Video Anonymizer & Machine Learning Model Data Analytics - End Project

César Marrades Cortés

Contents

1. Abstract

This document presents the possibility of anonymizing sports videos, and assessing the accuracy of the outputs with Machine Learning. Collaborating with the NCCA, it covers their specific business case where their main objective is to preserve identity of their students before the examination videos presented for grading are visualized by the agents on the organisation, in order to comply with the new GDPR regulation. Combining both, Development and Data Analytics perspective, the project is divided in two different phases.

On a first stage, a Video Anonymizer process has been developed in C# and is delivered as an executable file (.exe). Supported by two different libraries (Accord and Emgu), given one video input, the optimal target is to generate an output video where any visible faces from the students have been recognized and blurred. 100 video inputs have been used for this stage resulting in 200 outputs (1 for each library).

Processing 200 hundred outputs involved a considerable amount of compute time, although performance measurements are out of the scope of the project. A micro AWS instance was initially setup, later on scaled to a large instance. A second machine was added at a later stage to reduce processing time down to 8 days.

A second component has been developed using WEKA, aiming to predict the accuracy of the outputs from the first component, based on attributes manually extracted from them, such as Camera quality, Camera Angle, Sport, and others that can be found in sections below. This model also hints limitations and boundaries on the libraries used.

The integration of input and outputs from both components is done manually, although there is wide scope for development and automation. This project not only demonstrates the capabilities and limitations of the two libraries used by the first component by stressing them under a multiple set of environments and conditions, such as multiple sports (soccer, karate, boxing, etc.), but also sets a base ground for NCCA research, automation and improvement on a full video anonymization system combined with a Machine Learning component that aims to predict the quality of the blurred videos. The output model provided a 94% accuracy using Naïve Bayes Classifier, with a 100% sensitivity, meaning it may be likely to predict, given a video, how feasible is to produce a perfect blurred output.

2. Literature Review

Research on this specific business problem has shown how Emgucv has been used on many other face recognition applications by researchers on the field. The following points, extract of the “Master Thesis” of Suad Hajr Ahmed Omar (2016), are a subset of the limitations he discusses about the library:

- The distance of the person from the camera should be in 1 to 3 feet for better result.
- The system will be applied and tested in a fix environment with ordinary illumination.
- The problems such as sunglasses, eyeglasses and other accessories that can partially or fully cover the face are not subject for face recognition.

These are discussed on later sections of the document. Time was also one of the constraints of the project, reducing the available error margin when processing 200 outputs as the basis for the second phase of the project. The "Kodak Facebook Collage Project" by Team, T. B. D., also had similar constraints and bottle neck concerns when developing their project. The team cites on his paper: "*High level language wrappers for OpenCV were evaluated and the team found that EmguCV, a .NET wrapper, was among the most stable and complete.*"

The paper "RECOGNITION OF FACE DETECTION SYSTEM BASED ON VIDEO by "SARI ABDO ALI MOHAMMED also chose Emgucv for his system, although he also remarks limitations on what appears to be one of the best libraries: "*there are many challenges that trouble researchers in this field. A lot of effective factors can limit face detection and recognition. These factors affect the appearance of face such as enlightenment, face poses, occlusion (sunglasses, hairstyle, make up), and face expressions and camera quality.*"

After analysing which properties of the inputs may have an impact on the quality of the produced results, this document sets Emgucv as a base for the face recognition system, and offers a new angle on its limitations.

3. Introduction

The project addresses a specific image processing problem presented by the NCCA. Exams tests are often recorded by students in video format and then evaluated by NCCA. This project aims to develop a face recognition system that shows the possibility of pre-processing videos prior "NCCA evaluation", and anonymize them, by using face recognition libraries that will detect students faces that will be blurred. This is obviously a main concern for the organization as identity preservation is one of the main shifts of the new GDPR regulation coming into play in May 2018.

The final goal of the organization is to have a system that allows to automatize the process of anonymizing videos to preserve the identity of their students and avoid biased marking when evaluating the contents.

