Science Fair Journal

Today we are going to start our project by researching some background information. The first thing we decided to research was,

What is Artificial Intelligence (AI)?

01/14/23

Many sources, such as Google, Amazon, and IBM, describe Artificial Intelligence as a technology that enables computers to solve problems associated with human intelligence. For example, artificial intelligence allows technologies to complete many human tasks, such as comprehending written or spoken language, analyzing datasets, and pattern recognition. In addition, AI allows computers to simulate human intelligence and imitate many things we can do.

The most primitive definition of Artificial Intelligence dates back to 1950 in Alan Turing's Computing Machinery and Intelligence. In this paper, he proposes the question, "Can Machines think?". He proposed a test to determine if computers could think and respond similarly to humans. The test would work by giving a prompt of a field of study with a specified time limit and pretext to both a human and a computer. Then, a secondary human would be given one of these responses randomly and asked to determine whether the response was written by the computer or by the human. This process would be repeated many times to get an accurate result. If the computer could fool the tester in more than half of the tests, it would be proven that the computer is artificially intelligent.

Stuart Russell and Peter Norvig described two approaches to achieving artificial intelligence in computers. One was the human approach, which would try and create technologies that thought and acted like humans. The ideal approach would be systems that thought and acted rationally.

Another topic that we decided today would be essential to research was the Applications of Artificial Intelligence.
Autonomous vehicles are one of the most well-known fields in which artificial intelligence is used. Artificial intelligence is used by self-driving automobiles to steer clear of hazards and make judgments. They perceive their surroundings using a combination of sensors, cameras, and lidar, and deep learning algorithms interpret the information. As a result, there is less need for human assistance while the car is driving itself. As a result, autonomous vehicles have the potential to expand mobility for those who are unable to drive, improve safety, and lessen traffic congestion.

The use of AI in healthcare is another significant application. AI can tailor treatment plans, forecast patient outcomes, and identify diseases. For instance, AI-driven diagnostic systems can examine medical photos to spot cancer or other disease symptoms. In addition, predictive analytics enabled by AI can assist doctors in identifying patients who are at risk of developing specific illnesses, allowing them to take early action. AI can also tailor treatment regimens to a patient's particular traits, such as their genetic makeup or medical background. This can lower healthcare costs while improving patient outcomes.

AI is also used to boost enterprises' and organizations' operational effectiveness. For example, finance professionals can utilize AI-powered algorithms to assess market trends and forecast future stock prices. AI can be applied in retail to improve pricing and inventory control. AI can be used in manufacturing to forecast equipment breakdowns and improve production schedules. These AI-based solutions can boost productivity, lower costs, and enhance client relations.

Artificial intelligence is also utilized in natural language processing (NLP). Human language may be generated and understood by AI. Personal assistants like Siri, Alexa, Google Assistant, chatbots, and virtual assistants used in customer care all use this technology. Additionally, text synthesis, text summarization, and machine translation all make use of AI-powered language models. In addition to helping consumers save time and effort, this technology can enable people in business, healthcare, and education to overcome language difficulties.

AI is also being utilized to raise educational standards. AI-powered tutors can give pupils individualized feedback, enhancing their learning. AI-driven tools can also be utilized to develop more dynamic and engrossing educational content. AI can also assess student performance data and spot problem areas, enabling teachers to offer more precise support.

Finally, AI has the potential to raise people's quality of life as a whole. For instance, assistive technology driven by AI can assist people with disabilities in carrying out routine chores. AI may also create smart houses that automatically change the lighting, temperature, and other settings according to the occupants' preferences. In addition, AI can create smart cities, enhance public transit, use less energy, and can increase livability.

I did some research here because we may need these **data sets** to test for different scenarios involving different skin tones, lighting, and other discrepancies.

The data sets of people that could be utilized to test the Ai

- https://www.pexels.com/search/people/
- https://www.shutterstock.com/search/multiple-ethnicities
- https://www.dreamstime.com/photos-images/black-people.html
- https://www.dreamstime.com/photos-images/indian-people.html
- https://www.dreamstime.com/photos-images/multi-racial-people.html
- https://www.dreamstime.com/photos-images/white-people.html
- https://www.dreamstime.com/photos-images/asian-people.html
- https://www.dreamstime.com/photos-images/people-hoodie.html
- https://unsplash.com/s/photos/people-light-in-dark

Another topic we thought would be essential to research was,

What is Machine Learning (ML)? 01/15/23

Machine learning is a branch of Artificial Intelligence that uses data science and algorithm thinking to create AI that gradually improves over time based on feedback. In general, Machine Learning is a set of algorithms and models that can teach AI how to deal with large sets of data in more efficient and accurate ways.

Since our project was on types of Ai detection, we decided to research all the types of important detection, and the first one we researched was Person detection.

