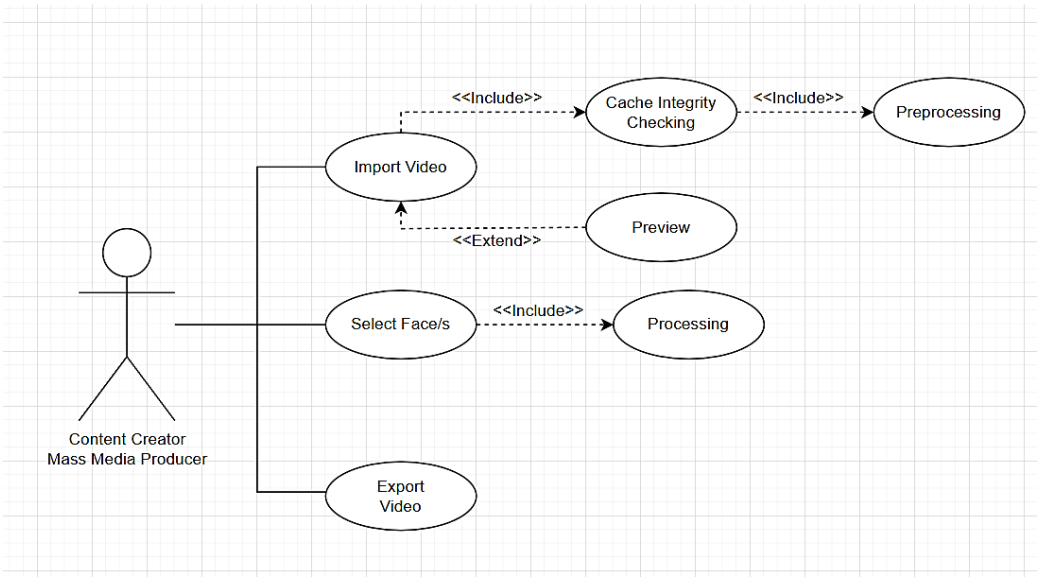


Figure 2. Use Case Diagram of SEENSORED Desktop Application

The following features are implemented on SEENSored Video Face Anonymizer desktop application in accordance with software requirements:

- Importation of video file
- Video file format validation
- Cache integrity checking
- Preprocessing of the imported video
- Selection of one or multiple faces
- Selection of blur types
- Anonymization of selected face/s
- Exportation of the video output

Flowchart

The process flow of the desktop application is represented through the flowchart, illustrated in Figure 2. The diagram reveals how the input, process, and output are sequenced accordingly to furtherly comprehend how the desktop application works.

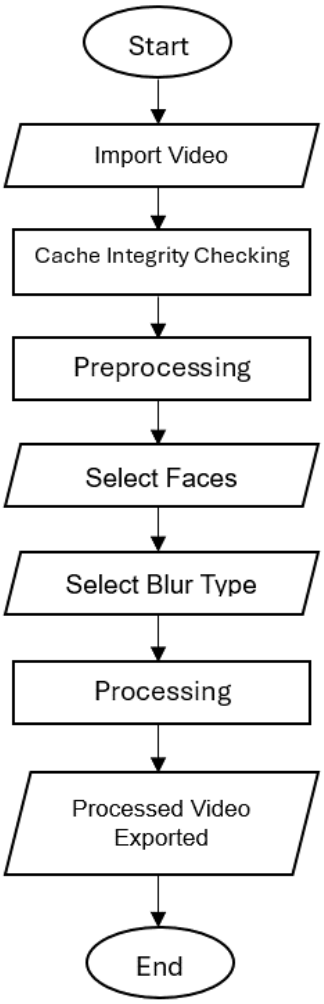


Figure 3. Flowchart of SEENSORED Desktop Application

The desktop application starts in initialization. Once the desktop application is ready, the user will be able to import a video file they desire to manipulate. There is a video format validation, wherein the video format accepted is only .mp4 format. In their research, Gloe, Fischer, and Kirchner (2014) observed that most videos resembling the .mp4 format in their database typically commence with compatibility with editing software. Given the versatility of MP4-like container formats, accommodating various data formats, codecs, and parameterizations.

Once the video file is imported, the cache integrity checking takes place. The hash will be generated based on the file. If the folder name with hash value as folder name does not exist, it will proceed to preprocessing. If there is one folder existing, it will be checking the label and face hash file. Next, it compares each hash in face hash file to the actual hash file of the images in the hash folder. If the conditions are met, it will be mark as okay. Otherwise, mark as tempered if not matched, or mark as missing if the image is not found. Then, the comparison of label hash file to the actual hash files of the labels in the labels folder. If the conditions are met, it will be mark as okay. Otherwise, mark as tempered if not matched, or mark as missing if the label file is not found. If there are missing or tampered files, it will undergo preprocessing. And if the files are verified, it will proceed face selection.

During preprocessing, the YOLOv7 object detection algorithm is responsible for the face detection. For tracking the detected faces, Simple Online and Real-time Tracking (SORT) algorithm is utilized. The desktop application will go through each frame of the video file to detect and track faces. Afterwards, it will generate face images and label files. Once all the generated images and label files are completed, it will generate a hash for each

file and save it in a .has file. Lastly, it will generate a preprocessed video file and a video thumbnail for later use in the Face Selection phase.

Project Development

Model Training

1. The WIDER FACE Dataset's 32,203 images have been split into training and validation, and testing data splits.
    a. The Training data split consisted of 12,879 images, covering 40% of the dataset.
    b. The Validation data split consisted of 3,225 images, covering 10% of the dataset.
    c. The Testing data split consisted of 16,099 images, covering 50% of the dataset.
2. The WIDER FACE Dataset has been converted into a format that YOLOv7 requires.
3. The model that was used for object detection is the YOLOv7-tiny. This model has undergone transfer learning with WIDER FACE Dataset for 100 epochs, using a batch size of 16 and 8 workers.
4. The trained model has been tested to check its accuracy and further improve the rating using Transfer Learning.
    a. To evaluate the prediction results of the trained model, the predicted result for each image in the testing data split has been submitted to the

authors of the WIDER FACE dataset. The authors have analyzed these results and provide feedback.

Program Coding

1. Installing the CustomTkinter library is required for making the interface of the project.
2. Create a user interface with 1920x1080 frame window fitting for a desktop application.
3. Create subframes based on the functions of UI for better management.
4. Make an asset folder for loading and implementing assets on the UI if needed.
5. Customize the colors of the UI.

Operation and Testing Procedure

Operation Procedure

The desktop application must be utilized in accordance with the following steps:

1. The user should open the desktop application.
2. The user should provide permission for the application to access the media files.
3. The application's user interface should be able to be shown to the user.
4. The application would be able to let the user choose from "Import video", "Blur/Anonymize Faces", and "Exit".
5. After selecting "Import Video", the user should choose the video file (mp4 file) to be imported to the desktop application.

6. Once the video file (mp4 file) has been imported to the desktop application, the desktop application would now process the imported video file for face detection using YOLOv7 Object Detection Algorithm.
7. The application should be able to show the detected faces with the bounding boxes.
8. The user would be able to select among the detected faces in a bounding box, to blur the detected faces, or remove the blur for a specific detected face.
9. The user would be able to export the video file with the processed blurred faces in the same video format (mp4 file).
10. The user would be asked to for the location of the processed video output. The application would process exporting the video output to the selected location.
11. After using the application, the user can choose the “Exit” option, or close the application.

Testing Procedure

The testing was based on the ISO 25010 software quality model. This assessment features systematically structured tables, each dedicated to scrutinizing specific criteria across the application's functionality, performance efficiency, compatibility, usability, and maintainability.

Table 1.

Functional Suitability Testing

Test Case	Steps Undertaken	Expected Results	Types
Blur a face from a video file.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video file and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application should be able to accept .mp4 video files. Upon finishing the detection of the faces in the video file, it would display the ID of the faces. The user should be able to select the faces that will be blurred and export them in the desired storage location.	Functional Completeness
Select faces to be blurred from a video file. Not blurring all detected faces.	1. Developed the desktop application by utilizing The YOLOv7 Object Detection Algorithm. 2. Opened the application. 3. Import a .mp4 video file to be processed. 4. Detected the faces in imported video file and create cache files of .mp4 video file. 5. Previewed of the detected faces with ID. 6. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 7. Exported processed video on desired storage location.	The desktop application should be able to accept .mp4 files only, should be able to detect faces, and blur accordingly to the user's desired outcome.	Functional Correctness
Blur faces and change the pitch tone to secure anonymity.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video files and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Changed the pitch tone of the mp4 video file. 6. Exported the processed video file to desired storage location.	The desktop application should be able to accept the imported .mp4 video files by the user, detect the faces, blur the selected detected faces, adjust the tone pitch. The user's configuration should be aligned to the desktop application's configuration	Functional Appropriateness

This table details the methodology for evaluating the functional suitability of the YOLO-derived video face anonymization tool. The assessment focuses on key functionalities, including video importation, precise facial detection, effective facial anonymization, and efficient video exportation. Each test case specifies the actions to be taken and the expected outcomes, aiming to verify the optimal performance of these essential features. This structured evaluation ensures that the tool meets the required standards for functional completeness, correctness, and appropriateness to meet user satisfaction.

Table 2.

Performance Efficiency Testing

Test Case	Steps Undertaken	Expected Results	Types
Compare efficiency and time it takes between traditional editing and the application in blurring the faces in a video file.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video file and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location. 6. Using a video editing software, manually blurred the faces in each frame 7. Compared the time it took for the task of blurring the faces.	The desktop application should be capable of processing the imported file, performing the face blurring process, and exporting the processed .mp4 video file. The time it will take to handle this task will be based on the number of frames in the imported .mp4 video file and the number of detected faces.	Time Behavior

This table outlines the evaluative procedures employed for gauging the operational efficiency of the application. It encompasses examinations of processing speed via the ingestion of sizable video files, alongside the scrutiny of temporal durations. Additionally, it incorporates the surveillance of resource deployment metrics, particularly those pertaining to the central processing unit (CPU) and memory utilization throughout the application's operational phase. The anticipated results aim to appraise the application's responsiveness and proficiency in resource management.

Table 3.

Compatibility Testing

Test Case	Steps Undertaken	Expected Results	Types
Share the application to other desktops.	1. Copied the application's files folder to a desktop's files. 2. Opened the application's files folder. 3. Ran the .exe file of the desktop application.	The developed desktop application should be able to run at a desktop if the folder package of the application has been copied completely. The desktop application should be able to perform even without the necessary tools used in developing, or the programming language installed.	Co-existence

The third table underscores compatibility assessment, scrutinizing the application's capacity to function across .mp4 video files in multiple Windows machines. Evaluation scenarios encompass the ingestion of films in assorted formats and subjecting the program to trials on multiple Windows machines. The anticipated results seek to affirm the application's versatility in diverse contextual settings.

Table 4.

Usability Testing

Test Case	Steps Undertaken	Expected Results	Types
User tests the application for news media industry.	1. Opened the application. 2. Imported a .mp4 video file to be processed. 3. Detected the faces in imported and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to a desired storage location.	The desktop application should be able to accept the imported .mp4 video files by the user, detect the faces, blur the selected detected faces, adjust the tone pitch. The user's configuration should be aligned to the desktop application's configuration.	Appropriateness Recognizability
Test the user to interact with friendly-user UI design of the application.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application should be user-friendly, allowing users to easily import .mp4 video files, detect faces, blur selected faces, and adjust the tone pitch. The application should intuitively align with the user's configuration preferences, ensuring a seamless and efficient learning curve for new users.	Learnability
Upload video files that are not in mp4 format.	1. Opened the application. 2. Import video files that are not the file type .mp4	The application will not accept any imported files that are not in the .mp4 format. The application will not proceed to post-processing if there are no selected faces to be blurred and the blurring type.	User Error Protection
Interact with the application's main frontend design.	1. Opened the application 2. Interacted with the buttons.	It will show the main screen of the application.	User-Interface Aesthetics

This table illustrates the intricate procedures involved in usability testing, specifically emphasizing the evaluation of the user interface, facial selection process, and video exporting method. The primary objective is to appraise the application's usability by scrutinizing the intuitive aspects of the UI, the user-friendliness of face selection for anonymization, and the straightforwardness of the video export process. Anticipated results center on confirming a positive and user-centric experience.

Table 5.

Security Testing

Test Case	Steps Undertaken	Expected Results	Types
Secure and anti-tampering of the cache folders.	1. Opened application 2. Imported Video 3. Generated of Cache Folder - containing preprocessed video 4. Tampered with values of hash inside folders	Upon importing of the video, a folder will be generated containing the cache files of the imported .mp4 video file. The detected faces will be saved in hash values inside the folder of cache files. Each detected face should match its hash value if one of the faces is tampered with or goes missing. The pre-processing will be conducted again to ensure that no errors will take place.	Integrity

Table 5 illustrates the evaluation protocol for Security Testing; mainly for file integrity, emphasizing the scrutiny of data protection measures, vulnerability assessment, and the application's capacity to detect and handle unauthorized modifications or corruption. Evaluation instances encompass examining and verifying the robustness of integrity checks. Tests include the intentional modification or corruption of files to gauge the application's response. The anticipated outcomes aim to validate the application's

security measures and ensure its ability to maintain data integrity without compromising performance.