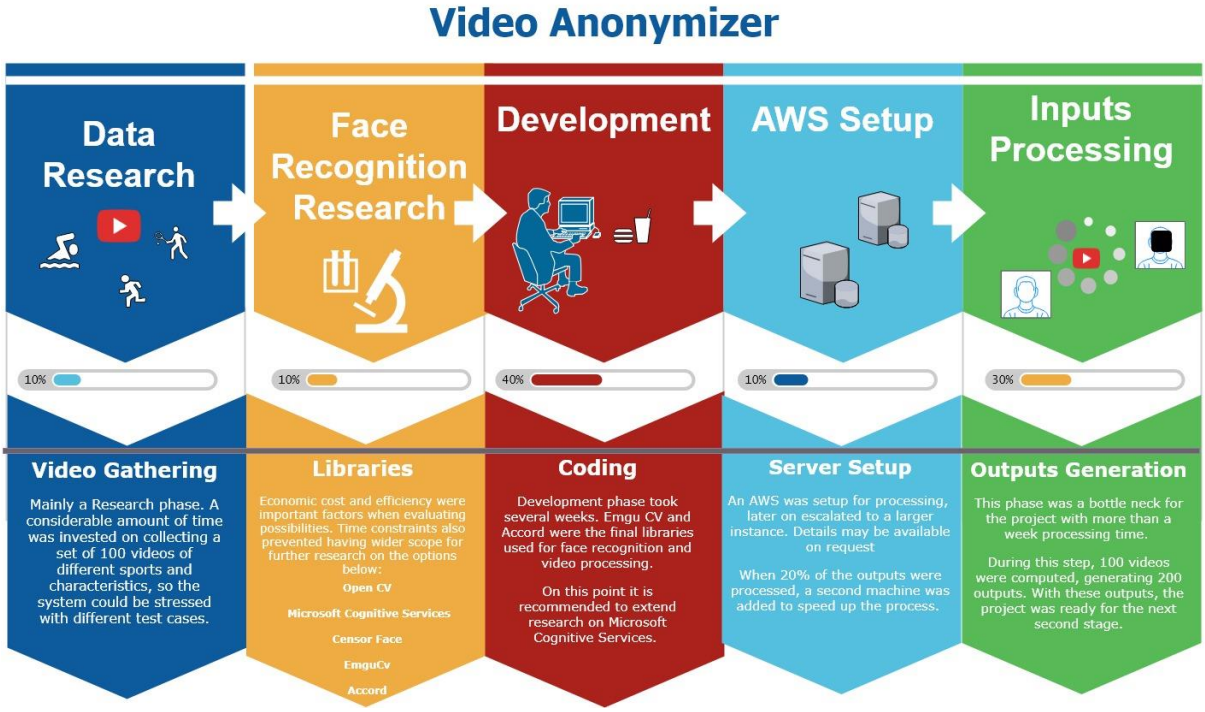
This document not also prioritizes findings and recommendations for future development and integration purposes over the big picture of a software engineering product, but also tackles the problem by isolating responsibilities and addressing them from two different perspectives: software and data analytics. Where the first one oversees video anonymization, the second one tries to find limitations and boundaries on facial recognition libraries used, and provides a prediction on the quality of the outputs processed by this first component.

The project is divided into two main phases, which at the same time are broken into sub-steps. Each of these had to be completed in order in order to continue with the next one.

3.1. Diagrams

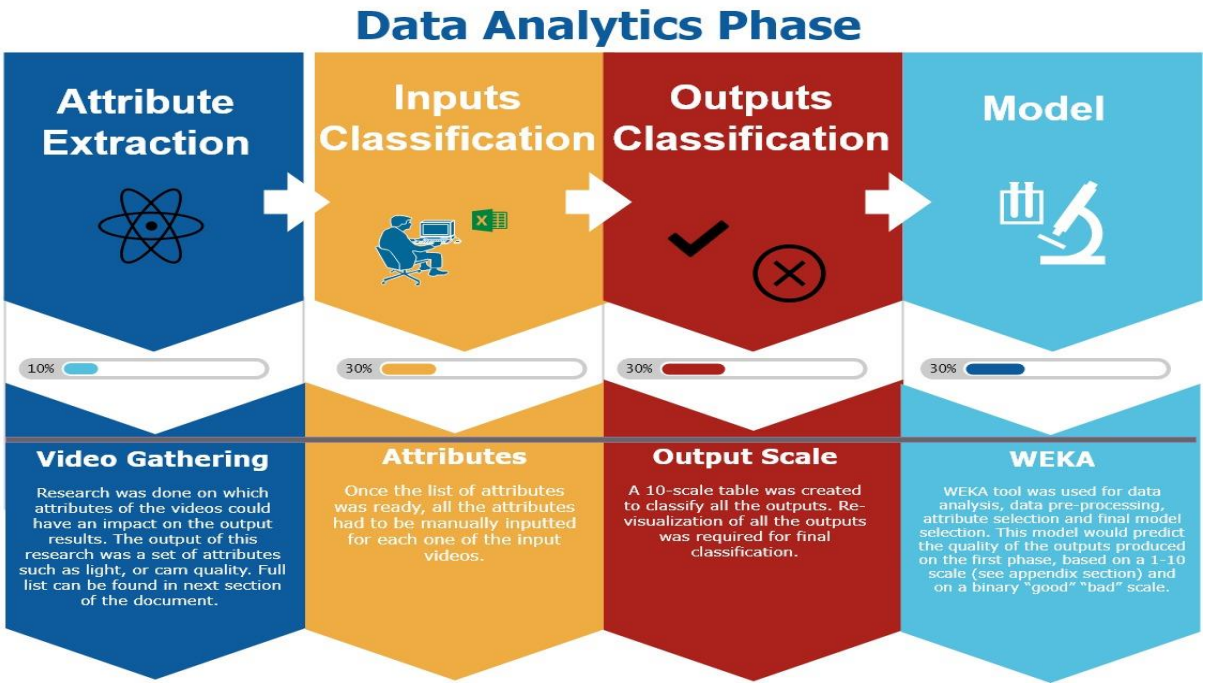
The diagrams below offer a comprehensive view of all the phases and sub-phases of the project, where this document conforms the final one gathering all information and results. Each of these phases is expanded on the next section.