- Person detection
  - A system used to detect a human using a computer/Ai's vision
- Why is person detection proper?
  - Person detection is beneficial because of the many ways it can be utilized. It can be used to detect the number of people entering a stadium. In order to keep a tally, it can be used to detect if everyone is following the rules of the street in a major city. Many other significant applications of person detection include weapon detection, work safety, gender identification, and person recognition. Those applications can be used in incredibly beneficial ways that would help improve the world.
- Using Ai in those applications will create a much-improved society because there will be far fewer errors than in specific processes today, making jobs much more accessible. Having the ability to do a job with an Ai instead of a human will create fewer errors and more job openings. At first glance, people might question that, but taking away a job someone works, does not mean there will be fewer jobs. It would create more openings because
- GitHub - pushprajkatiyar/person_detection_from_cctv_video: detect the number of people every second entering the building gate. #person-detection

Augustus Rollin & Marrio Jeyakumar

- Pedestrian recognition - [GitHub - vinay0410/Pedestrian_Detection: Detects Pedestrians in images using HOG as a feature extractor and SVM for classification]()
- Crowd counting

The third stage of Ai detection we will use is Clothing detection.

- Clothing detection is the second step in Ai detection to help enhance the product, and that is because clothing is something that you would look into after detecting a person. It adds more layers to tests that can be run, so if there is a particular color or texture of the uniform, it can detect those people to ensure they are doing their job and could be used in other ways.

- [Teaching AI to Identify Clothes]()
- The overall shape and texture
- Clothes Catalog
- Does camouflage cause an issue in detection?

The fourth stage of Ai detection we will use is Logo detection.

- Logo detection is similar to clothing detection but is even more specific; it detects the logo of someone's uniform. It can be used with clothing and person detection to create a secure system to ensure no one can bypass the Ai. Instead of someone wearing the same colored clothing that could cause an error, it would detect the logo, severely decreasing the chance of bypassing the Ai.

Today we researched background information that may help us in the future on our project.

The first thing we decided to research was Reinforcement Learning (RL).

Reinforcement learning is a method for using machine learning to train AI models. It is comparable to how people train their canine companions. The procedure entails giving the AI model a task and rewarding it when it completes the assignment successfully. The AI model then evaluates the challenge and its performance to enhance its comprehension of subsequent problems of a similar nature. Thousands of simulations of this process are run, allowing the model to pick up new information quickly. Reinforcement learning uses computers' processing ability to compensate for their lack of intrinsic learning abilities, which is one of its key advantages.

While SARSA is similar to Q-Learning, it also considers the agent's future move. The agent considers the expected reward for the following state-action combination and the maximum expected reward for the upcoming state. This strategy is helpful when the agent needs to carry out several tasks to accomplish a goal. SARSA aims to find a policy that will provide the most significant rewards over time while considering the following action.

Another approach without a Q-table is Monte Carlo. The predicted cumulative reward for various policies is instead estimated using a method known as Monte Carlo sampling. To calculate the predicted cumulative reward for various policies, the agent interacts with the environment and gathers a series of episodes or experiences. Monte Carlo aims to find a strategy to produce the most significant rewards over time.

As an extension of Q-Learning, TD(0) and TD() update the Q-table after each step rather than after each episode. This helps the issue of Q-learning with long-term dependencies. Another method that directly optimizes the policy's parameters rather than the Q-values is the policy gradient method. It makes direct changes to the policy's parameters to make it better.

Overall, RL offers various solutions, and picking the best one for a particular issue is crucial. However, before choosing a method, it is vital to understand the problem and the features of the data because every method has advantages and disadvantages of its own.

The next thing we decided to research was

Supervised Learning. 01/18/23

A less random and more methodical approach to machine learning is supervised learning. Unlike reinforcement learning, supervised learning occurs less randomly and relies on the programmer/tester adjusting factors throughout the learning process. It is typically used with large datasets to train algorithms to make discretions. A training dataset trains models to complete a specific task in supervised learning. The dataset contains input data and the appropriate outputs, which enables the model to develop over time. A loss function gauges the algorithm's accuracy, repeatedly changing until the error is minimized.

Classification and regression are the two categories of supervised learning issues in data mining. To classify test data into distinct categories, classification employs an algorithm. It attempts to determine how the data should be categorized or described by identifying patterns within the dataset. Decision trees, k-nearest neighbors, support vector machines, and naive Bayes are examples of common classification techniques. To comprehend the relationship between dependent and independent variables, regression is used. Predictions are frequently made using it, such as predicting sales for a company. Logistic, polynomial, and linear regression are all common regression algorithms.

The final thing we decided to research for the day was

Unsupervised Learning. 01/22/23

Unsupervised learning is machine learning in which the model is left to discover patterns and structures independently without labeled input. It can be used to solve a wide range of issues, including data visualization, anomaly detection, and clustering. In addition, it is used to find hidden patterns or groupings in the data.