The tables outline inspection methodologies aimed at furnishing a comprehensive assessment of the YOLO-based face anonymizer desktop application. This scrutiny encompasses a comparison of its operational efficacy, performance efficiency, compatibility, user-friendliness, and maintainability against the quality software benchmarks articulated in the ISO 25010 model.

Evaluation Procedure

The desktop application's evaluation involved a meticulous assessment by a diverse panel of 45 respondents comprised of content creators, mass media professionals, common users, and IT professionals. Employing the ISO 25010 Software Quality Model, a recognized framework for its methodical approach to software quality evaluation, this assessment covered various dimensions such as functionality, reliability, usability, efficiency, security, maintainability, and portability. Evaluators analyzed the application based on predefined criteria within the ISO 25010 framework, offering a nuanced perspective on its acceptability. This method ensures a comprehensive and holistic evaluation that considers the distinct needs and expectations of end-users, content creators, and IT professionals, thereby contributing to a thorough understanding of the desktop application's overall quality.

Table 6.

Likert Scale

Scale	Range	Adjectival Rating
4	3.26 – 4.00	Highly Acceptable
3	2.51 – 3.25	Very Acceptable
2	1.76 – 2.50	Acceptable
1	1.00 – 1.75	Not Acceptable

Chapter 4

RESULTS AND DISCUSSION

This chapter provides a thorough summary of the results that were discovered during this study project. Its contents include detailed investigation of the found insights, evaluations, and interpretations. It also provides a structure for combining these results into insightful conclusions that highlight the significance for the project's goals in a bigger context. This section also offers a series of carefully considered suggestions that are specifically designed to advance the project's expansion and advancement. By providing insightful information to guide future research initiatives, this chapter seeks to enhance the scholarly discourse surrounding the subject matter at hand through a comprehensive analysis of the findings, conclusions, and recommendations.

Project Structure

The study conducted aimed to develop a desktop application which enables an efficient editing software application for content creators and video editors. The desktop application’s main feature is to blur the detected faces in an mp4 input. The study has utilized the YOLOv7 object detection algorithm, by training the model to detect faces and blur the detected faces upon the user's decision. This methodology effectively eliminates the necessity for manually applying blurs to a face desired for each frame of the video, this task alone demands a time-intensive nature. By automating this process, considerable time and effort are saved, streamlining the overall workflow, and enhancing efficiency in face anonymization within a video content.

Figure 4. Application's Splash Screen

The project structure of the face blurring application begins at the initial File Selection Screen, as shown in Figure 5. The initial screen presents the option for the user to drop mp4 files or click the browse option to open an mp4 file via File Explorer. The user may choose the option below "Use custom order of faces" to identify the faces easier, and "Disable short face appearances" to disregard the short face appearances (faces that appear distorted).

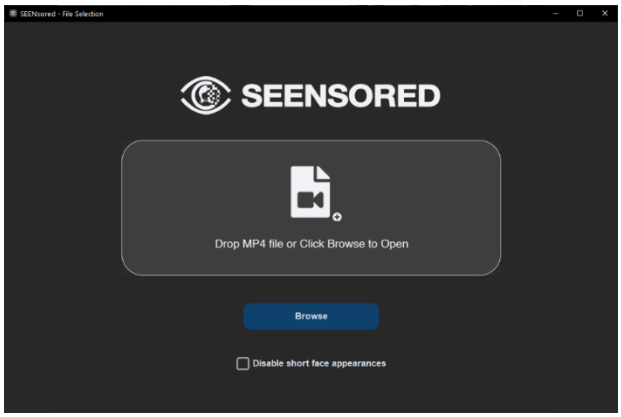


Figure 5. File Selection Screen

If the user has opted to select the "Browse" option, it opens the file explorer for the user to select the video file to be uploaded to the application. This process is shown in Figure 6. This process has validation since the application only accepts .mp4 files.

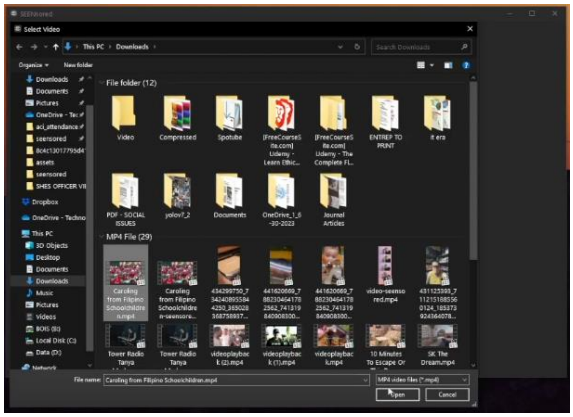


Figure 6. Selecting Video File

Once the user has successfully chosen a .mp4 file to be uploaded to the application, it would be imported to the application for the detection of the faces. If there is no cache files or corruption in the integrity of the cache files, it would proceed to the preprocessing or the detection of the faces. This process allows the creation of cache files, including thumbnails, and a preview video for Face Selection later. This process may take some time depending on the size of the .mp4 file. This process can be seen in Figure 7.

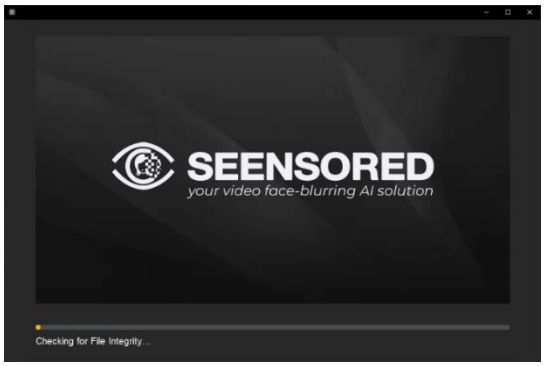


Figure 7. Preprocessing Loading Screen

After the detection process, with the help of the YOLOv7 object detection algorithm, all the faces detected in each frame in the video would be loaded up for the user

to select. This process is called the initialization of the faces, as seen in Figure 8. A preview of the video file can also be seen in this phase.

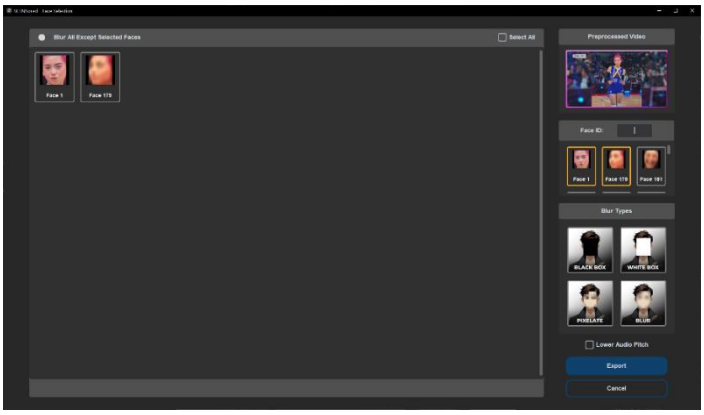


Figure 8. Face Selection Screen, without done action

If the user has chosen to play the preview of the video file, it would then proceed to open the video player application for the user to see. In this process, the detected faces would be shown in Figure 9, the face ID are also indicated for each detected face.

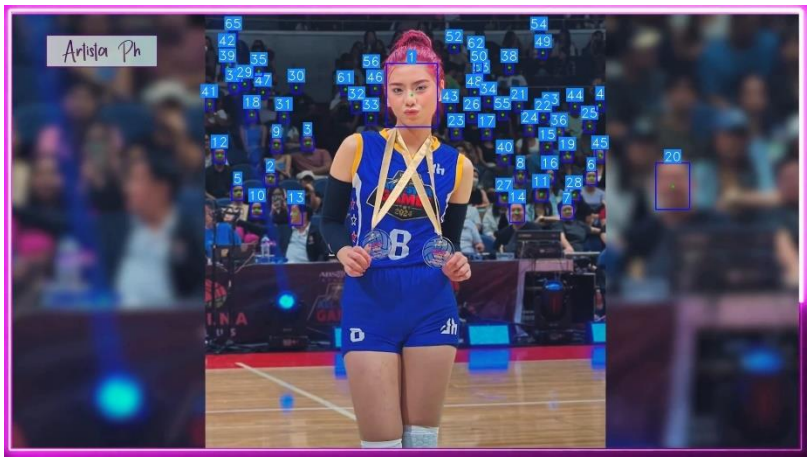


Figure 9. Preview of the Detected Faces

Once the preview of the video file is done, the application would then proceed to the editing screen. The user can now select from among the detected faces in the video to

be blurred. The user may manually select the faces that the user desires to blur. This process may take some time as it depends on the number of the detected faces on the video, but the feature of “Select All” allows all faces to be selected at the same time. The user may now choose among the blur types on the right side to fit in the desired blurring type. This is shown in Figure 10.

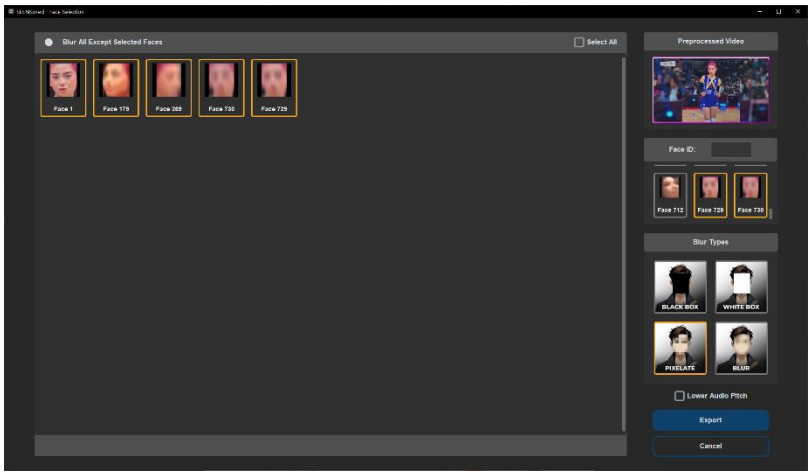


Figure 10. Face Selection Screen, with done action

If the user has successfully finished in selecting the faces to be blurred, and the blurring type, the user may now choose the “Export” button. This process would let the user choose the export path and renamed the processed video. This process is shown in Figure 11.

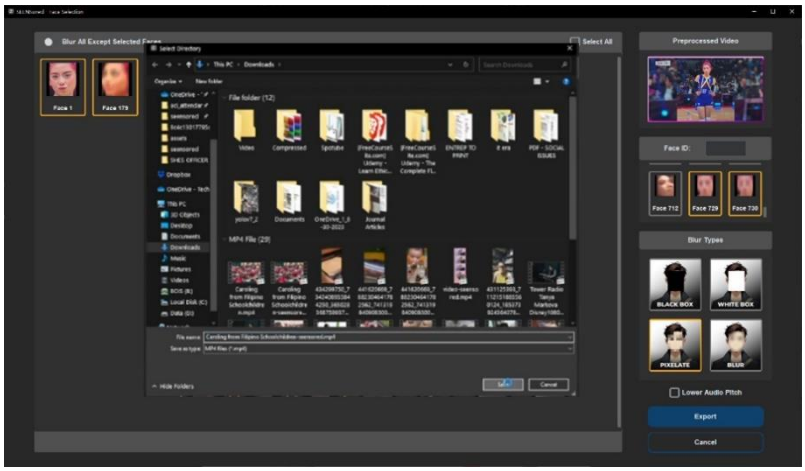


Figure 11. Choosing of Export File Name and Path

After selecting the export path for the processed video, and if the user selects both faces and blur type, it will proceed to postprocessing. As seen in Figure 12, a loading screen will pop-up that enables the user to track the progress of exportation.

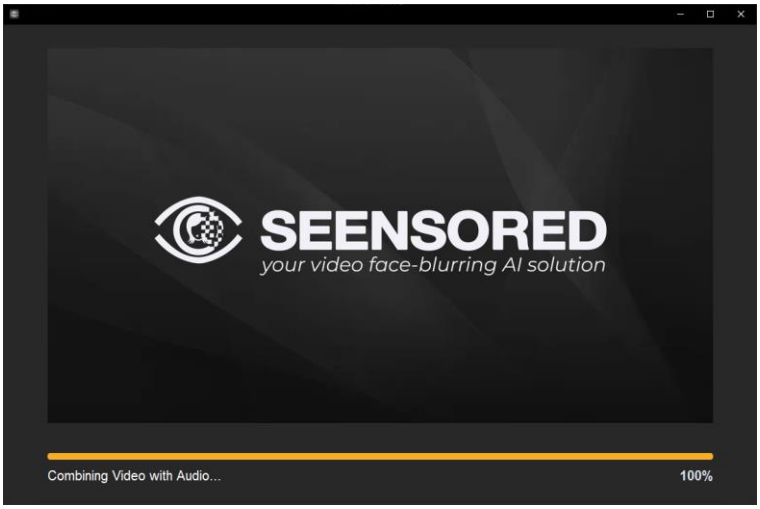


Figure 12. Post-processing Loading Screen

Once the loading of the video is successful, the application would proceed to post-processing loading screen. In this phase, the user can choose to view the processed video, it the user chooses this option, then it proceeds to open the video player to view the processed video. If the user has selected the “View Video Location”, the application shows the user the location of the processed video. While selecting the option “Get another video SEENsored” takes the application’s screen to Figure 5. This process can be seen in Figure 13.