First Phase, Video Anonymizer:



On the first phase, the problem was approached from a Development perspective, by isolating the face recognition problem and video anonymization into an executable file. Once the development phase finished, 2 machines were required to bring down the processing time to 8 days. This process does not aim to solve the problem, but to set a base line of accuracy and performance using third party libraries, and provide some insights on their limitations, and recommendations for future research and development.

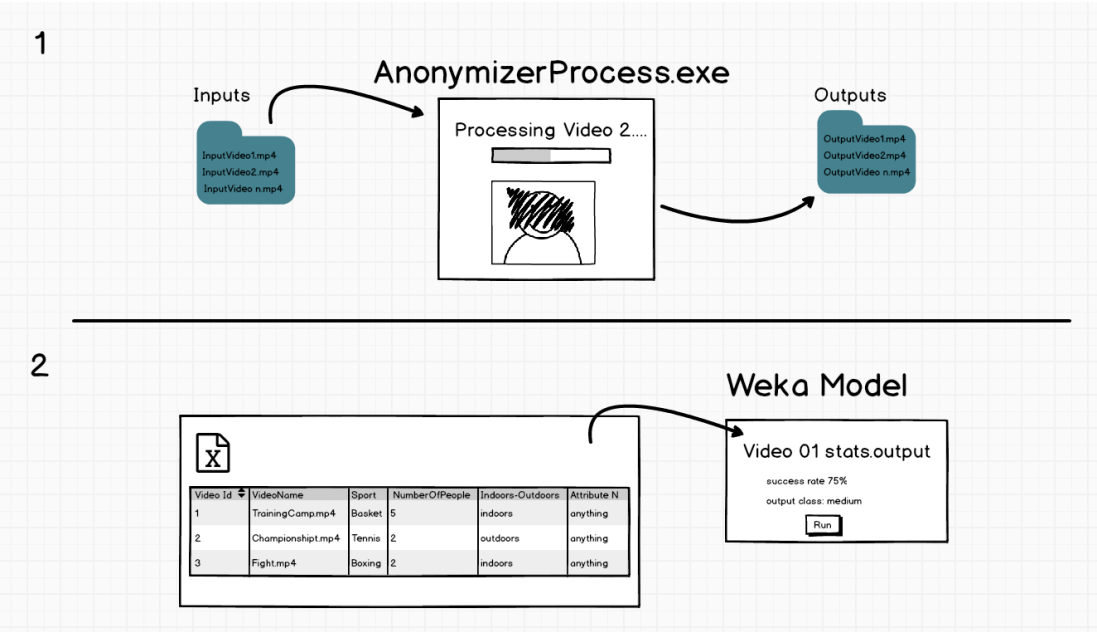
Second Phase, Prediction Model:



On the second phase, the problem was approached from a Data Analytics perspective. Inputs were analysed, and as a result, attributes were manually extracted. Output results were labelled according to the scale that can be found in sections below. Having a dataset with all this information, we then generated an *".arff"* file and started working with WEKA tool, which we used for data analysis, data pre-processing, attribute selection and final model selection. This model would predict the quality of the outputs produced on the first phase, based on a 1-10 scale (see appendix section) and on a binary "*full*" "*partial*" scale.

These two marked phases took an approximate equal amount of time. Processing the videos, marked as 30% of the first phase, took more than 7 days.

Architecture Sketch



The picture above shows two differentiated sections. The first one visualizes the workflow process of the "Anonymizer" executable, responsible of "blurring" students faces on the videos. As we can observe on the picture, the process will blur any videos in the inputs folder and outputs are generated in the outputs folder.
The second section corresponds to the second phase of the project detailed above. Inputs properties are processed by the model that predicts whether the output is "fully" or "partially" blurred.

3.2. Component Breakup

- Video Anonymizer Component (C# executable file) (phase 1).
    - The executable file has been developed using C# using EmguCv and Accord libraries. The tool processes each video generating two different video outputs (one per each library), where on a perfect scenario, student faces cannot be recognized.