Clustering is a popular method for unsupervised learning. Based on their properties, clustering algorithms put related data points together. For instance, a clustering algorithm could classify clients according to their purchasing patterns, demographics, or other traits. Then, you may use this data for purposes like targeted advertising or consumer segmentation.

Dimensionality reduction is an additional unsupervised learning method. Reducing the number of features in a dataset makes the data easier to display and comprehend. For instance, a dataset with thousands of features could be condensed to a small number while preserving the most crucial data. Popular dimensionality reduction methods include principal component analysis (PCA) and t-SNE (t-Distributed Stochastic Neighbor Embedding).

Anomaly detection, used to spot data points that deviate from the norm, is another type of unsupervised learning. This method is applied to manufacturing quality control, network intrusion, and fraud detection.

Unsupervised learning can also be applied in natural language processing (NLP), extracting characteristics from text data or identifying topics in a collection of articles. Unsupervised learning applies to various issues and can produce insights that would be challenging or impossible to find using manual power. For example, tasks that would take humans much time to complete can be completed instantly using a model trained by unsupervised learning due to the computational ability of computers.

One of the crucial pieces of background information we believed was necessary to continue with the project was:

Classification, Object Detection, and Segmentation. 1/28

Classification is dividing data into several groups according to its attributes or characteristics. Classification is essential in determining if an object or person is present in an image or video when using an object/person detection model. Object/person detection models use deep learning techniques to assess and categorize visual input. Using a convolutional neural network, the first step in this procedure is to extract features from the picture or video (CNN). These attributes are then fed into a classification model, such as a decision tree or a support vector machine (SVM), which uses them to predict whether an object or person is present in the image.

Ensuring that the model is trained on a broad and representative dataset is one of the issues in classification. This entails training the model for object/person detection on a wide range of photos and videos, including various things and people in various lighting situations and perspectives. This makes it more likely that the model will be reliable and able to categorize things and people in various real-world situations correctly. The delicate balance between accuracy and speed in classification is another crucial factor. Fast image and video classification are essential for object/person identification models, especially in real-time applications like security or self-driving automobiles. They must, though, be precise and limit false positives and false negatives. The model parameters and architecture must be carefully tuned to achieve this balance. This topic is essential to start our work as this is considered the first step, and it is crucial that we understand it and how to proceed from here.

Segmenting an image into distinct segments or regions based on attributes like color, texture, or intensity is crucial in computer vision. As a student interested in computer vision and object/person detection models, I have been researching segmentation. The pre-processing phase of segmentation, which divides the image into various sections before recognizing and categorizing objects or persons in each segment, is a critical component of object/person detection models. For instance, in a model for pedestrian detection, the image may initially be divided into areas based on color, texture, and other attributes likely to contain pedestrians. This pre-processing phase aids in minimizing the detection model's search space, making it more efficient and accurate.

Thresholding, clustering, and edge detection are a few of the segmentation techniques that can be applied. Setting a threshold value for a specific characteristic, such as color intensity, and segmenting the image based on pixels that fulfill the threshold criterion are both steps in the thresholding process. Edge detection involves locating the edges of objects in a picture and segmenting the image based on these edges, whereas clustering involves grouping similar pixels based on their properties. Finding the ideal balance between over- and under-segmentation is one of the challenges of segmentation. When an image is over-segmented, it becomes easier to distinguish between the items or persons in each segment. Conversely, under-segmentation happens when an image is separated into too few segments, resulting in incomplete or inaccurate object/person detection.

One important topic we came across while doing research on datasets and classes were

Confusion Matrices. 02/01/23

It was an essential topic to research and discussion as it is a helpful way to see the basics of a dataset. A confusion matrix is a table/matrix of all the outcomes of a given prediction and the classifications to help visualize the outcomes. The grid will typically be a square as each class on the x-axis will also be on the y-axis, so the grid representing the same class for both x and y will be on the square's diagonal. The most simple 2x2 confusion matrix consists of positive and false on each axis. This results in 4 possible outcomes for each situation that all have their definitions. Upbeat means a true positive, which is ideal. It means that the test you ran was supposed to return positive and the model also gave you a positive. Then there are positives and negatives, which results in a false negative. This means that the model was supposed to give you True but gave you falsely instead. There is also negative, positive, a false positive, meaning the model was supposed to give you falsely, but it gave you true. The last possibility is negative, which results when the model was supposed to return false, and that is what it returned. This simple 2x2 confusion matrix allows an easy way to see the strengths and weaknesses of a model and what needs to be optimized.

Although the matrix might give a baseline for assessing the model, it can be hard to tie this to a metric. We found that accuracy and precision were two ways to assess the model. They are given by (TP + TN) / (TP + TN + FP + FN) and TP/(TP+FP), respectively. These formulas turn meaningless numbers of tests on a model into quantifiable numbers that have meaning and purpose.