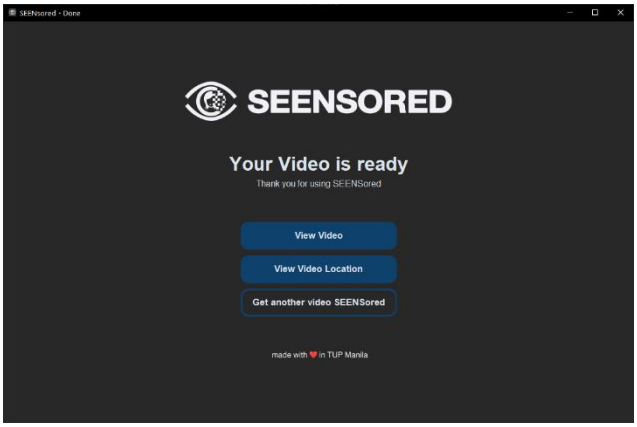


Figure 13. Done Post-processing Screen

The processed video with the detected faces being blurred are shown in Figure 14. The selected faces to be blurred remains blurred throughout the whole video, regardless of their frame appearance occurrence.

Figure 14. Processed Video with Custom Blurred Faces

Project Capabilities and Limitations

The following are the capabilities of the developed desktop application:

1. The project by the researchers aimed to develop a software application that is designed to aid the content creators, and video editors who are in the field of media and broadcast organization.
2. The desktop application automates the face-blurring process, the need for manual intervention in each video frame is eliminated, significantly reducing the time and effort required for face blurring within a video.
3. The blurring mechanism of the developed desktop application enhances workflow efficiency, allowing users to focus on other aspects of video editing or analysis.
4. The desktop application, with the YOLOv7 object detection algorithm, enables the detection of faces accurately, creating a bounding box for the detected faces for an efficient identification of faces within the video content for a user-friendly selection of faces.
5. The desktop application includes cache files that store the location of detected faces in each frame. These files contain essential information such as face ID, detected faces, and hash files. The hash serves to maintain the integrity of the files, preventing any unauthorized modifications to the crucial detection data, including file labels and face files.
6. The developed application can process the blurring from the importation of a video file. The FFmpeg framework has been utilized for video manipulation and transcoding operations. Additionally, CustomTkinter was

employed in GUI design, chosen for its widespread adoption in developing Python-based graphical user interfaces.

7. The application accurately identifies facial features and applies blur filters judiciously, ensuring minimal disruption from the original video file. Extensive testing and optimization procedures have been conducted to validate the application's efficacy across mp4 file formats.

The following are the limitations of the developed desktop application:

1. The desktop application is only limited to processing video from file importation only, it cannot process a real-time video. The user should have a recorded video to be able to blur a face using the application.
2. The file importation feature of the desktop application only accepts the .mp4 video file format. The file importation has a verification process on whether the imported file is accepted or not.
3. There is no database for the detected faces on the desktop applications, the algorithm will apply the cache files.

Test Results

Table 7.

Functional Suitability Testing

Test Case	Steps Undertaken	Observed Result
(Functional Completeness) Blur a face from a video file.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video file and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application successfully accepts .mp4 video files. After detecting the faces in the video file, it displays the IDs of the faces. Users can then select which faces to blur and export the processed video to their desired storage location.
(Functional Correctness) Select faces to be blurred from a video file. Not blurring all detected faces.	1. Developed the desktop application by utilizing The YOLOv7 Object Detection Algorithm. 2. Opened the application. 3. Import a .mp4 video file to be processed. 4. Detected the faces in imported video file and create cache files of .mp4 video file. 5. Selected the faces to be blurred, and the blurring type to be used for the selected faces. Exported the processed video.	The desktop application can accept .mp4 files exclusively. It effectively detects faces and applies blurring according to the user's specified preferences.
(Functional Appropriateness) Blur faces and change the pitch tone to secure anonymity.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video files and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application successfully accepts .mp4 video files imported by the user. It effectively detects faces, applies blurring to the selected faces, and adjusts the tone pitch as desired. The user's configuration aligns seamlessly with the desktop application's settings.

Table 8.

Performance Suitability Testing

Test Case	Steps Undertaken	Observed Result
(Time Behavior) Compare efficiency and time it takes between traditional editing and the application in blurring the faces in a video file.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported video file and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application effectively processes the imported file, performs the face blurring process, and exports the processed .mp4 video file. The processing time is determined by the number of frames in the imported .mp4 video file and the number of detected faces.

Table 9.

Compatibility Testing

Test Case	Steps Undertaken	Observed Results
(Co-existence) Share the application to other desktops.	1. Copied the application's files folder to a desktop's files. 2. Opened the application's files folder. 3. Ran the .exe file of the desktop application.	The developed desktop application successfully runs on a desktop when the application's folder package is copied entirely. It operates independently of the tools used in its development or the programming language installed on the desktop.

Table 10.

Usability Testing

Test Case	Steps Undertaken	Observed Results
(Appropriateness Recognizability) User tests the application for news media industry.	1. Opened the application. 2. Imported a .mp4 video file to be processed. 3. Detected the faces in imported and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 1. Exported the processed video file to desired storage location.	The observed functionality of the desktop application includes the ability to accept .mp4 video files imported by the user. It effectively detects faces, blurs the selected detected faces, and adjusts the tone pitch according to the user's preferences. The application ensures that the user's configuration aligns seamlessly with its own settings.
(Learnability) Test the user to interact with friendly-user UI design of the application.	1. Opened the application. 2. Import a .mp4 video file to be processed. 3. Detected the faces in imported and create cache files of .mp4 video file. 4. Selected the faces to be blurred, and the blurring type to be used for the selected faces. 5. Exported the processed video file to desired storage location.	The desktop application is designed to be user-friendly, providing a smooth experience for users to effortlessly import .mp4 video files, detect faces, blur selected faces, and adjust the tone pitch. It intuitively aligned with the user's configuration preferences, ensuring a seamless and efficient learning curve for new users.
(User Error Protection) Upload video files that are not in mp4 format.	1. Opened the application. 2. Imported video files that has the file type .mp4	The application did not accept any imported files that are not in the .mp4 format.
(User-Interface Aesthetics) Interact with the application's main frontend design.	1. Opened the application.	It showed the main screen of the application.

Table 11.

Security Testing

Test Case	Steps Undertaken	Observed Results
(Integrity) Secure and anti-tampering of the cache folders.	1. Opened application 2. Imported Video 3. Generated of Cache Folder - containing preprocessed video 4. Tampered with values of hash inside folders	Upon importing of the video, a folder was generated containing the cache files of the imported .mp4 video file. The detected faces are saved in hash values inside the folder of cache files. Each detected face matched its hash value if one of the faces are tampered or goes missing. The pre-processing has been conducted again to ensure that no errors will take place.

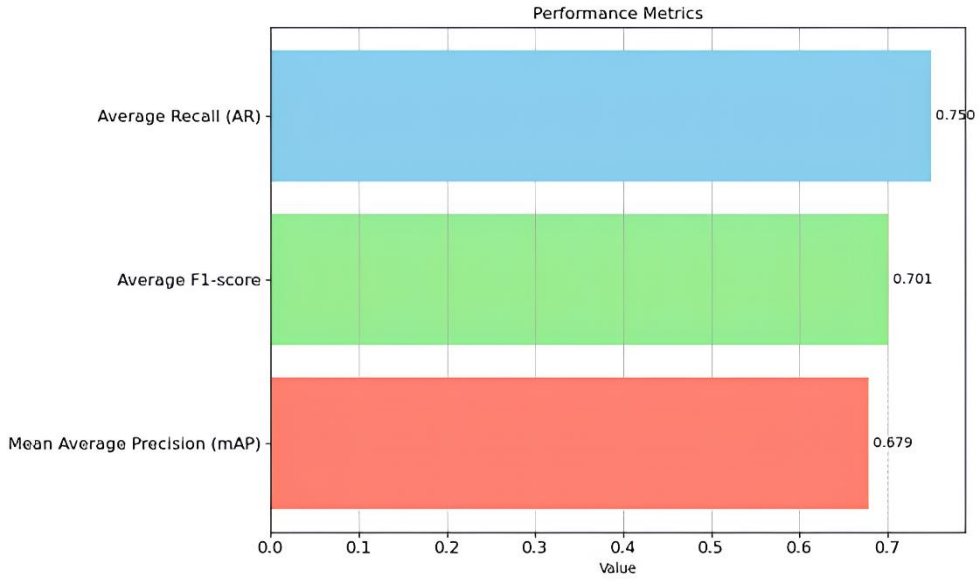


Figure 15. Performance Metrics of the Custom Trained YOLOv7 model

Figure 15 presents the Performance Metrics, which include a detailed analysis of the Average Recall, Average F1 Score, and Mean Average Precision. These metrics are critical in evaluating the effectiveness and reliability of the model.

- Average Recall: This metric measures the ability of the model to identify all relevant instances within a dataset.
- Average F1 Score: The Average F1 Score provides a balance between precision and recall, offering a single metric that considers both false positives and false negatives. This score is crucial for understanding the overall accuracy and reliability of the model in various scenarios.

- Mean Average Precision (mAP): Mean Average Precision evaluates the precision across different recall levels, offering a holistic view of the model's performance in identifying relevant instances.

The assessment conducted on the YOLOv7 model for face detection has revealed promising outcomes. With an average recall rate of 0.750, the model showcases its efficacy in identifying faces within the provided dataset, effectively capturing a considerable portion of the total faces present. The F1-score, standing at 0.701, signifies a well-balanced performance concerning both precision and recall, indicating the model's capability not only in accurately detecting faces but also in minimizing false positive identifications. Furthermore, the mean average precision (MAP) attaining a value of 0.679 further substantiates the model's robustness in precisely localizing faces across diverse scenarios. Collectively, these metrics endorse the YOLOv7 model's strong performance in face detection tasks, positioning it as a viable option for real-world applications where precise and efficient face detection holds utmost importance.

Evaluation Results

The desktop application was evaluated to determine its acceptability. The evaluators were purposely selected content creators and mass media professionals who are the primary users. Also, common users and IT professionals were also included in the evaluation. Table 12 depicts the summary of evaluation.

Table 12.

Summary of Evaluation

Criteria	Weighted Mean	Description
A. Functional Suitability		
1. Functional Completeness	3.69	Highly Acceptable
2. Functional Correctness	3.69	Highly Acceptable
3. Functional Appropriateness	3.73	Highly Acceptable
Criterion Weighted Mean	*3.70*	*Highly Acceptable*
B. Performance Efficiency		
1. Time Behavior	3.60	Highly Acceptable
2. Resource Utilization	3.67	Highly Acceptable
3. Capacity	3.56	Highly Acceptable
Criterion Weighted Mean	*3.61*	*Highly Acceptable*
C. Compatibility		
1. Coexistence	3.58	Highly Acceptable
2. Interoperability	3.64	Highly Acceptable
Criterion Weighted Mean	*3.61*	*Highly Acceptable*
D. Usability/Interaction Capability		
1. Appropriateness Recognizability	3.64	Highly Acceptable
2. Learnability	3.67	Highly Acceptable
3. Operability	3.71	Highly Acceptable
4. User Error Protection	3.42	Highly Acceptable
5. User Interface Aesthetics	3.51	Highly Acceptable
6. Self-Descriptivity	3.67	Highly Acceptable
Criterion Weighted Mean	*3.60*	*Highly Acceptable*
E. Reliability		
1. Maturity	3.57	Highly Acceptable
2. Availability	3.49	Highly Acceptable
3. Fault Tolerance	3.49	Highly Acceptable
4. Recoverability	3.47	Highly Acceptable
Criterion Weighted Mean	*3.50*	*Highly Acceptable*
F. Security		
1. Integrity	3.60	Highly Acceptable
Criterion Weighted Mean	*3.60*	*Highly Acceptable*
G. Maintainability		
1. Modularity	3.53	Highly Acceptable
2. Reusability	3.58	Highly Acceptable
3. Analyzability	3.64	Highly Acceptable
4. Modifiability	3.62	Highly Acceptable
5. Testability	3.60	Highly Acceptable
Criterion Weighted Mean	*3.59*	*Highly Acceptable*
H. Portability/Flexibility		
1. Adaptability	3.58	Highly Acceptable
2. Scalability	3.67	Highly Acceptable
3. Installability	3.71	Highly Acceptable
Criterion Weighted Mean	*3.65*	*Highly Acceptable*
Grand Weighted Mean	**3.61**	**Highly Acceptable**

As shown in Table 12, the desktop application obtained its highest rating under Functional Suitability with an overall weighted mean of 3.70 and interpreted as Highly Acceptable. This implies that the desktop application had a very satisfactory performance. Under Functional Suitability, the desktop application's function performed its requirements to meet user needs.

The desktop application obtained its second highest rating category under portability with an overall weighted mean of 3.65 and interpreted as Highly Acceptable. This implies that the desktop application could adapt and transferred to different environments, handle scales of workload, and installed or removed in a specified environment.

The desktop application obtained its lowest rating on Reliability with an overall weighted mean of 3.50 and interpreted as Highly Acceptable. This implies that the desktop application still highly reliable but could have some more needed requirements as improved error handling and comprehensive testing to ensure the application's reliability.