- WEKA model predicting accuracy of anonymizer process (phase 2).

3.3. Inputs

- 100 YouTube Videos for the anonymizer process (mp4 format).
- Excel sheet with a set of agreed attributes for each of the videos

3.4. Outputs

- 2 sets of 100 anonymized videos. 2 hundred videos in total (mp4 format).

- Weka prediction on video blurring success (fully or partially blurred).

3.5. Comments

This section covered a general overview of the two main phases of the project, and steps. The next section analyses in further detail the steps followed for each of the phases.

4. Methodology

This section breaks up the different phases of the project providing a more granular overview with technical details and specific results. As discussed above, there are to main differentiated phases:

- First, the Video Anonymizer process.
- Second, the Prediction Model

The second phase was developed using two different configurations, in an attempt to produce the best possible prediction model. The first configuration is based on a 1 to 10 scale, rated from worst to best possible output, and the second one based on a binary scale, assessing whether outputs are "fully" or "partially" blurred. These two different configuration approaches and result comparisons are detailed in under the "2nd Phase - Prediction Model" section.

4.1. 1st Phase - Video Anonymizer

4.1.1. Research and Coding (Anonymizer.exe)

The output of this process is, based on one input video, to produce a new output video where, after applying face recognition techniques and image filters, existing faces cannot be recognized.

Few libraries were analysed on this stage. Research was done on the following:

- Microsoft Cognitive Services.
- Censor Face (pay per usage)
- Open CV (free)
- Emgu CV (free)
- Accord

Microsoft Cognititive Services option was discarded due to high costs associated to video processing. Also, this API is optimized not for video processing, but for image processing. Where both can be presented as the same problem, video processing requires heavy payloads and longer processing times than single images, which would have extended the first phase of the project for months.

Free versions were taken as a start base, and **Emgu** and **Accord** were chosen based on available documentation (accord) and social feedback research (emgu).

The tool that anonymizes videos is presented as a Console ".exe" file, written on C#. Once the process starts, it processes all the input videos located on the "input" folder (configured on the settings file), generating 2 outputs per each input (1 per library).

As the internals of the program are out of the scope of the project, no further analysis is presented.

No performance tests have been run, although there is room for performance improvements on parallelization and/or multithreading.

Further research on Microsoft Cognitive Services is recommended for base comparison purposes. Further recommendations can be found on the last section of this document.

4.1.2. Data Gathering

A set of 100 random sport videos have been extracted from YouTube platform. The initial approach was to provide a variety of different scenarios that would cover the most common cases in NCCA, when evaluating their students for exams. Longer videos require longer processing time, therefore shorter videos had a preference over longer ones. On these premises, the whole batch of videos is classified in 10 different sports, indoors and outdoors conditions, a range from 1 to more than 10 persons, recorded on mobile and normal cameras, on static and dynamic movement conditions and on multiple different field types.

The 10 sports have been chosen without knowing the final real business cases, trying to address the problem from a wide variety of possible scenarios and conditions. The following table offers a breakup of the different sports with a minimized overview of generalized tendencies found after inputs visualisation:

Sport	Conditions
Basketball	Indoors, outdoors, dynamic cameras, medium and far distances
Karate, Judo	Indoors, multiple backgrounds, multiple angles, tendency to close to medium distances, head protectors
Boxing	Indoors, multiple angles, head protectors, referee. Close, medium and far ranges
Hurling	Tendency to far distances, high angles, with multiple people
GaelicFootball	Tendency to far distances, medium and high angles, with multiple people.
Tennis	Multiple angles, mixed cam movement types, high angles
TableTennis	Medium and far distances,
Soccer, SoccerInterior	Multiple angles, mixed cam movement types, medium, high angles
HighJump, HighJumpPole	Medium angles from close and medium distances and different camera types
WeightLifting	Static cameras, from close and medium distances and medium angles

This information indicates different sports have a tendency of sharing common patterns, although different inputs from same sport have multiple combinations of attributes.

The table below also offers further details on specific value ranges, scenarios, and other attributes that are considered to have an impact on the output results, all used when working on the second phase of the project.