Although having a 2x2 matrix might be helpful to see a matrix's weakness visually, our research found that many of our potential datasets used more complex confusion matrices that were much larger. This is because the axes were expanded and filled with classes instead of positive and negative. In our case, the classes of the dataset are all the objects we train our model to identify. Although it is larger and there are different meanings, the core of the confusion matrix remains the same. The diagonal represents the true positives, and the others represent false positives or false negatives. In an ideal model, the largest numbers would be on the diagonal. This would mean that the model correctly recognizes the objects.

Today we looked at some object detection models we could use as a base for our project. First, we needed to find the best model, as the rest of the detection pipeline is based on this model. We plan to have person and clothing detection working off this model after we fine-tune it to our needs and train it with our data sets.

We decided to use the YOLOv8 model which is state of the art and excels in all the areas we need it to. It is better than the rest of the competition in many areas, and it is open source, which means that all of the code we need is publicly available on GitHub. YOLO stands for "You Only Look Once" and has had many models over the years. It has been proven reliable and had third-party organizations release other versions of the models.

There are many reasons that we chose YOLOv8 over the other models, but the main points were its speed, detection accuracy, and fantastic ability to generalize. YOLO does not use any complex pipelines which makes it incredibly fast. On a scale of fps, it is less than double its closest competitor at an fps of 91. It additionally has two times the mean Average Precision (mAP) compared to other systems that run in real-time. mAP is a metric used to determine the strengths of object detection models such as YOLO. Those two factors alone made YOLO far ahead of the competition, but other valuable features made it appealing for our project. *Generalization* is a term used in Ai detection models to refer to the adaptability of a model. For example, let us say I trained a model to detect oranges and used a dataset of regular oranges. If my model were good at generalization, it would succeed in determining oranges that were not purebred or of the type I trained it with, such as clementines. This component is crucial to our project as there are always many clothes in different cultures, and we aim to account for as many of them as possible to create a reliable model. Good generalization is the ability of our model to adapt to new data that it was not trained from and still make a confident guess on what the object is.

Although object detection might seem complicated and confusing, YOLO's detection model can be broken down and explained in a few simple steps. Object detection models work by taking a video and separating it into individual frames, or in other words, images. This process is followed whether the video is pre-recorded or live. They are separating the video into individuals. Consecutive frames make it easier to track motion by looking at the difference between pixels after each image. YOLO starts by taking the first frame and creating residual boxes. Next, the image is set up into an NxN grid of squares or cells where each cell is used to localize and predict the object or class within its cell.

YOLO then uses bounding box regression, meaning it will try to create a more well-defined rectangle around the object. It does this by first looking at all the squares and seeing where the probability of object detection is greater than 0. It then weights each box based on the probability chance and uses this formula to create a more well-defined rectangle. For example, y = [pc, bx, by, bh, bw, c1, c2] After getting a bounding box around the object, it refines it by scanning sections of the bounding box to see where an object is detected and adjusting the dimensions as needed. YOLO has already been used for various applications, from agriculture to self-driving cars. It is considered by many in the industry as a state-of-the-art model that is far better than its competitors. It has significantly improved and evolved since 2015 and is predicted to stay leading the field for years. Due to our project's extensive capabilities and good fit, we have chosen to train YOLOv8 as our object detection model.

After we decided on our object detection model, we had to decide on a dataset that we could use to train our model. So we decided it would be useful to research on

02/07/23

A dataset is a group of data that has been structured and presented orderly. A dataset usually contains data relevant to a single subject or area of interest and can be used to gain knowledge, predict the future, or spot patterns and trends. When we discuss datasets, we often describe them as made up of distinct data points or observations. Depending on the data collected, they can take many different forms. Each data point might be a person's height and weight measurements, for instance, if we were gathering information on people's heights and weights. A wide range of sources, including surveys, trials, and open data sources, can be used to produce datasets. In addition, datasets can occasionally be created through data scraping, which gathers information from websites and other internet sources.

The ability to use statistical and machine learning methods on the data to derive insights or make predictions is one of the major advantages of datasets. For instance, we can analyze a dataset of consumer purchase data to spot trends in their behavior, or we might analyze a dataset of medical records to identify the individuals who are most likely to contract a specific disease. Of course, datasets must be carefully curated and prepared to be helpful. This could entail eliminating unnecessary or erroneous data points, normalizing the data to ensure consistency between measurements, or converting category data into numerical values that machine learning algorithms can analyze. With all this being said, we realized that choosing the correct data set was essential to the rest of the project.