Overall, the desktop application obtained a grand weighted mean of 3.61 and interpreted as Highly Acceptable which suggests that the SEENSored desktop application has the capabilities to perform its tasks that are integral for content creation and can be used by a diverse group of users by maintaining ethical and legal standards of content creation.

Chapter 5

SUMMARY OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS

This chapter provides a comprehensive summary of the results, discusses the conclusions derived from the evaluations and test outcomes, and offers recommendations for further improvements and advancements of the project.

Summary of Findings

Based on the test and evaluation results, the developed desktop application entitled "SEENSORED: Face Blurring Using YOLOv7 Object Detection Algorithm", the desktop application could be the efficient solution for blurring faces in a .mp4 video file providing convenience for content creators or editors in the industry of mass media. The following is a summary of the findings using the desktop application:

- Test Results
    1. **Functional Suitability**: The evaluators found the desktop application to be highly acceptable in terms of its responsiveness and overall functionality.
    2. **Performance Efficiency**: The application was rated highly acceptable for its efficiency and fast response times during most user interactions.
    3. **Compatibility**: The application received high ratings for its compatibility with various desktop products and was able to perform on different desktops.
    4. **Usability/Interaction Capability**: Evaluators deemed the application highly acceptable for meeting user needs, highlighting its user-friendly

5. interface, clear navigation, intuitive controls, and easy-to-understand instructions.
6. **Reliability**: The application was rated highly acceptable for its reliability, proving to be stable and dependable with no major crashes or errors during testing.
7. **Security**: The application received high marks for its security, having implemented the file integrity.
8. **Maintainability**: Evaluators rated the application highly for its good analyzability and modifiability, allowing for easy improvements and corrections.
9. **Portability/Flexibility**: The desktop application was rated highly acceptable for its ease of use and compatibility with a wide range of desktops as it was able to run despite not having the necessary tools or language for developing it, contributing to a positive user experience.

- Evaluators' Highlights

In relation to the Evaluation Procedure conducted consisting of students, IT professionals, and media journalist, here are a few statements from them. With the evaluator's consent, all statements are shared on this article and research within and are based on unaltered transcripts of their responses in the survey and voice recordings.

The IT professional (See Appendix C.1), following the evaluation highlighted that the application's ease of operability stands out as a key feature, significantly aiding editors and content creators in efficiently censoring faces within video content. The IT

professional stated that the intuitive interface allows individuals with minimal to no experience in video editing to use the software effectively. Additionally, the IT professional further recommended optimization to reduce hardware resource consumption would be beneficial, ensuring that even low-spec laptops and desktops can fully utilize the software's functionality.

The second IT professional (See Appendix C.2) has commended the application for its potential to raise awareness about the importance of privacy among social media users and content creators. The second IT professional mentioned about streamlining the process of ensuring individuals' security in digital content, the application significantly reduces the effort required to protect privacy. This innovation promises to be a valuable tool in promoting privacy-conscious behavior. The evaluator sees the application’s ensuring of its accessibility to a wide audience would further enhance its impact, making it an indispensable resource for anyone concerned with digital security.

The third IT professional (See Appendix C.3) provided a significant feedback that the application exhibits considerable potential; however, several enhancements would significantly augment its functionality and user experience, particularly for media creators. Firstly, expanding the application’s capability to support a variety of video file formats beyond MP4 is crucial. Media creators often work with diverse formats, and limiting the application to MP4 alone restricts its utility. Secondly, the analysis screen, where objects to be masked are identified, would benefit from displaying the confidence levels of scans. Categorizing low-probability scans separately would provide users with a clearer understanding of the results, enhancing the decision-making process regarding which objects to mask. Lastly, the inclusion of sample low-

resolution videos for testing purposes would be a valuable addition. This feature would allow users to familiarize themselves with the application's capabilities and performance before applying it to their own content, thereby improving the overall user experience and satisfaction.

The journalism student, as referenced in (See Appendix C.4), emphasized the critical importance of protecting the privacy of underaged individuals by using the Seensored application to blur faces. This feature brings significant convenience, especially when dealing with videos containing numerous faces. The student noted that in the fast-paced world of news reporting, the application helps ensure that news can be made readily available while still respecting individual privacy. Furthermore, the student highlighted that this application is vital for ethical journalism, as it allows users to efficiently select and blur faces without needing to redo the entire process, thus saving time and effort. The ability to save these edits in one attempt was particularly commended, as it enhances the practicality and usability of the software for journalists who must adhere to ethical standards while working under tight deadlines.

The second journalism student, as referenced in (See Appendix C.5), praised the application's efficiency and its suitability for the news media industry. The student highlighted the pressing need for rapid editing in news production, where timely delivery is paramount. The application's feature that allows for the quick blurring of all or selected faces was particularly commended, as it significantly enhances the speed and efficiency of the editing process, meeting a critical requirement in the fast-paced news environment. This capability is especially valuable in ensuring that sensitive information is protected without delaying the dissemination of news. The student's

feedback underscores the application's potential to address a crucial gap in the industry, making it an indispensable tool for journalists who must balance accuracy, privacy, and expedience.

Conclusions

The following conclusions were drawn from the findings of the conducted study:

1. File Importation
    a. The application allowed the user to import a .mp4 file, and the application would only accept .mp4 video files only.
    b. This created an application that is error-free as any other file types are rejected, as the desktop application only accepts .mp4 video files.
2. Pre- Face Selection properties
    a. The application enabled the user to choose between organizing the faces in a custom order based on the detected similarities among the faces or arranging them according to their assigned Face ID. This feature allows for a more personalized and intuitive user experience by offering different methods of face arrangement depending on the user's preference.
    b. It also enabled the user to disregard the faces that appear and disappear briefly for a more organized and cleaner method of choosing the faces to blur.
3. Face Selection
    a. The application effectively detected faces and displays their actual Face ID in the face selection window, ensuring accurate identification and easier management of faces.

b. A preview option allowed users to view the detected Face IDs throughout the entire video, helping them decide which Face IDs to apply the blur effect to.

c. Users may utilize a "Select All" option to conveniently choose all detected Face IDs if they wish to blur all faces in the video.

d. The application offered users a choice of four blur types, including black box, white box, pixelation, and standard blur, allowing for versatile customization of the blurring effect.

e. The application included a lower pitch option, which reduces the pitch of the original video, altering the audio to achieve a deeper or lower tone.

f. Users can define a personalized pathway for the output video, granting them the ability to designate a specific location or directory where the processed video is stored.

4. Post processing (rendering)

    a. The output is the video with blurred faces, ensuring that the configuration aligns with the selections made by the user during the Face Selection Phase, including:

        I. Identified Face IDs chosen for blurring.

        II. Selected blur type.

        III. Preference for lowering pitch, if applicable.

        IV. Customized output path specified by the user.

    b. Users were given the choice to either watch the output video, access the folder where it's stored, or opt to edit another video for face blurring,

providing them with flexibility and convenience in managing their editing tasks.

5. The successful development of the desktop application involved the utilization of various tools and technologies. These tools included the implementation of a machine learning algorithm for face detection, employing the YOLOv7 object detection algorithm. Python served as the primary programming language for the development process. Additionally, frameworks such as OpenCV and FFmpeg were integrated into the application to enhance its functionality. CustomTkinter was also utilized as part of the development toolkit, contributing to the overall effectiveness of the desktop application.
6. The desktop application successfully fulfilled its purpose by serving as a tool to contribute to a safer online community in content creation. It enhances efficiency for editors in the content creation and mass media industries, streamlining their workflows and ensuring higher standards of safety and quality in the produced content.
7. The test results indicated that the desktop application is highly functional as it has comprehensive set of features, accuracy in performing detection, and the relevance of its functionalities on user requirements.
8. The developed desktop application was evaluated to be highly acceptable in terms of Functional Suitability, Performance Efficiency, Compatibility, Usability/Interaction Capability, Reliability, Security, Maintainability, and Portability/Flexibility. This evaluation demonstrates that the desktop application can be very helpful and effective in editing faces to blur them in a video.

Recommendations

The following recommendations are proposed for further enhancement of the developed desktop application:

1. The model can be improved by training it on a more diverse dataset that includes various head images. The model's dataset could be improved if it encompasses a range of variations, such as different head angles relative to the camera,
2. One of the improvements is the tracking of the head of a person in varying lighting conditions, reducing the likelihood of misidentification and increasing the overall accuracy and robustness of the application.
3. The distances from the camera, and different hairstyles, as it can lead to some people recognizing someone. These enhancements help the model to better recognize and handle a wider array of head images and increase the effectiveness in hiding the identity of the person being blurred.
4. Training from another dataset that includes blocking on different angles that partially cover the face. This would ensure the face detection and blurring despite the slight obstruction on the face.
5. The whole cache should not be reset; instead, only the affected file or a tampered file should be repaired to maintain the integrity feature of the application.
6. The feature of detecting similar faces can be improved by employing better similarity comparison algorithms, ensuring that identical or similar faces are recognized and processed as a single entity.

REFERENCES

Abdusalomov, A. B., Mukhiddinov, M., & Whangbo, T. K. (2023). Brain tumor detection based on deep learning approaches and magnetic resonance imaging. Cancers, 15(16), 4172. DOI: https://doi.org/10.3390/cancers15164172

Abraham, A., Hanne, T., Gandhi, N., Pooja Manghirmalani Mishra, Bajaj, A., & Siarry, P. (2023). Proceedings of the 14th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2022). Springer Nature. https://pdfdrive.to/filedownload/proceedings-of-the-14th-international-conference-on-soft-computing-and-pattern-recognition-socpar-2022

Abraham, A., Hanne, T., Gandhi, N., Manghirmalani Mishra, P., Bajaj, A., & Siarry, P. (2023). Proceedings of the 14th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2022) (pp. 485–486). Springer, Cham. https://doi.org/10.1007/978-3-031-27524-1_1

Amos, D. (2023, January 30). Python GUI programming with Tkinter. Retrieved from: https://realpython.com/python-gui-tkinter/

Arali, R., M, S., P, S., & Halawal, V. (2021). Colour Detection from Image. *International Journal of Advances in Engineering and Management (IJAEM)*, *3*(7), 2164–2171. DOI: https://doi.org/10.35629/5252-030721642171

Arshad, U. (2021). Object Detection in Last Decade - A Survey*. Scientific Journal of Informatics, 8(1), 60–70. https://journal.unnes.ac.id/nju/index.php/sji/article/view/28956/pdf

Association for Progressive Communications (APC). (2015). The right to freedom of expression and the use of encryption and anonymity in digital communications. Retrieved from https://www.ohchr.org/sites/default/files/Documents/Issues/Opinion/Communications/AssociationForProgressiveCommunication.pdf

Ayodele, T. O. (2010). Types of machine learning algorithms. New advances in machine learning, 3, 19-48.

Aziz, Z. A., Abdulqader, D. N., Sallow, A. B., & Omer, H. K. (2021). Python parallel processing and multiprocessing: A Review. *Academic Journal of Nawroz University*, *10*(3), 345–354. DOI: https://doi.org/10.25007/ajnu.v10n3a1145

Balaji, T., Blies, P., Gori, G., Mitsch, R., Wasserer, M., & Schon, T. (2021). Temporally coherent video anonymization through GAN inpainting. Retrieved from https://www.researchgate.net/publication/306363755_Impact_of_Anonymity_and_Identity_Deception_on_Social_Media_eWOM

Balaji, T., Blies, P., Gori, G., Mitsch, R., Wasserer, M., & Schon, T. (2021). Temporally Coherent Video Anonymization through GAN Inpainting. https://www.researchgate.net/publication/352177723_Temporally_coherent_video_anonymization_through_GAN_inpainting

Beijbom,O. (2023, July). Discriminative vs. Generative AI: Which Should You Build Into Your Product?. Nyckel. https://www.nyckel.com/blog/building-ai-into-your-product-understand-the-difference-between-discriminative-and-generative-ai/

Boesch, G. (2023). YOLOv7: The Most Powerful Object Detection Algorithm (2024 Guide). Visio.AI. Retrieved from: https://viso.ai/deep-learning/yolov7-guide/

Boyle, M., Neustaedter, C., & Greenberg, S. (2003). Privacy factors in video-based media spaces. In Media Space 20 + Years of Mediated Life (pp. 97–122). https://link.springer.com/chapter/10.1007/978-1-84882-483-6_7

Busacca, A., & Monaca, M. A. (2023). Deepfake: creation, purpose, risks. In Studies in systems, decision and control (pp. 55–68). DOI: https://doi.org/10.1007/978-3-031-33461-0_6

Caparas, J. (2023, November 13). Social Media in the Philippines 2023: Facts, Statistics, History & Trends. Spiralytics. Retrieved from: https://www.spiralytics.com/blog/social-media-in-the-philippines/

Carrelo, C. (2022). YouTube Family Vlogging as a Promoter of Digital Child Labour: A Case Study on 'The Bucket List Family' [MA Thesis]. Malmö University.