All this information was gathered on a file, along with a unique identifier for each video, length, and a final “class” attribute for later classification, which is submitted along with this document:

Attribute Id	Value	Values	Description
1	Id		x
2	Sport	{Basketball, Karate, Judo, Boxing, Hurling, GaelicFootball, Tennis, TableTennis, Soccer, SoccerInterior, HighJump, HighJumpPole, WeightLifting}	x
3	InOut	{in, out}	Whether sport is indoors or outdoors
4	Light	{clear, dark}	x
5	CamType	{mobile, other}	x
6	CamQuality	{low, medium, high}	x
7	CamMovement	{static, steady, dynamic}	Static: cameras without movement Steady: just rotation or zoom Dynamic: camera in movement
8	CamAngle	{medium, high, mixed}	High: Camera is located above the head level Medium: at head level Mixed: combined

9	CamDistance	{close, medium, far, mixed}	x
10	People	numeric	Max number of people in frame
11	HeadProtector	{0,1}	Whether people is wearing any kind of head protection that may influence on the output result.
12	FieldType	{court, floor, grass, mat, sand, mixed}	x
13	FieldColor	{black, blue, gray, green, lightbrown, lightgray, mixed, red, white, yellow}	x
14	BackgroundColor	{blue, brown, crowd, gray, green, lightbrown, mixed, red, white}	
15	Crowd	{0, 1}	Whether video has crowd in the background
16	Referee	{0, 1}	Whether there is referee
17	GroundPosition	{0, 1}	This attribute was an intend to idenfity sports where persons would be lying on the floor, such as combat sports.
	Class	{fully, partialy} and scale from 0-10 (scale can be found in appendix)	Output class

***Video Attributes table**: As the table above shows, there are 18 attributes including the "class". Description column expands information about the possible values and their meanings. All this data had to be manually imputed conforming, along with the "Class" attribute, the basis of the final prediction model.*

The table below was used to classify the outputs on the Data Preparation Phase, based on noise and accuracy:

Rank	Description
1	no accuracy, high noise
2	no accuracy, noise
3	extremely low accuracy, noise (any)
4	low accuracy, high noise
5	low accuracy, noise
6	good accuracy, high noise
7	good accuracy, low noise
8	almost no failures, noise (any)
9	no failures, noise
10	no failures, no noise

***10-scale Classification table**:* Noise is considered a relevant factor, as many of the outputs present what is commonly known as "false positives". This aspect becomes also relevant for lower grading when no accuracy is found.

Although evaluation may be subjective to interpretation, the following rules were used when grading outputs with the scale above:

- "Extreme low accuracy" was considered when nearly none of the faces were recognized on the video.
- "Low accuracy" was considered when some of the faces were recognized in some parts of the video.
- "Good accuracy" was considered when faces were recognized many times.
- The 7th mark was used to consider when most of the parts of the video were successfully blurred, but still with failures.
- The 8 mark was used for videos with almost no failures
- The 9 mark, the maximum found on the dataset, was used for a video where nobody could be recognized, but with the "noise" due to what is known as "false positives", illustrated on the image below with a red X mark:

It is also important to remark as it will be shown later on, that most of the videos were marked lower and only 10% of them were graded above 7, with only one rated as 9.

Some of these tasks were done in parallel while the "*Anonymizer*" component was already processing the videos required to fill in the dataset that would be used on a later stage to create the model.

4.1.3. Data preparation (AWS Setup / Inputs Processing)

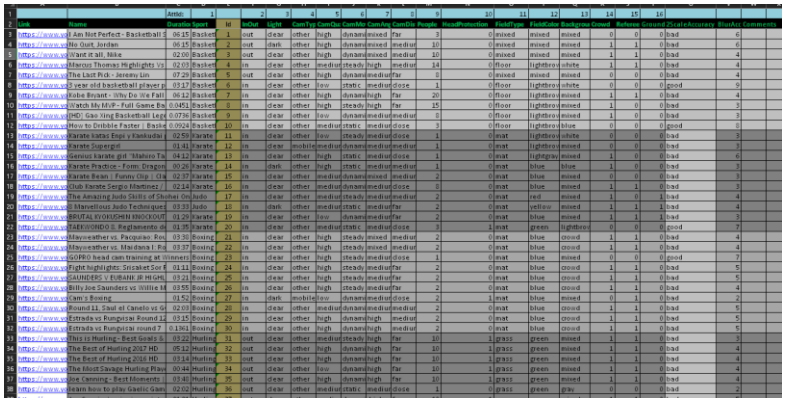
Processing 200 hundred outputs (1 per each library) took a considerable amount of time. A micro AWS instance was initially setup processing the 100 inputs located in the "inputs" folder. This instance was later on scaled to a large instance as the process would max out the CPU. After few days processing inputs, a second machine INTEL CORE i7 2.60GHz was added to speed up the process. Processing 200 outputs took around 8 days following these steps. No performance measurements have been taken. There is however room for performance improvements, detailed in sections below.

The generation of the dataset was done on two steps:

1. Input properties were *inputted* manually by using the "**Video Attributes table**" on the previous section. This task was done while the "VideoAnonymizer" was generating the outputs. Visualization if **inputs** was required to fill in this information.

2. Once the "Anonymizer" component generated the 200 outputs, the accuracy of these was assessed and filled in the "Class" attribute column, along with the rest of attributes in the file previously inputted, using the "***10-scale classification table***" on the previous section. Visualization of **outputs** was required to fill in this information.