I chose the DeepFashion dataset to train my object detection model using YOLOv8. The DeepFashion dataset is a large-scale clothing dataset that contains over 800,000 images and annotations of clothing items in various styles and poses. This dataset is ideal for training an object detection model like YOLOv8 because it provides a diverse set of clothing items that can be used to accurately train the model to identify different types of clothing. Researchers at the Chinese University of Hong Kong created the DeepFashion dataset, a sizable collection of apparel data. It is one of the largest and most complete clothing datasets online, with over 800,000 photos and annotations of clothing items in different styles and positions. The dataset is divided into several categories: apparel, accessories, footwear, and luggage. The collection contains extensive annotations for each piece of clothing under each category, including information on the item's size and placement within the image, its category and qualities, and the locations and postures of any models who may be wearing it.

Developing an annotation system that could manage the intricacy of clothing items was a significant issue when producing the DeepFashion dataset. Garment articles frequently feature complex patterns, textures, and seams, making recognizing and annotating them challenging. The researchers created a unique annotation system that blended manual and machine annotation methods to address this difficulty. They could swiftly and accurately annotate many photographs using this approach. Computer vision models that recognize and categorize clothing items are frequently trained using the DeepFashion dataset. The dataset is particularly useful for training object identification models, like YOLOv8, that can precisely locate and identify clothing articles within an image/frame.

Convolutional neural networks (CNNs), a deep learning method, are used by object detection models like YOLOv8 to analyze and classify photos. Given its capacity to develop hierarchical representations of images that capture ever more complicated information, CNNs are a particular neural network class well-suited to image analysis applications. For example, the photos and annotations are often fed into the network in batches while training an object detection model using the DeepFashion dataset. After extracting information from the photos using a sequence of convolutional layers, the model employs fully connected layers to identify and locate the clothing articles therein. Backpropagation is a training method for the model that modifies the network's weights to reduce the gap between the predicted and actual results.

Today we started and finished training our dataset, but that came with a few issues. The main issue is that we realized that if we needed an accurate model we would need to train YOLOv8 with hundreds of images. Without the proper GPU, this process could take weeks, and we would be left with a smaller pool of images to use. We decided to use Google's cloud platforms to subscribe to our project's NVIDIA A100 on the cloud. This was a significant obstacle; if we did not find this solution, our project might have never moved forward, so we were glad we could solve it. The other problem came after we trained the model. Although we trained it with many images, we needed to account for what was in some of them. Our dataset was more skewed to certain articles of clothing making it more unbalanced. This meant that the model might detect tops on sweaters easier than its ability to detect pants.

02/10/23

Today I searched for various images and put them into our GitHub. Unfortunately, these images had various resolution qualities, points of view, skin colors, and many other things that may impede the Ai detection process of whether they were humans.

Why I picked each image:

Amsa Uniform - I picked this image because our project will eventually require the detection of logos, and starting with our school is a good start. It will also be used for clothing detection, face detection, and person detection.

Work Uniform: This image has a better resolution which is why this was picked along with the Amsa uniform, to test if the difference in quality will have different results. It can use person detection, clothing detection, face detection, and logo detection.

Variety of Uniforms - I picked this image along with the other two uniform pictures because of the variety of clothing. I wanted to see how the Ai would react to these different outfits. It would use person, clothing, and face detection.

Image from a sky view: This image is important to test because you can see glimpses of people's faces and clothing, but it may be hard to see. This image would use clothing, face, and person detection.

Image from the side - I selected this image because of the angle at which the picture was taken and the skin color of this person. Trying to see if the Ai could have some bias from the skin color, or not be able to detect the face or clothing because of the given angle. The only detection methods we can use on this one are clothing, face, and person detection.

People who painted their bodies - all of the paint on their bodies may affect how the Ai would detect them, whether it even detects a person at all or the face, and clothing. So the detection methods that would be attempted to use would be face, person, and clothing (despite barely any clothing).

Mascot with people: This image is more standard than the others, besides the mascot in the middle. There may be some sort of mistake in attempting to detect the mascot because it is cluttered with people, which is why it is a good thing to test. The person, clothing, and face detection methods are the only types used in this scenario.

The back of people from a distant view - The quality and size of the image are both pretty poor and are taken from behind them, which is a good reason to test if person detection would work because all you can see is their backs. The types of detection methods used are person and clothing detection.

Picture of people's backs with poor resolution - This image does not have the best quality and can barely see some of the people, so this would be a great way to test person detection, to see if it can detect people without most of their body. Face, Clothing, and person detection can all be used for this image.

Most giant selfie ever: This image is unique because it differs in certain sections. At one part, all you can see are faces and others' entire bodies, so this would be a great way to detect a person and face. The types of detection that would be used are the face, clothing, and person detection.

The back of someone wearing fancy clothes - Like the giant selfie ever image, there are multiple sections to this image, which can be tested in different ways. You can barely see people and their faces in some parts, but the central part of the image has different types of clothing, and they are not facing you. So the types of detection methods that can be used are person, clothing, and an attempt at face detection.