CEOP Education. (n.d.). Social media community guidelines. Retrieved from: https://www.thinkuknow.co.uk/11_18/lets-talk-about/online-safety/community-guidelines/

Chang, R., Xiao, P., Wan, H., Li, S., Zhou, C., & Li, F. (2023). A transmission line defect detection method based on YOLOV7 and Multi-UAV collaboration platform. *Journal of Electrical and Computer Engineering*, 2023, 1–14. DOI: https://doi.org/10.1155/2023/9943589

Cheng, Y., Liu, Q., Zhao, C., Zhu, X., & Zhang, G. (2012). Design and Implementation of Mediaplayer Based on FFmpeg. Software Engineering and Knowledge Engineering: Theory and Practice, 867–874. DOI:10.1007/978-3-642-25349-2_114

Chua, K. (2021, January 31). PH remains top in social media, internet usage worldwide – report. Retrieved from: RAPPLER. https://www.rappler.com/technology/internet-culture/hootsuite-we-are-social-2021-philippines-top-social-media-internet-usage/

Chua, Y. (2022). Philippines. Reuters Institute for the Study of Journalism. Retrieved from: https://reutersinstitute.politics.ox.ac.uk/digital-news-report/2022/philippines

Cloudinary. (2023). FFMPEG: Features, Use Cases, and Pros/Cons You should know. Retrieved from: https://cloudinary.com/guides/video-formats/ffmpeg-features-use-cases-and-pros-cons-you-should-know

Cristina, S. (2023, October 21). A gentle introduction to OpenCV: an open source library for computer vision and machine learning. MachineLearningMastery.com. Retrieved from: https://machinelearningmastery.com/a-gentle-introduction-to-opencv-an-open-source-library-for-computer-vision-and-machine-learning/

De Vera, M. J. B., & Saludadez, J. A. (2021). YouTube vlogging as access to possibilities: A phenomenological study of an OG Filipino vlogger. Plaridel, 18(2), 1-27. https://doi.org/10.52518/2021-17dvesal

Enlyft. (n.d.). OpenCV commands 2.72% market share in Artificial Intelligence. Retrieved from: https://enlyft.com/tech/products/opencv

Fernandes, A. F. A., Dórea, J. R. R., & Rosa, G. J. D. M. (2020). Image analysis and computer vision applications in animal sciences: an overview. Frontiers in Veterinary Science, 7, 551269.

Fire, M., Goldschmidt, R., & Elovici, Y. (2014). Effectiveness and users' experience of face blurring as a privacy protection for sharing photos via online social networks. Computers in Human Behavior, 35, 256-267. https://hatlab.org/wp-content/uploads/2017/10/li_hfes2017.pdf

Gales, M. J. F., Watanabe, S., & Fosler-Lussier, E. (2012). Structured discriminative models for speech recognition: An overview. IEEE Signal Processing Magazine, 29(6), 70-8

Gallo, I., Rehman, A. U., Dehkordi, R. H., Landro, N., La Grassa, R., & Boschetti, M. (2023). Deep Object Detection of Crop Weeds: Performance of YOLOv7 on a Real Case Dataset from UAV Images. Remote Sensing, 15(2), 539. DOI: https://doi.org/10.3390/rs15020539

Garg, R., & Raheja, S. (2022). Fuzzy Distance-Based approach for the assessment and selection of programming languages. *International Journal of Decision Support System Technology*, *15*(1), 1–19. DOI: https://doi.org/10.4018/ijdsst.315761

Gayathridevi, K., & Kanmani, S., (2023). A Comparative Study of Various Versions of YOLO Algorithm to Detect Drones. *Recent Research Reviews Journal*, *2*(1), 54-61. doi:10.36548/rrrj.2023.1.05

Geetha, R., & Bhanu, S. R. D. (2018). Recruitment through artificial intelligence: a conceptual study. *International Journal of Mechanical Engineering and Technology*, *9*(7), 63-70.

Gossett, E. (2018). Family Vlogging – Good or bad? – The Research Center on Values in Emerging Science and Technology. Retrieved from: https://rcvest.southernct.edu/family-vlogging-good-or-bad/

Gloe, T., Fischer, A., & Kirchner, M. (2014). Forensic analysis of video file formats. Retrieved from: https://www.researchgate.net/publication/261701780_Forensic_analysis_of_video_file_formats

Handalage, U., & Kuganandamurthy, L. (2021). Real-Time Object Detection Using YOLO: A Review. ResearchGate, 6. https://doi.org/10.13140/RG.2.2.24367.66723%20Authors:

Harrington, P. (2012). Machine learning in action. Simon and Schuster.

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2018). Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1–1. https://doi.org/10.1109/tpami.2018.2844175

Hukkelas, H., Mester, R., & Lindseth, F. (2019). DeepPrivacy: A Generative Adversarial Network for Face Anonymization. Retrieved from https://www.researchgate.net/publication/335737970_DeepPrivacy_A_Generative_Adversarial_Network_for_Face_Anonymization/download?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6Il9kaXJlY3QifX0

Ilia, P., Polakis, I., Athanasopoulos, E., Maggi, F., & Ioannidis, S. (2015). Face/Off: Preventing Privacy Leakage From Photos in Social Networks. Retrieved from https://www.researchgate.net/publication/301419612_FaceOff_Preventing_Privacy_Leakage_From_Photos_in_Social_Networks

Johnson, B. (2019). Introducing Visual Studio Code. Retrieved from https://www.researchgate.net/publication/335606865_Introducing_Visual_Studio_Code

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260

Kadhum, A. N., & Kadhum. (2023). View of Literature Survey on YOLO models for face recognition in COVID-19 pandemic. *Journal of Image Processing and Intelligent Remote Sensing*, *3*(4), 27–35. DOI: https://doi.org/10.55529/jipirs.34.27.35

Kang, K., Li, H., Liu, X., Ouyang, W., Wang, X., Xiao, T., & Yan, J. (2017). Object Detection in Videos with Tubelet Proposal Networks. Core.ac.uk. https://core.ac.uk/works/39082431

Khalili, S., & Shakiba, A. (2022). A face detection method via ensemble of four versions of YOLOs. 2022 International Conference on Machine Vision and Image Processing (MVIP). https://doi.org/10.1109/mvip53647.2022.9738779

Khan, A. A., Laghari, A. A., & Awan, S. A. (2021). Machine learning in computer vision: a review. EAI Endorsed Transactions on Scalable Information Systems, 8(32), e4-e4.

Knuttila, L. (2011). User unknown: 4chan, anonymity and contingency. *First Monday, 16*(10). https://doi.org/10.5210/fm.v16i10.3665

Krishna, K. M., Chowdhari, K., Anusha, J., Sravani, J., & Chowdary, K. (2022). *IJFANS International Journal of Food and Nutritional Sciences*, *12*(1), 1796–1805. DOI: https://doi.org/10.48047/IJFANS/V11/I12/190

Kumar, A., Gupta, S., Rai, A., & Sinha, S. (2013). Privacy Protection in Online Social Networks. *International Journal of Advanced Computer Research*, *3*(2), 251-256. https://hatlab.org/wp-content/uploads/2017/10/li_hfes2017.pdf

Lateef, Z. (2023). Artificial Intelligence With Python: A Comprehensive Guide. Edureka! Retrieved from: https://www.edureka.co/blog/artificial-intelligence-with-python/

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2018). Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1–1. https://doi.org/10.1109/tpami.2018.2858826

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. (2016). SSD: Single Shot MultiBox Detector. https://arxiv.org/pdf/1512.02325.pdf

Liu, Y., Wang, H., Liu, Y., Luo, Y., Li, H., Chen, H., Liao, K., & Li, L. (2023). A Trunk Detection Method for Camellia oleifera Fruit Harvesting Robot Based on Improved YOLOv7. Forests, 14(7), 1453. DOI: https://doi.org/10.3390/f14071453

Lu, Y., & Young, S. (2020). A survey of public datasets for computer vision tasks in precision agriculture. Computers and Electronics in Agriculture, 178, 105760.

Kränzler, M., Herglotz, C., & Kaup, A. (2019). "Extending Video Decoding Energy Models for 360° and HDR Video Formats in HEVC," 2019 Picture Coding Symposium (PCS), Ningbo, China, pp. 1-5, DOI: 10.1109/PCS48520.2019.8954563.

Ma, T., Li, D., Wang, W., & Dong, J. (2021). Face Anonymization by Manipulating Decoupled Identity Representation.

Ma, X. (2015). Anonymity, intimacy and self-disclosure in social media. AEA Randomized Controlled Trials. https://doi.org/10.1257/rct.840-3.0

Nottingham, E. (2019). 'Dad! Cut that Part Out!' Children's Rights to Privacy in the Age of 'Generation Tagged': Sharenting, Digital Kidnapping and the Child Micro-Celebrity'. SSRN Electronic Journal. Retrieved from: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3602712

OpenCV. (2020, November 4). About - OpenCV. Retrieved from: https://opencv.org/about/

Pedamkar, P. (2023, July 10). Python Tkinter. EDUCBA. Retrieved from: https://www.educba.com/python-tkinter/

Peña, K. D. (2023, February 3). PH social media craze: 77% of Filipinos more engaging online than in real life | Inquirer News. INQUIRER.net. Retrieved from: https://newsinfo.inquirer.net/1724608/ph-social-media-craze-77-of-filipinos-more-engaging-online-than-in-real-life

Plainer, M. (2020). Study of Visual Studio Code: Practical Course — Contributing to an Open-Source Project. Retrieved from https://www21.in.tum.de/teaching/osp/WS20/assets/pr-plainer-vscode.pdf

Puyda, V., & Stoian, A. (2017). ON METHODS OF OBJECT DETECTION IN VIDEO STREAMS. Retrieved from https://www.researchgate.net/publication/347996502_ON_METHODS_OF_OBJECT_DETECTION_IN_VIDEO_STREAMS

Python. (n.d.). What is Python? Executive Summary. Python.org. Retrieved from: https://www.python.org/doc/essays/blurb/

Qi, D., Tan, W., Yao, Q., & Liu, J. (2023). YOLO5Face: Why reinventing A face detector. Lecture Notes in Computer Science, 228–244. https://doi.org/10.1007/978-3-031-25072-9_15

Rahaman, Md. F. (2023, August). The Current Trends of Object Detection Algorithms: A Review. ResearchGate. https://www.researchgate.net/publication/373392107_The_Current_Trends_of_Object_Detection_Algorithms_A_Review

Rahaman, Md. F. (2023, August). The Current Trends of Object Detection Algorithms: A Review. ResearchGate. https://www.researchgate.net/publication/373392107_The_Current_Trends_of_Object_Detection_Algorithms_A_Review

Ramesh, H. (2021, December 8). Implementing Transfer Learning in PyTorch - Harin Ramesh - Medium. Medium. DOI: https://harinramesh.medium.com/transfer-learning-in-pytorch-f7736598b1ed

Redmon, J., Farhad, A. (2019). YOLOv3: An Incremental Improvement. https://pjreddie.com/media/files/papers/YOLOv3.pdf

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149. https://doi.org/10.1109/tpami.2016.2577031

Rizzica, A. (2021). Sexually explicit deepfakes: to what extent do legal responses protect the depicted persons? [MA Thesis]. Tilburg Law School.

Rosberg, F., Aksoy, E. E., Englund, C., & Alonso-Fernandez, F. (2023). FIVA: Facial Image and Video Anonymization and Anonymization Defense. Retrieved from https://www.researchgate.net/publication/373822963_FIVA_Facial_Image_and_Video_Anonymization_and_Anonymization_Defense

Rouf, M. A., Wu, Q., Yu, X., Iwahori, Y., & Wu, H. (2023). Real-time Vehicle Detection, Tracking and Counting System Based on YOLOv7. Embedded Self Organizing Systems, 10(7). https://doi.org/10.14464/ess.v10i7.598

Rouse, M. (2022). Python. Techopedia. Retrieved from: https://www.techopedia.com/definition/3533/python

Santos, C., De Aguiar, M. S., Welfer, D., & Belloni, B. (2022). A new approach for detecting Fundus lesions using image processing and deep neural network architecture based on YOLO model. Sensors, 22(17), 6441. DOI: https://doi.org/10.3390/s22176441

Schimansky, R. (2024). CustomTkinter. Retrieved from: https://customtkinter.tomschimansky.com/

Shah, D. (2020). YOLOv4 — Version 1: Bag of Freebies, Medium. Retrieved from: https://medium.com/visionwizard/yolov4-bag-of-freebies-dc126623fc2d

Solawetz, J. (2022). What is YOLOv7? A Complete Guide. Roboflow. Retrieved from: https://blog.roboflow.com/yolov7-breakdown/

Sriram, V. P., Lakshmi, K., Podile, V., Naved, M., & Kumar, K. (2021). Role of machine learning and their effect on business management in the world today. *Vidyabharati International Interdisciplinary Research Journal*.

Subhash, V. (2023). Format conversion. In Apress eBooks (pp. 47–69). DOI: https://doi.org/10.1007/978-1-4842-8701-9_5

Szeliski, R. (2022). Computer vision: algorithms and applications. Springer Charm. DOI: https://doi.org/10.1007/978-3-030-34372-9

Tan, M., Pang, R., & Le, Q. (2020). EfficientDet: Scalable and Efficient Object Detection. In arxiv.org (p. 8). https://arxiv.org/pdf/1911.09070.pdf

Thokal, V., Kirtkar, S., Tinwala, A. Q., & Shakil, M. A. M. (2021). A Modern GUI Desktop Application with Advance Features for GST Billing and Management. *International Journal of Computer Science Trends and Technology (IJCST)*, *9*(4), 14–17.