The screenshot below shows a partial dataset delivered along with this document, containing all the information of the outputs:



4.2. 2nd Phase - Prediction Model

This section will describe the steps followed to come up with the final prediction model. Due to the low accuracy obtained by the Accord library, the video outputs generated were dismissed for the creation of this model, meaning, the following tests and steps of this document are based on the output videos and dataset generated using Emgu library.

As explained at the beginning of this section, this point is also divided in two subsections, detailing the results of two different configurations. The **first sub-phase** (First Configuration - 10 Scale Output Class), describes an attempt to create a model with the **10-scale output class**. As the prediction results produced by this model were very low (slightly above 50% accuracy), a **second sub-phase** is described on (Second Configuration - 2 Scale Output Class) section. A new configuration was created by rescaling the 10-scale output class into a **binary output (fully / partially)** blurred **scale**. Both phases are detailed below step by step. Comparison of final results from both models can be found on the Results section.

4.2.1. First Configuration - 10 Scale Output Class

4.2.1.1. 10 Scale Output Class - Data preparation

Once the excel file was prepared, no major changes were required. Some of the colours were readjusted to remove unnecessary elements. Once the manual errors were manually fixed, the dataset was converted to a WEKA file, the tool used for this second phase.

Description

There are no missing records, or missing values, as they all have been provided by hand after the anonymizer process finished processing the initial set of inputs. On these premises no outliers will be affecting the model.

There are 100 instances. Visualisations below provide an initial understanding on the most relevant attributes in the dataset:

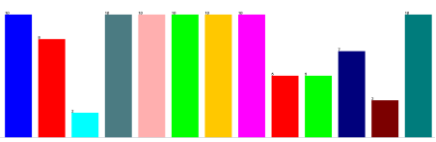


Figure 1: Sport

We are picking here 10 different sports and we added 3 other sub classifications such as (football indoors, high Jump Pole, or Judo from Karate). This graph shows the data is equally distributed.

Figure 2: In-Out

Indoor on the left, outdoor on the right, data is equally distributed between both types.

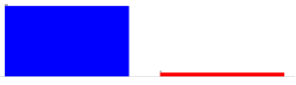


Figure 3: Light

95% of the video inputs have been recorded in good light conditions.

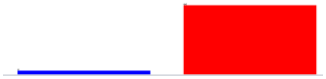


Figure 4: Cam Type

Only 5% of the inputs were recorded with a mobile phone. Other 95 percent are recorded with other camera types.

Figure 5: Cam Quality

CamQuality shows a close distribution across the 3 different types. From left to right, low, medium and high.

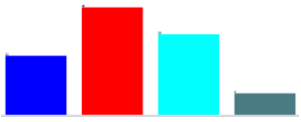


Figure 6: Cam Distance

From left to right, close, medium, far and mixed, show also an equilibrate distribution.

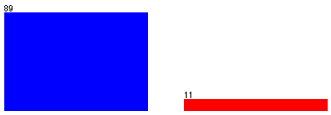


Figure 7: HeadProtector

Predominance on sports without Head Protector.

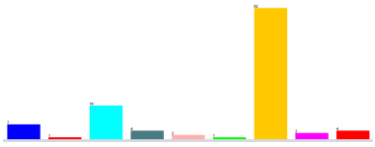


Figure 9: Background Colour

Data shows equally distributed across all the backgrounds, except for “crowd” and “mixed” values that appear to have a significant higher volume. Mixed background colours indicate movement and are related to dynamic cameras.

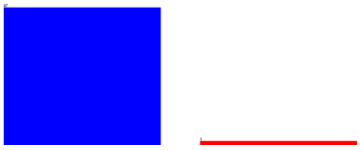


Figure 10: GroundPosition

Graph shows a distribution of the attribute which is on its 97% urinary. This attribute was introduced trying to measure the impact that ground sports would have on the accuracy. There is however very little scenarios presenting this characteristic which has shown to not be one of the most determinant attributes.

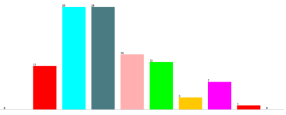


Figure 10: GroundPosition

Data appears a bit skewed to the left. This indicates there is a predominance of processed videos with lower accuracy. The tail appears to be on the right side, meaning, the higher the scale, the fewer the outputs on that rank.

Following assumptions can be extracted based on the graphs: Sports are evenly distributed on the dataset. Videos are nearly 50% divided between indoors and outdoors. Only 5% of the videos are recorded under bad light conditions, and there is a high tendency to record these videos on cameras rather than on mobile phones, as only 5% of them are recorded using a phone. There are very few of the videos where persons are wearing head protections and there is a tendency to keep the same cam distance along all the duration of the video. Recordings are evenly divided with and without referee, and either with or without crowd. It is remarkable the very few high scored videos found on the dataset.