I then had all of these images run through our face, person, and clothing detection models, and the results were not too surprising. Depending on the image and type of model detection that was used, there was a considerable variance of success, which ended up showing us that person detection was incredibly effective, face detection worked pretty well as well, and clothing detection can be very finicky, especially if the quality of the image is subpar, because that makes it too complicated, with the amount the clothing model has been trained.

Overall from what we learned today, we need to train the clothing model more, and we could do that by increasing the size of the data sets because that would make it easier to train the model.

02/14/23

Today we revisited the problem of our model's ability to clothing detection. After running numerous tests and using metrics such as mAP to test our model, we decided to retrain it with a larger, more refined dataset. After careful consideration and research, we pinpointed problems with the model/dataset and retrained it to fit our needs. It works much better and can quickly identify clothing, given that the model's view is not obstructed. The clothing detection was one of our significant failures in this project as we first needed to learn how to train, and we also trained it poorly. We had to revisit the engineering/scientific method, re-evaluate our solution, and see where we went wrong. We were delighted after retraining the model and seeing its results compared to its previously trained version.

02/16/23

Today we were planning out what we would put on the poster board. We brainstormed for a while and came up with a 4x4 board and 16 titles of the slides. These were the ideas we came up with for the sections:

- Background (1+2)
- Determination of Membership via Uniform Compliance
    - Goal Statement
    - Details
    - Use Cases
- Models/Data Sets
- The capability of the YOLOv8 Object Detection Model
- Great Image Examples
- Mediocre Image Examples
- Bad Image Examples
- Pipeline
- Confusion Matrix + F-1 Confidence Curve
- The downside of the Deep Fashion Model to the YOLO model
- YOLO Architecture + Timeline
- Future Extensions & Applications
- Analysis
- Conclusion
- Alternative approach

02/18/23

Today we started working on slides. We got through the first six slides, which took a moderate amount of time because we wanted to find a good color theme and ensure the background and goal statement was clear because those two things are essential to start the presentation.

02/20/23

We worked on the following six slides today, and the main emphasis on these slides was to find good examples for each category we had, great, mediocre, and wrong; for the clothing modelfindingnd images that would work great was hard at first. If the resolution were good, it would work better. However, if the image were missing any body part or cut off in any way because of the way the image was taken, it would not detect both the pants and the shirt. Another way errors were caused was if they were crossing their arms, or doing anything besides standing straight. After we figured out the criteria for us to have a great result, it was reasonably easy to do so. We would have to remove any watermarks that may not be on the image and convert the image to a jpg because that is the type of image the model accepts. Overall we learned a lot about how an Ai works with datasets and how to fix minor errors.

02/22/23

On day #3 of working on slides, we worked on the final four slides, which were:

- Analysis
- Conclusion
- Future Extensions + Applications
- Alternative Approach

All of these slides we did were very important to the project because explaining our goals clearly and concisely for our conclusion is vital because it helps us plan for the future. Also, knowing the results is necessary to know how to improve them. So we spent much time on these slides to ensure they were well written for others to comprehend them fully.

Our plan was exciting because we thought we could eventually train our model with a custom-made data set for a specific company/school logo, e.g. (FedEx, AMSA, Military, Walmart, etc.). Moreover, for us to create our own business would be an exciting thing to accomplish.

02/24/23

Today I did a couple of things. First, I updated the GitHub by making a new README, added images like a confusion matrix, created a QR code for our GitHub, organized our GitHub, and added our slides to our GitHub.

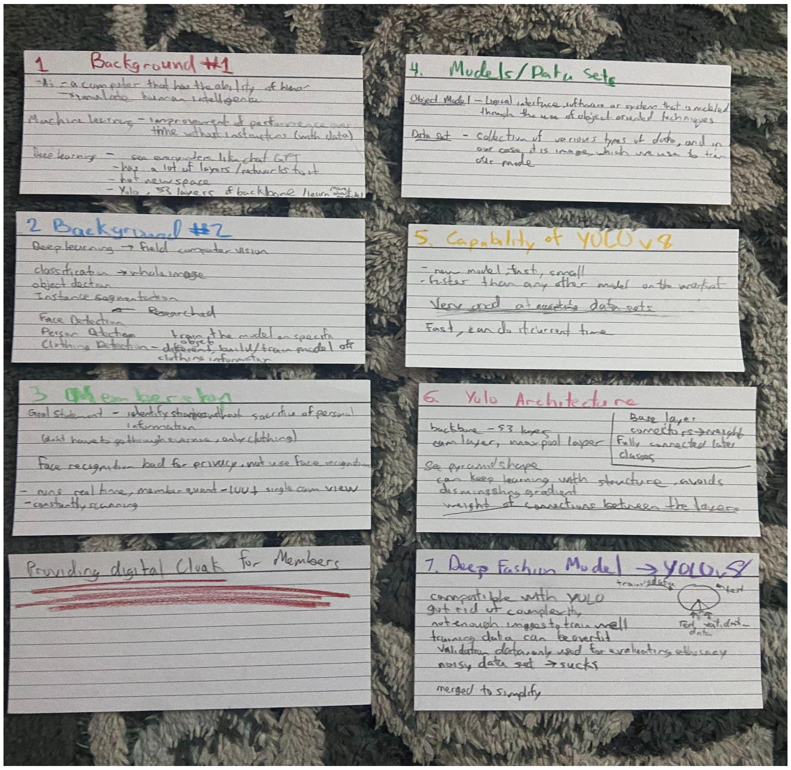
The goal of making the new README was to put all of the essential/interesting information there for people who would scan our QR code at the presentation because it would be the first thing anyone would see. In the README, I put our Goal Statement, Details, and Use Cases of our project because that is the easiest way to understand our project. Then underneath that, I added the confusion matrix and f-1 confidence curve for people to help visualize its effectiveness. Overall it would be the most effective way for someone completely new to the topic to understand what it is, and how well it works.