Trytko, K. (2015). Blessing or curse of the digital world — perceptions of online anonymity in Polish daily newspapers. Retrieved from https://wuwr.pl/cejc/article/view/6778/6424

Vicente-Martínez, J. A., Márquez-Olivera, M., García-Aliaga, A., & Hernández-Herrera, V. (2023). Adaptation of YOLOv7 and YOLOv7_tiny for Soccer-Ball Multi-Detection with DeepSORT for Tracking by Semi-Supervised System. Sensors, 23(21), 8693. DOI: https://doi.org/10.3390/s23218693

Wang, C., Bochkovskiy, A., & Liao, H.-Y. M. (2022, July 6). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. DOI: https://doi.org/10.48550/arXiv.2207.02696

Wei, G., Wan, F., Zhou, W., Xu, C., Ye, Z., Liu, W., Lei, G., & Xu, L. (2023). BFD-YOLO: A YOLOV7-Based detection method for building façade defects. Electronics, 12(17), 3612. DOI: https://doi.org/10.3390/electronics12173612

Xiong, Y., Zhu, K., Lin, D., & Tang, X. (2015). Recognize complex events from static images by fusing deep channels. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2015.7298768

Xuan, X., Deng, J., Wang, W., Li, Z., Yao, H., & Wang, Z. (2020). A review of research on object detection based on deep learning. *Journal of Physics: Conference Series, 1*. https://doi.org/10.1088/1742-6596/1684/1/012028

Yang Fu, C. (2020). SINGLE-SHOT OBJECT DETECTORS AND THE FUTURE DIRECTION OF OBJECT DETECTION. 3. https://cdr.lib.unc.edu/concern/dissertations/vd66w548t

Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). Wider face: A face detection benchmark. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2016.596

Zaatouri, K., & Ezzedine, T. (2018). A Self-Adaptive Traffic Light Control System Based on YOLO. 2018 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC). doi:10.1109/iintec.2018.8695293

Zeng, H., & Fang, Y. (2013). Implementation of Video Transcoding Client Based on FFMPEG. Advanced Materials Research, 756-759, 1748–1752. DOI:10.4028/www.scientific.net/amr.756-759.1748 10.4028/www.

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). THIS PAPER HAS BEEN ACCEPTED BY IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS FOR PUBLICATION 1 Object Detection with Deep Learning: A Review (p. 1). https://arxiv.org/pdf/1807.05511.pdf

Appendix A

SAMPLE EVALUATION SHEET
Technological University of the Philippines
College of Science
Computer Studies Department

Name (Optional):_____Occupation:_____Date: _____

Direction: Please encircle the appropriate number of your rating to evaluate the project entitled "SEENSORED: Face Anonymizer Using YOLOv7 Object Detection Algorithm" using the scale below:

4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Face Anonymizer Desktop Application				
Criteria	**Rating**			
A. **Functionality Suitability**				
1. Functional Completeness. SEENSored desktop application covers all the specified tasks and user objectives.	**4**	**3**	**2**	**1**
2. Functional Correctness. SEENSored desktop application provides the correct result with the needed degree of precision.	**4**	**3**	**2**	**1**
3. Functional Appropriateness. SEENSored desktop application facilitates the accomplishment of specified tasks and objects.	**4**	**3**	**2**	**1**
B. **Performance Efficiency**				
1. Time Behavior. SEENSored desktop application's processing times and throughput rates meet the necessary requirements.	**4**	**3**	**2**	**1**
2. Resource Utilization. The types and amounts of resources that the SEENSored desktop application uses to fulfill its requirements.	**4**	**3**	**2**	**1**
3. Capacity. Maximum limits of SEENSored desktop application meet requirements.	**4**	**3**	**2**	**1**
C. **Compatibility**				
1. Co-existence. SEENSored desktop application can perform its required functions efficiently while sharing a common environment with products, without detrimental impact to other products.	**4**	**3**	**2**	**1**
2. Interoperability. SEENSored desktop application can exchange information with other products and mutually use the information that has been exchanged.	**4**	**3**	**2**	**1**
D. **Usability**				
1. Appropriateness Recognizability. SEENSored desktop application is recognizable according to the needs of the user.	**4**	**3**	**2**	**1**
2. Learnability. SEENSored desktop application can be learned to be used by specified users within a specified amount of time.	**4**	**3**	**2**	**1**
3. Operability. SEENSored desktop application has attributes that make it easy to operate and control.	**4**	**3**	**2**	**1**
4. User Error Protection. SEENSored desktop application prevents users against operation errors	**4**	**3**	**2**	**1**
5. User Interface Aesthetics. SEENSored desktop application has pleasing and satisfying interaction for the user.	**4**	**3**	**2**	**1**
6. Self-Descriptivity. SEENSored desktop application presents appropriate information, where needed by the user, to make its capabilities and use immediately obvious to the user without excessive interactions with a	**4**	**3**	**2**	**1**

product or other resources (such as user documentation, help desks or other users).				
E. **Reliability**				
1. Maturity. SEENSored desktop application functions consistently and as expected under normal use conditions.	**4**	**3**	**2**	**1**
2. Availability. SEENSored desktop application's cache files are accessible if a repeated video file is edited.	**4**	**3**	**2**	**1**
3. Fault Tolerance. SEENSored desktop application operates as intended despite the presence of hardware or software faults.	**4**	**3**	**2**	**1**
4. Recoverability. SEENSored desktop application can be restored to a functional state quickly after a failure.	**4**	**3**	**2**	**1**
F. **Security**				
1. Integrity. SEENSored desktop application ensures that the cache files are not tampered.	**4**	**3**	**2**	**1**
G. **Maintainability**				
1. Modularity. SEENSored desktop application is composed of discrete components such that a change to one component has minimal impact on other components.	**4**	**3**	**2**	**1**
2. Reusability. SEENSored desktop application can be used as an asset in more than one system, or in building other assets.	**4**	**3**	**2**	**1**
3. Analyzability. SEENSored desktop application can be evaluated for the effects of an intended change to one or more of its components, for identifying parts that need to be changed, or for diagnosing faults in the application.	**4**	**3**	**2**	**1**
4. Modifiability. SEENSored desktop application can be effectively and efficiently modified without introducing defects or degrading existing product quality.	**4**	**3**	**2**	**1**
5. Testability. SEENSored desktop application is testable as it is easy to create clear and comprehensive tests to assess all the important functionalities.	**4**	**3**	**2**	**1**
H. **Portability**				
Adaptability. SEENSored desktop application can effectively and efficiently be adapted for or transferred to different hardware, software or other operational or usage environments.	**4**	**3**	**2**	**1**
Scalability. SEENSored desktop application can handle growing or shrinking workloads or to adapt its capacity to handle variability.	**4**	**3**	**2**	**1**
Installability. SEENSored desktop application can be successfully installed and/or uninstalled in a specified environment.	**4**	**3**	**2**	**1**

Comments/Suggestions:

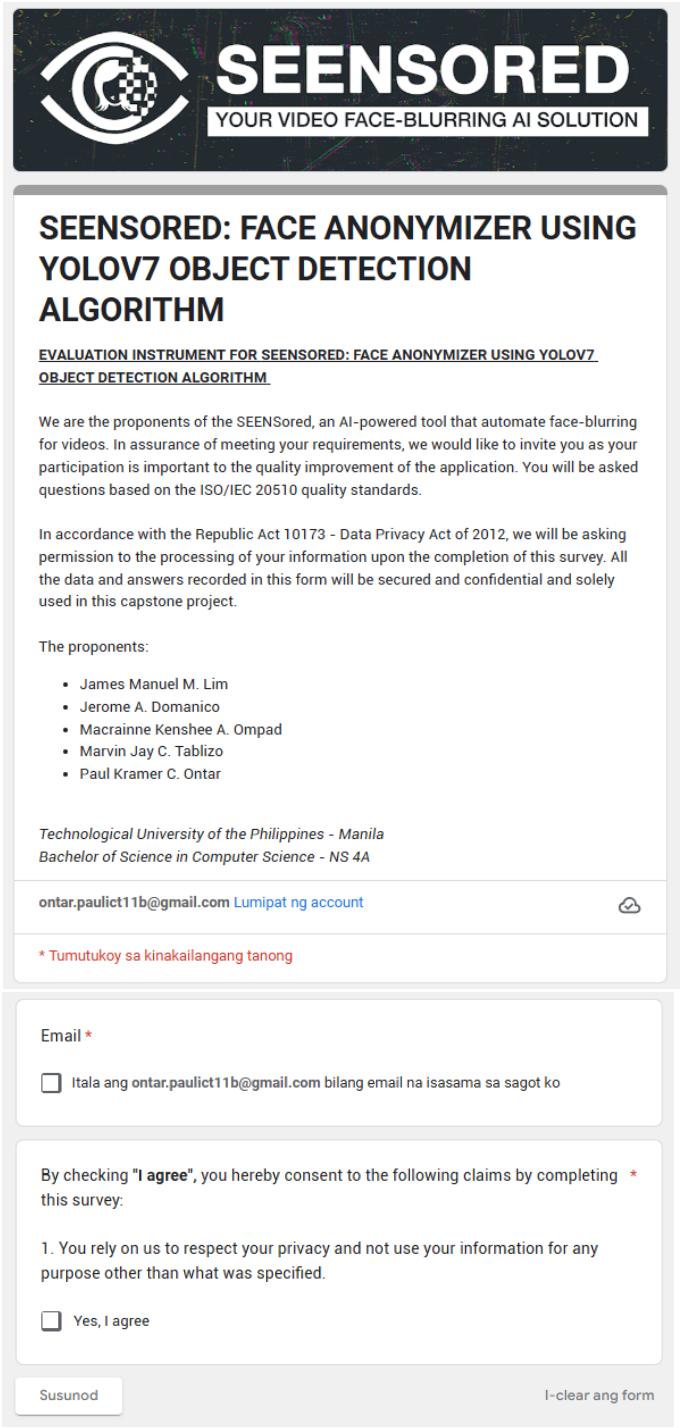
_____

_____

_____

Appendix B

GOOGLE FORMS EVALUATION TOOL



Respondent's Data

Name (Optional):

Iyong sagot

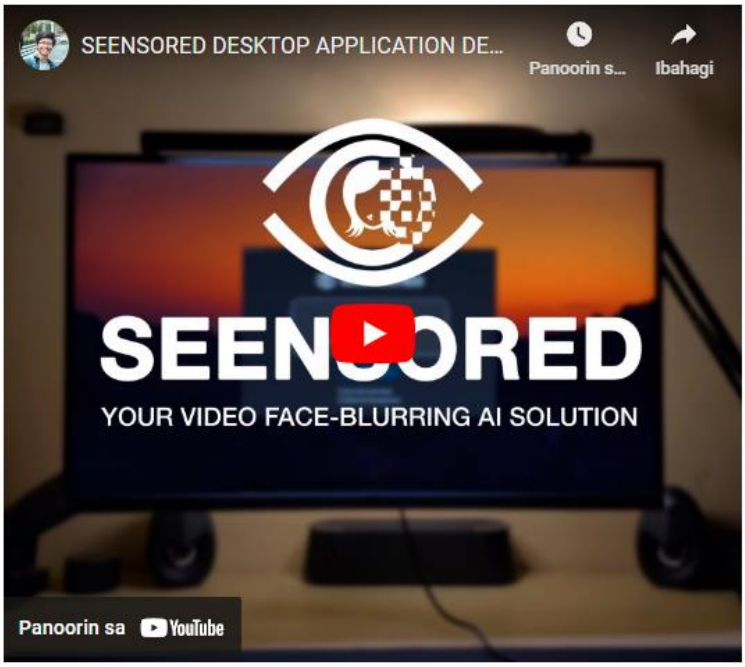
Categories *

- ◯ Content Creator
- ◯ Mass Media
- ◯ Common User
- ◯ IT Professional

SOFTWARE PRODUCT VIDEO DEMONSTRATION OF SEENSORED

We would like to invite you to watch this video for further understanding of our developed desktop application SEENSored.

SEENSORED DESKTOP APPLICATION DEMONSTRATION VIDEO



Bumalik Susunod I-clear ang form

FUNCTIONAL SUITABILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Functional Completeness *

SEENSored desktop application covers all the specified tasks and user objectives.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Functional Correctness *

SEENSored desktop application provides the correct result with the needed degree of precision.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Functional Appropriateness *

SEENSored desktop application facilitates the accomplishment of specified tasks and objects.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

PERFORMANCE EFFICIENCY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Time Behavior *

SEENSored desktop application's processing times and throughput rates meet the necessary requirements.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Resource Utilization *

The types and amounts of resources that the SEENSored desktop application uses to fulfill its requirements.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Capacity *

Maximum limits of SEENSored desktop application meet requirements.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

COMPATIBILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Coexistence *

SEENSored desktop application can perform its required functions efficiently while sharing a common environment with products, without detrimental impact to other products.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Interoperability *

SEENSored desktop application can exchange information with other products and mutually use the information that has been exchanged.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

USABILITY/INTERACTION CAPABILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Appropriateness Recognizability *

SEENSored desktop application is recognizable according to the needs of the user.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Learnability *

SEENSored desktop application can be learned to be used by specified users within a specified amount of time.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Operability *

SEENSored desktop application has attributes that make it easy to operate and control.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

User Error Protection *

SEENSored desktop application prevents users against operation errors

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

User Interface Aesthetics *

SEENSored desktop application has pleasing and satisfying interaction for the user.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Self-Descriptivity *

SEENSored desktop application presents appropriate information, where needed by the user, to make its capabilities and use immediately obvious to the user without excessive interactions with a product or other resources (such as user documentation, help desks or other users).