Other attributes omitted on the table, such as Cam movement or Cam angle don´t show any remarkable observations.

4.2.1.2. 10 Scale Output Class - Attribute Selection Model selection

Different attribute selection techniques have been run, trying to gain an understanding on which attributes may have higher impact on the generated outputs. Full screenshot details of these can be found on appendix section.

Correlation Attribute Evaluation

Attribute	Rank
PeopleNumber	0.226
Referee	0.2194
Crowd	0.1973

When running Correlation Attribute evaluation we found the top 3 attributes were PeopleNumber, Referee and Crowd. This means, based on this algorithm, these were the attributes having a bigger impact when producing a blurred output.

Information Gain Attribute Evaluation

Attribute	Rank
Sport	1.1366
FieldColor	0.6423
BackgroundColor	0.5992
FieldType	0.5789
CamDistance	0.5647

When running Information gain however, the most ranked attributes were Sport, FieldColor, BackgroundColor, FieldType and Cam distance.

Learner Based Feature Selection (Wrapper Subset Evaluation)

Rank	Attribute
1	Sport
2	In-Out
3	CamQuality
4	CamDistance

Wrapper Subset Evaluation with Naïve Bayes configuration was also run, ranking Sport, In-out, CamQuality and CamDistance as first attributes.

This last table hints cam quality could be one of the most important attributes affecting the accuracy of the anonymization. Sport obviously plays a fundamental role along with the cam distance, and indoor or outdoor sport types may play an important role when predicting results.

4.2.1.3. 10 Scale Output Class – Model Development

Several tests were performed based on the third attribute selection results (**learner based feature selection**). Algorithms and attribute configurations are displayed on the tables below:

Id	Configuration
1	1 Sport 2 CamQuality 3 CamDistance 4 PeopleNumber 5 FieldType 6 FieldColor 7 BackgroundColor 8 Referee 9 Class
2	1 Sport 2 In-Out 3 CamQuality 4 CamDistance 5 PeopleNumber 6 FieldType 7 FieldColor 8 BackgroundColo 9 Referee 10 Class

Algorithm	Config 1 Accuracy	Config 2 Accuracy
Naïve Bayes	53%	56%
Support Vector Machines (functions.SMO)	43%	43%
Trees J48	36%	36%

The table on the left shows 2 attribute configurations, adding "in-out" property on the second one, while the table on the right shows the accuracy results for different models using both configurations.

It can be observed adding "InOut" predictor has a small increase on Naïve Bayes algorithm from 53 to 56%. It is also remarkable the low performance of this models, therefore Sensitivity and Specificity calculations have been ignored for them.

4.2.2. 10 Scale Output Class Comments

All the steps followed above, with different attribute configurations and based on different attribute selection techniques produced a model with a very low accuracy using different attribute configurations.

Further investigation and analysis was required.

4.2.3. Second Configuration - 2 Scale Output Class

In an attempt to improve the accuracy of the model, the 10-scale output class was analysed closer, aiming to provide a binary output class (fully/partially) blurred. Below its distribution:

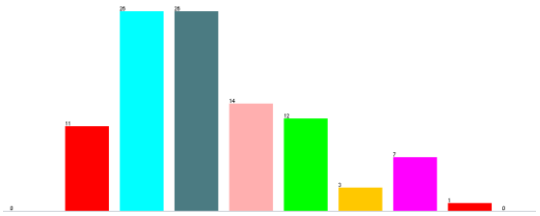


Figure 11

There are some observations that can be done about the data based on Figure 11:

- There are no 0,1, or 10 values.
- The data appears to be slightly skewed to the left, indicating most of the videos are ranked lower than 5. Also there are very little videos ranked as 8 or 9.

When trying to rescale the outputs into "yes/no" classes we run into the problem on where to draw the line. For this purpose, outputs in scale 7 (3 videos) were re-assessed, in an attempt to reclassify them on valid or invalid videos. These were number 13, 20 and 23. After reassessment, 13 was reranked with 6, and line was drawn after this value:

IF(ACCURACY>6,"full","partially")

Rank	Description	Blurring success
~~1~~	~~no accuracy, high noise~~	~~partially~~ (no videos with this rank)
2	no accuracy, noise	Partially
3	extremely low accuracy, noise (any)	partially
4	low accuracy, high noise	partially
5	low accuracy, noise	partially
6	good accuracy, high noise	partially
7	good accuracy, low noise	full
8	almost no failures, noise (any)	full
9	no failures, noise	full
~~10~~	~~no failures, no noise~~	~~full~~ (no videos with this rank)

4.2.3.1. 2 Scale Output Class - Attribute Selection Model selection

Same attribute selection techniques were run with the 2-scale output class: Correlation, and Info Gain. Learner Based was dismissed.