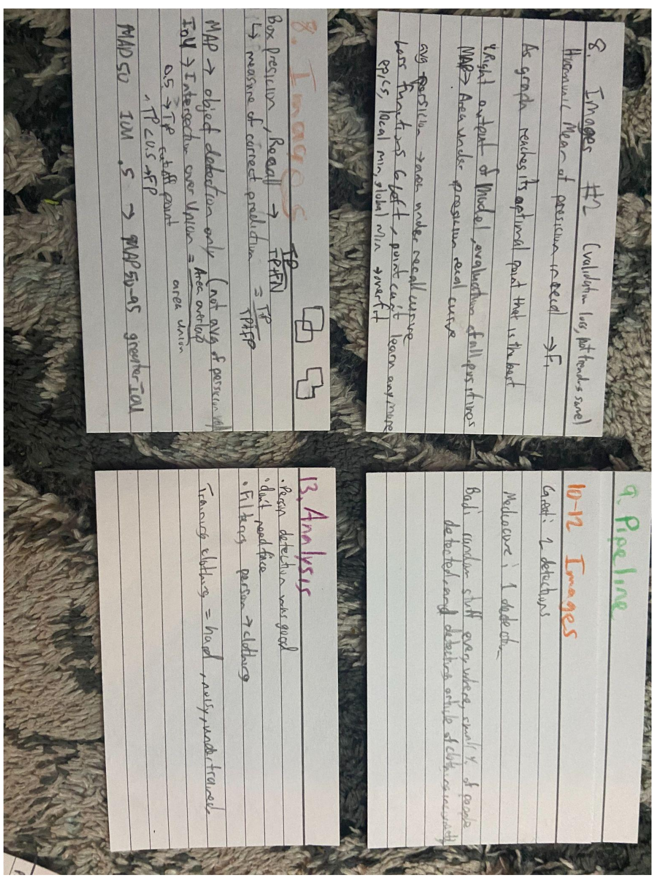
Did some organization as well to make sure the correct README was shown so the images in the new readers could be added. I had to make a folder called images because that was the easiest way to add them to the markdown file.

Other resources that could be of use were also added for anyone that may need a refresher on the project or forgot/missed anything at the presentation. The slides were uploaded for that exact reason, and a QR code to bring you back to the home page of GitHub, just in case someone gets stuck on a certain page/does not know where something is. Later (prior to our regional presentation), we will add the journal to GitHub as an additional page for people to use to help them understand what we have done.

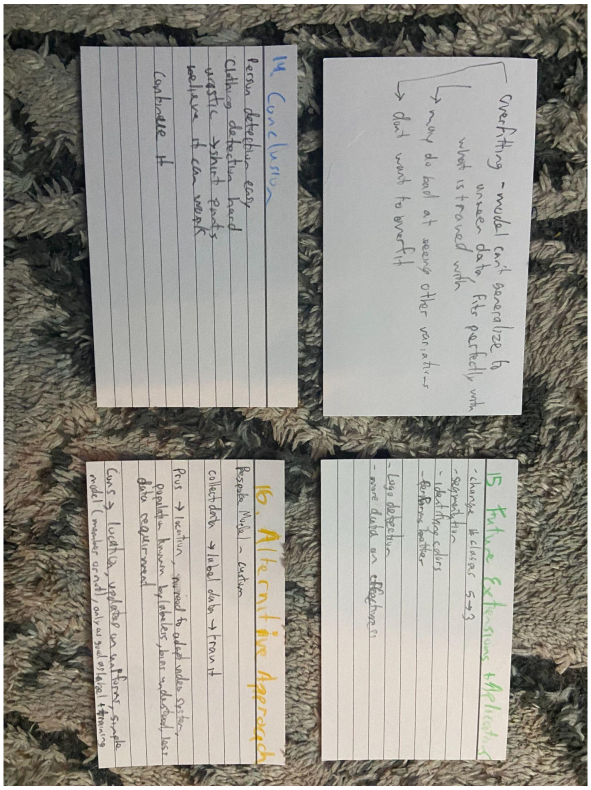
3/5/23

Today I looked through our slides and added other information I thought would be essential while presenting to note cards (to help review or potentially use during presenting if I blank). I also found some pictures I could use to hand out to the judges or other people viewing our project. I also came up with an idea to bring a whiteboard to give a live explanation of something that may need to be clarified or easier to understand with a live visual representation. Below are the note cards for each slide I created. (Sorry, it is hard to fit on the page; it looks wonky).