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

RELIABILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Maturity *

SEENSored desktop application functions consistently and as expected under normal use conditions.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Availability *

SEENSored desktop application's cache files are accessible if a repeated video file is edited.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Fault Tolerance *

SEENSored desktop application operates as intended despite the presence of hardware or software faults.

- ◉ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Recoverability *

SEENSored desktop application can be restored to a functional state quickly after a failure.

- ◉ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

SECURITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Integrity *

SEENSored desktop application ensures that the cache files are not tampered.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

MAINTAINABILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 5 – Excellent 4 – Very Good 3 – Good 2 – Fair 1 – Bad

Modularity *

SEENSored desktop application is composed of discrete components such that a change to one component has minimal impact on other components.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Reusability *

SEENSored desktop application can be used as an asset in more than one system, or in building other assets.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Analyzability *

SEENSored desktop application can be evaluated for the effects of an intended change to one or more of its components, for identifying parts that need to be changed, or for diagnosing faults in the application.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Modifiability *

SEENSored desktop application can be effectively and efficiently modified without introducing defects or degrading existing product quality.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Testability *

SEENSored desktop application is testable as it is easy to create clear and comprehensive tests to assess all the important functionalities.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

PORTABILITY/FLEXIBILITY

Instructions: Please evaluate using the given scale and selecting values corresponding to your response.

Numerical Rating: 4 – Highly Acceptable 3 – Very Acceptable 2 – Acceptable 1 – Not Acceptable

Adaptability *

SEENSored desktop application can effectively and efficiently be adapted for or transferred to different hardware, software or other operational or usage environments.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Scalability *

SEENSored desktop application can handle growing or shrinking workloads or to adapt its capacity to handle variability.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Installability *

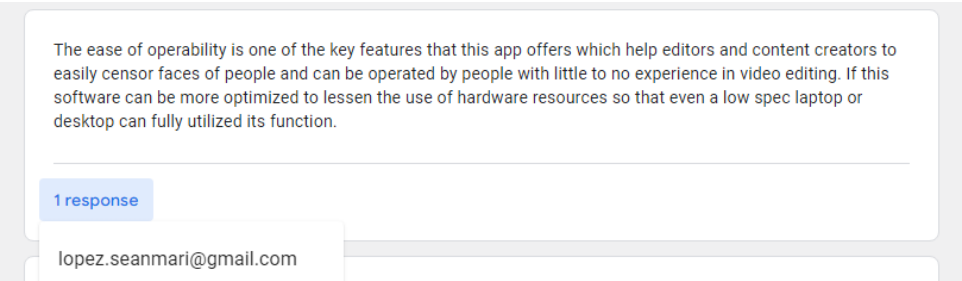
SEENSored desktop application can be successfully installed and/or uninstalled in a specified environment.

- ◯ Highly Acceptable
- ◯ Very Acceptable
- ◯ Acceptable
- ◯ Not Acceptable

Appendix C

EVALUATORS' HIGHLIGHTS

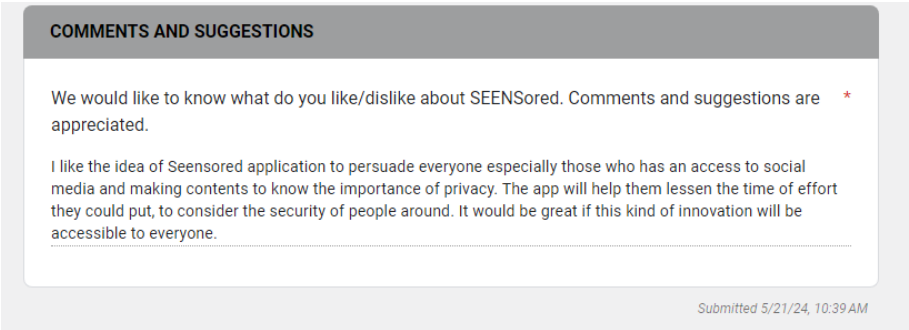
1. **Sean Mari Lopez**
   IT Professional



Answered from Google forms:

"The ease of operability is one of the key features that this app offers which help editors and content creators to easily censor faces of people and can be operated by people with little to no experience in video editing. If this software can be more optimized to lessen the use of hardware resources so that even a low spec laptop or desktop can fully utilized its function."

2. **Kim Gumila**
   IT Professional



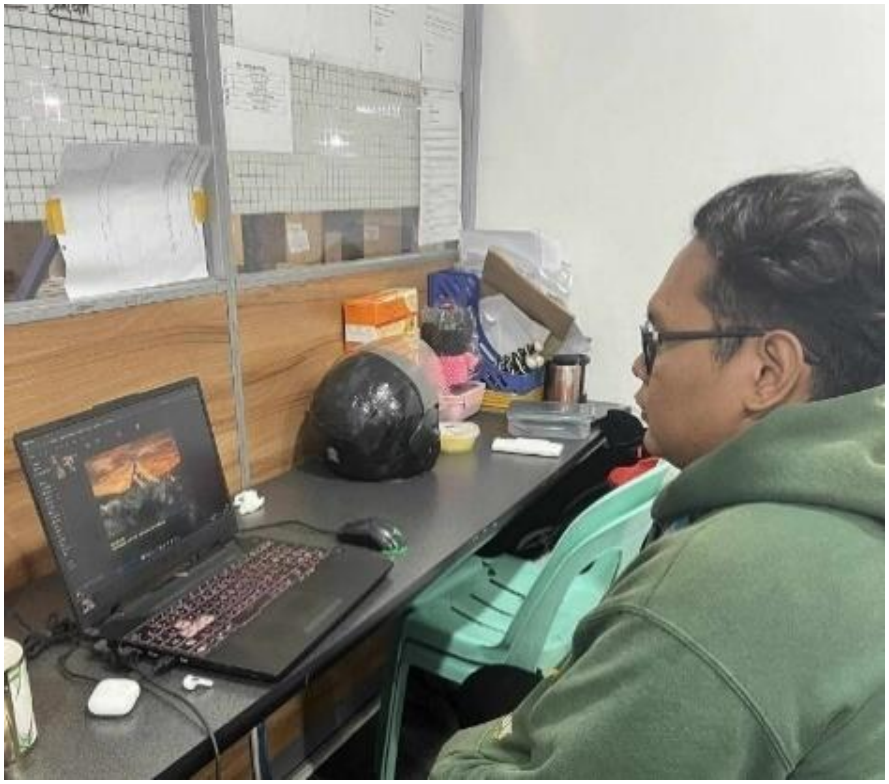
Answered from Google forms:

"I like the idea of Seensored application to persuade everyone especially those who has an access to social media and making contents to know the importance of privacy. The app will help them lessen the time of effort they could

put, to consider the security of people around. It would be great if this kind of innovation will be accessible to everyone."

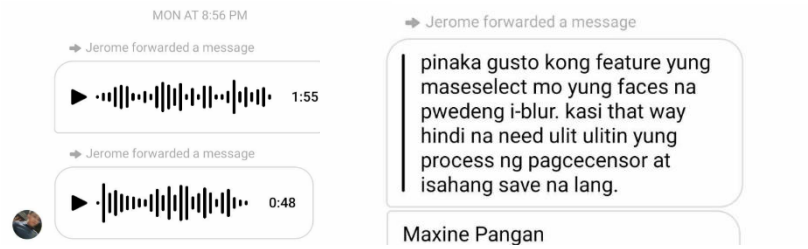
3. **J. Carpio**
   Software Developer at ERT Philippines

s



"First, application should allow to upload/process other video file formats. MP4 alone is not sufficient for media creator type of users. Second, on the analysis screen of which objects are to be masked, application should be able to categorize the likeliness of a match like displaying the confidence level of the scan or placing the low-probability scans to another category on UI so that it is more understandable for the users. Lastly, should have sample low-resolution sample videos for testing."

4. Maxine Jade Pangan
Senior Feature Writer, The Communicator
Former Vice-President for Internal Affairs, PUP Journalism Guild



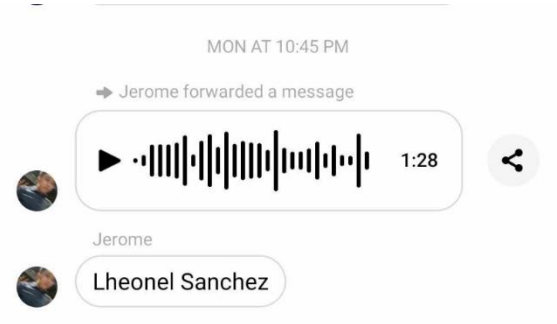
"Bilang isang journalism student and, siyempre nakapag-intern na rin sa mainstream media, napakahalaga sa amin ng censorship sa mga binabalita naming. Dun sa student publication na kinabibilangan ko, meron ding instances na kunwari sa mga video reports; kailangan naming magcensor kasi for example, bata yung nakasama sa video o picture. Siyempre hindi dapat ma-tamper yung privacy nila so kailangan iblur yung mga mukha nila. Nung nakita ko tong Seensored, super convenient niya if ever na ilalabas siya publicly kasi nga to be honest kasi minsan nakakatamad naman talagang magcensor lalo na kung maraming mukha yung kita sa frame and siyempre kailangan di ba lagging mabilis yung paglalabas ng balita para hindi napapanis so minsan nakakatamad din talagang isa-isa pang iedit yung video or photos. Kailangan mabilis at competitive sa paglalabas ng balita. Kung mailalabas publicly tong Seensored, magiging malaking tulong to lalo na sa industry naming kasi super convenient kasi konting clicks lang yung kailangan gawin lalo na sa video."

"Dagdag ko lang din na this would be a vital software if ever man na irerelease kasi mas mapapabilis yung pagcecensor. Mahalaga kasi itong pagcecensor ng mga faces sa ethical reporting ng journalists. Ayun bukod sa mapoprotektahan na yung privacy ng mga individuals na makakasama sa frame, maeensure din na maayos yung pagrereport at ethical. Walang malalabag na ethical procedures sa paglalabas ng balita."

"Pinaka gusto kong feature yung maseselect mo yung faces na pwedeng i-blur. kasi that way hindi na need ulit ulitin yung process ng pagcecensor at isahang save na lang."

5. Lheonel Sanchez

National Vice-Chair for External Affairs, Alyansa ng Kabataang Mamamahayag
Managing Editor, The Communicator



"Actually, maganda siya especially sa mga integrated news, lalo na kapag may minor na nafeature o nakita sa news package. Siguro isang comment lang din, dun sa haba ng pagproprocess ng video, since balita tapos on the spot usually, or yung mga usual na balita na dapat mabibilis na ipresent, siguro dun lang yung need pa iimprove. Based din sa experience ko sa past internship ko, mabilisan talagang record lalo na sa mga clip na kinakailangan iput on-air, pero okay siya. Yung blur sa mga mukha ang bilis nga eh, pati yung pag-identify, nakita agad yung mga mukha individually."

Appendix D

THESIS GRAMMARIAN CERTIFICATION

	TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd., Ermita, Manila, 1000, Philippines Tel No. +632-5301-3001 local 608| Fax No. +632-8521-4063 Email: cos@tup.edu.ph | Website: www.tup.edu.ph	Index No.	REF-COS-3.5-INT-TGC
		Revision No.	00
		Effectivity Date	06132022
VAA-COS	**THESIS GRAMMARIAN CERTIFICATION**	Page	1 / 1

THESIS GRAMMARIAN CERTIFICATION

This is to certify that the thesis entitled,

SEENSORED: FACE ANONYMIZER USING YOLOV7 OBJECT DETECTION ALGORITHM

authored by

Domanico, Jerome A.
Lim, James Manuel M.
Ompad, Macrainne Kenshee A.
Ontar, Paul Kramer C.
Tablizo, Marvin Jay C.

has undergone editing and proofreading by the undersigned.

This Certification is being issued upon the request of Jerome A. Domanico, James Manuel M. Lim, Macrainne Kenshee A. Ompad, Paul Kramer C. Ontar, and Marvin Jay C. Tablizo for whatever purposes it may serve them

Marilyn M. Ignacio
Prof. Marilyn M. Ignacio
Grammarian

Technological University of the Philippines

June 13, 2023
Date of Issuance

Transaction ID	
Signature	

Appendix E

CERTIFICATE OF SIMILARITY INDEX USING TURNITIN FROM URDS

	TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES Ayala Blvd., Ermita, Manila, 1000, Philippines Tel No. +632-5301-3001 local 711 | Fax No. +632-521-4063 Email: urds@tup.edu.ph | Website: www.tup.edu.ph	Index No.	REF-URD-INT-CSI
		Issue No.	01
		Revision No.	01
		Date	04132021
VRE-URD	**CERTIFICATE OF SIMILARITY INDEX USING TURNITIN**	Page	2 / 4
		QAC No.	CC-04132021

This is to certify that the manuscript entitled

"SEENSORED: FACE ANONYMIZER USING YOLOV7 OBJECT DETECTION ALGORITHM"

authored by

James Manuel M. Lim
Jerome A. Domanico
Macrainne Kenshee A. Ompad
Marvin Jay C. Tablizo
Paul Kramer C. Ontar

Has been subjected to similarity check on June 14, 2024 using Turnitin with generated similarity index of 17%.