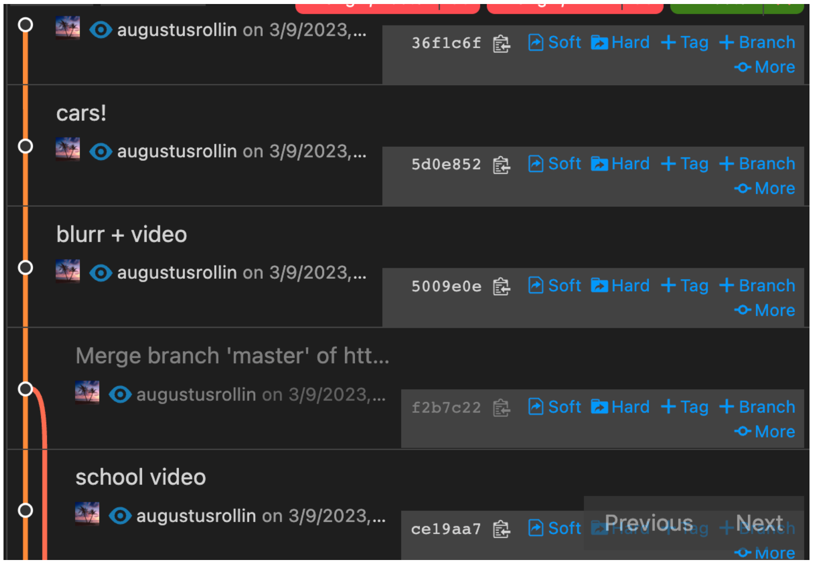


Augustus Rollin & Marrio Jeyakumar



3/6/23

Today I went through our GitHub and added a bunch of new images, consisting of a lot of new blurred images, and I also added a lot of different videos showing object detection. I made many mistakes on this because it was my first time uploading a video onto GitHub; I needed to know the max amount of mB per video you could upload and the max amount of mb(1000 or 1 GB) you can commit at a time. So when I attempted to commit a video that was 535mb, I attempted to use the GitHub app instead of VS code to commit it, but that did not work either. I also attempted to rename one of the files, and delete a couple to see if I commit that first, and then commit the other videos, but less at a time. However, that did not work either; it caused a backlog of commits. So I did some research and found out there is a history extension you can add to VS code which I ended up adding. That enabled me to eliminate those previous two commits, add back all videos less than 100mb, and commit the videos. (To understand what it looked like, here is an image of the history area).



I also added one of those videos to our readme just for viewers to be more engaged and help them understand little aspects about bounty boxes, object detection, and face blurring.

Bibliography

https://github.com/ultralytics/ultralytics
https://github.com/tsennikova/fashion-ai/blob/main/clothes-type-recogition.ipynb
https://github.com/awskids/detect-membership
https://github.com/switchablenorms/DeepFashion2
https://github.com/zalandoresearch/fashion-mnist
https://www.marktechpost.com/2022/11/18/top-object-detection-algorithms-and-libraries-in-artificial-intelligence-ai/

Augustus Rollin & Marrio Jeyakumar

https://user-images.githubusercontent.com/26833433/212094133-6bb8c21c-3d47-41df-a512-81c5931054ae.png
https://learnopencv.com/ultralytics-yolov8/
https://www.datacamp.com/blog/yolo-object-detection-explained
https://blog.roboflow.com/difference-semantic-segmentation-instance-segmentation/
https://raw.githubusercontent.com/ultralytics/assets/main/yolov8/yolo-comparison-plots.png
https://towardsdatascience.com/clothes-classification-with-the-deepfashion-dataset-and-fast-ai-1e174cbf0cdc
https://visualdata.io/discovery
https://blog.visualdata.io/popular-computer-vision-datasets-fashion-68784fdab723?gi=cc0b8866284d
http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html
https://aws.amazon.com/machine-learning/what-is-ai/
https://www.ibm.com/topics/artificial-intelligence#:~:text=Artificial%20intelligence%20leverages%20computers%20and,capabilities%20of%20the%20human%20mind
https://www.ibm.com/topics/machine-learning
https://www.ibm.com/topics/supervised-learning