Processed by:

DOROTHY P. PERNIS
Staff, URDS

Certified correct by:

FRANCISCO D. ESPONILLA II, LPT, Ed.D.
Director, URDS

Transaction ID	
Signature	

Appendix F

CERTIFICATE OF SIMILARITY INDEX USING TURNITIN

Similarity Report

PAPER NAME	AUTHOR
OMPAD_SEENSOREDFACEANONYMIZERUSINGYOLOV7OBJECTDETECTIONALGORITHM.pdf	MACRAINNE KENSHEE OMPAD
WORD COUNT	CHARACTER COUNT
24317 Words	143161 Characters
PAGE COUNT	FILE SIZE
127 Pages	3.0MB
SUBMISSION DATE	REPORT DATE
Jun 14, 2024 11:42 AM GMT+8	Jun 14, 2024 11:44 AM GMT+8

17% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 10% Internet database
- 6% Publications database
- Crossref database
- Crossref Posted Content database
- 14% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 8 words)

Summary

RESEARCHERS' PROFILE

JEROME A. DOMANICO

Bacoor City, Cavite, Philippines
0976-152-4650
jerome.ansusdomanico@gmail.com
linkedin.com/in/jerome-domanico

EDUCATION

Technological University of the Philippines — **Manila, Philippines**
Bachelor of Science in Computer Science — *August 2024*

SHS in San Nicholas III, Bacoor City — **Bacoor City, Cavite**
TVL – ICT — *April 2020*

Bacoor National High School – Main — **Bacoor City, Cavite**
Junior High School — *April 2020*

Talaba Elementary School — **Bacoor City, Cavite**
Primary School — *April 2014*

ACCOMPLISHED PROJECTS

SEENSored: A Face Anonymizer Application
Nov 2023 – May 2024

- Simplified face blurring/hiding in videos with one-click functionality.
- Reduced editing time by up to 80%, enhancing productivity.
- Integrated caching for swift re-anonymization and file integrity checking using hashing.
- Led development and system testing.
  Technologies: Python, CustomTkinter, YOLOv7, Git

Real-Time Headwear Detection with YOLOv5
Dec 2022 – Feb 2022

- Developed a Python-based desktop application for headwear detection, enhancing public security.
- Led development, achieving high accuracy and real-time detection.
- Trained YOLOv5 model with a custom headwear dataset.
- Technologies: Python, Tkinter, YOLOv5

sBOX: IoT Utility Box Powered by ESP8266
Jun 2022 – July 2022

- Created a smart home utility box using ESP8266 and C++.
- Enabled Google Home connectivity via Sinric Pro API.
- Implemented Wake-on-LAN and IR remote control for remote computer activation.

INTERNSHIP EXPERIENCE

Aboitiz Construction, Inc. — **Taguig City, Philippines**
Mobile Application Developer Intern — *Nov 2023 –May 2024*

AppSheet Incident Reporting Application for SHES Department
- Single-handedly developed for a leading construction company

Attendance App for Distributed Teams
- Developed for security, construction, and sales teams

SKILLS

Programming Languages: Dart, Python, C/C++, VB.NET, Java Version Control
Control: Git, Gitlab, Github
Database Management Systems: Cloud Firestore (Firebase) Machine Learning
Learning: Pytorch, Object Detection
Technical Skills: Computer Hardware/Software Troubleshooting, Network Configuration, Remote Desktop Services
Soft Skills: Strong leadership and collaboration abilities, eager to learn and adapt to new technologies
Certifications & Training:

- ISC2 Certified in Cybersecurity (In Progress)
- INE Security Junior Penetration Tester (eJPT) (In Progress)
- CompTIA Security+ (In Progress)
- Cisco Introduction to Cybersecurity
- Cisco Networking Basics
- Computer Systems Servicing NC II

CONFERENCE ATTENDED

HACKIN' KA NA LANG 2024
Lipa City, Batangas *April 27, 2024*
Sophie's Information Technology Services

- Attended seminar covering AI and cybersecurity topics.
- Gained knowledge on responsible AI use and cybersecurity.
- Explored AI innovations and understood cybersecurity attack frameworks.
- Learned about social engineering and cybersecurity vulnerabilities

JAMES MANUEL M. LIM

Malabon City, NCR, Philippines
0928-277-1954
lim.james0039@gmail.com

EDUCATION

Technological University of the Philippines — **Manila, Philippines**
Bachelor of Science in Computer Science — *August 2024*

STI College, Caloocan City — **Caloocan City, NCR**
TVL – ICT — *March 2020*

Malabon National High School — **Malabon City, NCR**
Junior High School — *April 2018*

Malabon Elementary School — **Malabon City, NCR**
Primary School — *March 2014*

ACCOMPLISHED PROJECTS

SEENSored: A Face Anonymizer Application
Nov 2023 – May 2024

- Simplified face blurring/hiding in videos with one-click functionality.
- Reduced editing time by up to 80%, enhancing productivity.
- Integrated caching for swift re-anonymization and file integrity checking using hashing.
- Led development and system testing.
  Technologies: Python, CustomTkinter, YOLOv7, Git

Real-Time Headwear Detection with YOLOv5
Dec 2022 – Feb 2022

- Developed a Python-based desktop application for headwear detection, enhancing public security.
- Led development, achieving high accuracy and real-time detection.
- Trained YOLOv5 model with a custom headwear dataset.
- Technologies: Python, Tkinter, YOLOv5

BOX: IoT Utility Box Powered by ESP8266
Jun 2022 – July 2022

- Created a smart home utility box using ESP8266 and C++.
- Enabled Google Home connectivity via Sinric Pro API.
- Implemented Wake-on-LAN and IR remote control for remote computer activation.

INTERNSHIP EXPERIENCE

Netizen Works Web Design Services — **Quezon City, Philippines**
Web developer intern — *March 2024 –June 2024*
Developed a website for multiple SMDC websites
- Developed for Philippines' fastest-growing real estate developer

SKILLS

Programming Languages: Java, Python, C/C++, VB.NET, C#, HTML, CSS, JavaScript
Control: Git, Github
Machine Learning: Pytorch
Technical Skills: Computer Hardware Troubleshooting,
Desktop Services Soft Skills: Attentive for fixing errors, Collaborative to other workers, Willing to learn gaining new skills
Certifications & Training:
- Introducion to WordPress

CONFERENCE ATTENDED

WDYM: Web Dev, You Mean?
January 11, 2023
- Attended seminar covering web design

MACRAINNE KENSHEE A. OMPAD

General Trias, Cavite, Philippines
0961-116-4359
Macrainneompad@gmail.com
linkedin.com/in/macrainne-kenshee-ompad

EDUCATION

Technological University of the Philippines — **Manila, Philippines**
Bachelor of Science in Computer Science — *August 2024*

Olivarez College — **Parañaque City, Metro Manila**
ABM — *April 2020*

Don Galo National Highschool — **Parañaque City, Metro Manila**
March 2018

Don Galo Elementary School — **Parañaque City, Metro Manila**
April 2014

ACCOMPLISHED PROJECTS

SEENSored: A Face Anonymizer Application
Nov 2023 – May 2024

- Simplified face blurring/hiding in videos with one-click functionality.
- Reduced editing time by up to 80%, enhancing productivity.
- Integrated caching for swift re-anonymization and file integrity checking using hashing.
- Led Documentation
  Technologies: Python, CustomTkinter, YOLOv7, Git

Real-Time Headwear Detection with YOLOv5
Dec 2022 – Feb 2023

- Developed a Python-based desktop application for headwear detection, enhancing public security.
- Led development, achieving high accuracy and real-time detection.
- Trained YOLOv5 model with a custom headwear dataset.
- Technologies: Python, Tkinter, YOLOv5

ROT47 Cipher Encoder and Decoder
Dec 2022 – Feb 2023

- Created an application using vb.net.
- Enabled hiding text's meaning and decoding it.

INTERNSHIP EXPERIENCE

Hyundai Motor Inc. — **Las Piñas, Philippines**
Marketing Data Analytics Intern — *March 2024 –June 2024*
Predictive Customer Segmentation and Targeting

- Collaborated with developers toDevelop a predictive model to segment customers based on their purchasing behavior and identify high-value segments for targeted marketing campaigns.

SKILLS

Programming Languages: Python, C/C++, Java, VB.NET, HTML, CSS, JavaScript
Control: Git, Github
Machine Learning: Pytorch
Technical Skills: Computer Hardware Troubleshooting,
Desktop Services Soft Skills: Attentive for fixing errors, Collaborative to other workers, Willing to learn gaining new skills
Certifications & Training:

- Introducion to WordPress

CONFERENCE ATTENDED

WDYM: Web Dev, You Mean?
January 11, 2023

- Attended seminar covering web design

PAUL KRAMER C. ONTAR

Bacoor City, Cavite, Philippines
0976-152-4650
jerome.ansusdomanico@gmail.com
linkedin.com/in/jerome-domanico

EDUCATION

Technological University of the Philippines — **Manila, Philippines**
Bachelor of Science in Computer Science — *August 2024*

SHS in San Nicholas III, Bacoor City — **Bacoor City, Cavite**
TVL – ICT — *April 2020*

Bacoor National High School – Main — **Bacoor City, Cavite**
Junior High School — *April 2020*

Molino Elementary School — **Bacoor City, Cavite**
Primary School — *April 2014*

ACCOMPLISHED PROJECTS

SEENSored: A Face Anonymizer Application
Nov 2023 – May 2024

- Simplified face blurring/hiding in videos with one-click functionality.
- Reduced editing time by up to 80%, enhancing productivity.
- Integrated caching for swift re-anonymization and file integrity checking using hashing.
- Led development and system testing.
  Technologies: Python, CustomTkinter, YOLOv7, Git

AI-Driven File Organization Using Particle Swarm Optimization Algorithm
Jun 2023 – Jul 2023

- Developed in Python.
- Utilized PSO algorithm in a file moving app.
- Enhanced file transfer efficiency using Artificial Intelligence.
- Project recognized for minimizing manual tasks.
- Collaborated to integrate user-friendly features.

Handwritten Character Recognition App
Dec 2022 – Feb 2023

- Developed in Python and Dart (UI).
- Led the creation of a handwritten character detection app.
- Trained CNN model on a custom handwritten character dataset.
- Achieved high accuracy in character detection.
- Deployed as a mobile application

INTERNSHIP EXPERIENCE

Gencys Digital Trading, Inc. — **Bacoor City, Cavite**
Quality Assurance Analyst — *Mar 2024 – Jun 2024*

Ecommuniversity Website
- Website developed with collaborative e-commerce companies that offers courses for the people

LHIKE ERP
- Web application developed that offers tools for e-commerce businesses

SKILLS

Programming Languages: Python, Java, C, VB.NET, SQL, HTML5, CSS3, JavaScript
Technical Skills: UI and UX, Computer Hardware/Software Troubleshooting, Network Configuration
Multimedia Skills: Adobe Photoshop, Adobe Premiere Pro
Certifications & Training:
- Computer Systems Servicing NC II
- Project League of Developers Initiative (Internship)

CONFERENCE ATTENDED

Quantum Computing: Are You Ready for the Future of Computing?
Manila, Philippines *April 27, 2024*

- Attended seminar covering quantum computing topics.
- Gained knowledge on quantum computing.
- Explored potentials of quantum computing in the world of programming.

MARVIN JAY C. TABLIZO

TRAMO, PASAY CITY, PHILIPPINES
0976-573-7903
tablizovinn@gmail.com
linkedin.com/in/marvin-jay-tablizo

EDUCATION

Technological University of the Philippines — **Manila, Philippines**
Bachelor of Science in Computer Science — *August 2024*

Arellano University – Jose Abad Santos Campus — **Pasay, Philippines**
TVL – ICT — *April 2020*

Pasay City East High School — **Pasay, Philippines**
Junior High School — *April 2018*

Apelo Cruz Elementary School — **Pasay, Philippines**
Primary School — *April 2014*

ACCOMPLISHED PROJECTS

SEENSored: A Face Anonymizer Application
Nov 2023 – May 2024

- Developed the initial idea and conceptualized the application's purpose and features.
- Conducted extensive research on the necessary data and technologies required for the development of the application.
- Reduced editing time by up to 80%, enhancing productivity.

Technologies: Python, CustomTkinter, YOLOv7, Git

Pokemon Recognition using CNN
Dec 2022 – Feb 2022

- Developed a C#-based desktop application for Pokémon recognition, enhancing user engagement and interaction.
- Led development, achieving high accuracy and real-time detection.
- Trained CNN model with a comprehensive Pokémon image dataset.
- Technologies: C#, Microsoft Visual Studio, CNN

sBOX: IoT Utility Box Powered by ESP8266
Jun 2022 – July 2022

- Responsible for wiring and hardware design.
- Implemented IoT functionality for seamless integration with smart home systems.

INTERNSHIP EXPERIENCE

ETR Total Business Sol'n Provider — **Paranaque, Philippines**
Research and Development (AI) , Technical Support, and Data Engineer Intern — *February –June 2024*

- **AWS Textract Application**
- **Technical Support**
- **Database Backup Automation**
- **Data Processing**

SKILLS

Programming Languages: Python, VB.NET, C# , Github
Machine **Learning:** Pytorch, Object Detection
Technical Skills: Computer Hardware/Software Troubleshooting, Network Configuration, Remote
Desktop Services Soft Skills: Strong leadership and collaboration abilities, eager to learn and adapt to new technologies