Correlation Attribute Evaluation

Attribute	Rank
Crowd	0.3242
CamDistance	0.3203
In-Out	0.3203
CamMovement	0.2958

Where on the 10-scale output configuration, the highest ranked attributes were PeopleNumber, Referee, Crowd, and CamAngle, we see here a different configuration where CamDistance and CamMovement attributes have more relevance over most of the attributes.

Information Gain Attribute Evaluation

Attribute	Rank
CamDistance	0.137386
BackgroundColor	0.123167
PeopleNumber	0.118877
In-Out	0.101851
CamMovement	0.085085

In this case there is a switch, where on the 10 scale, the most relevant attributes were Sport, FieldColor, BackgroundColor and FieldType, we can see there are new attributes which also match the results from the Correlation Attribute Evaluation (green highlighted).

Comments

Both algorithms have in common attributes that may have an effect to the outputs. Correlation Attribute Evaluation is used as a base for the next step where the data distribution of the highest ranked attributes is examined closer.

4.2.3.2. 2 Scale Output Class – Data Distribution

The histograms below show the highest ranked attributes by the "*Correlation Attribute Evaluator*", with an overlay of the output class (fully vs partially blurred):

Blue: fully blurred
Red: partial blurred

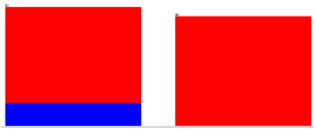


Figure 12: In/Out

Nearly half of the videos have been recording indoors. Data shows how all the "*fully*" blurred output videos are in the "indoors" bucket.

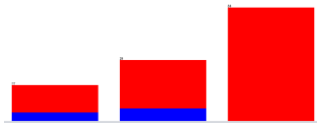


Figure 13: CamMovement

Having first static, second steady and last one dynamic, we find all "*fully*" blurred outputs are on the left of the distribution.

Figure 13: CamAngle

From left to right, medium, high and mixed. Again, all of "fully" blurred videos are in the "medium" bucket.

Figure 14: CamDistance

From left to right, close, medium, far and mixed. We find the same pattern than in the previous attributes.

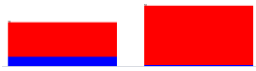


Figure 15: Crowd

Left to right, 0 and 1, where the presence of "*fully*" blurred outputs remains on the same side.

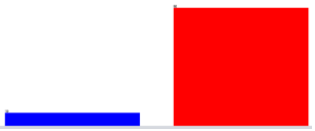


Figure 16: Output class Distribution. Blue: fully, Red: "partial"

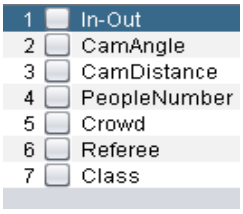
Figure 16 shows a 10% vs 90% ratio.

The distributions show "fully" blurred outputs have very specific characteristics, where the "partially" classes occupy most of the dataset. The plots indicate that all the "fully" outputs are taken from close distances. They also tend to be recorded from static or "steady" cameras, from a medium angle, with no crowd. These attributes will be important to determine whether a video will be successfully anonymized.

4.2.3.3. 2 Scale Output Class – Model development

Several tests were run using Naïve Bayes and J48 Tree with two different attribute sets attending the values obtained from the Correlation Attribute Evaluator on the previous phase. Output model images are can be found on the appendix section.

2 Scale Output Class - Attribute Set 1:

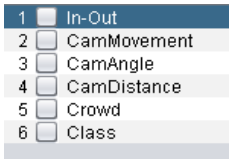


Algorithm	Accuracy	Sensitivity (TP rate)	Specificity (TN rate)
Naïve Bayes (2 scale class ouput)	91%	**100**%	90%
**** Naïve Bayes (10 scale ouput class)****	44%	x	x
Trees.J48 (2 scale class ouput)	89%	30%	95.5%

* *Results for the first configuration (10 output scale class) are shown for comparison purposes in the table above.*

2 Scale Output Class - Attribute Set 2:

Attribute "Referee" was removed to see how it would affect to the performance of the model. Results shown in the table below:



Algorithm	Accuracy	Sensitivity	Specificity
Naïve Bayes	**94%**	90%	94.4%
Trees.J48	89%	30%	95.5%

We can observe using Naïve Bayes, Specificity is slightly higher than using the previous configuration, although sensitivity drops from 100% to 90%. However, "Specificity" still remains at a 95.5% rate using the first attribute configuration.

This means, by using the Attribute Set 1 and based on the dataset, the system should be able to identify 100% of the times when outputs would be successfully fully blurred , and 95.5% of the times when they would be "partially" blurred.

4.3. Statistical techniques

Several areas have been covered on the points above. Having as a base the first dataset with a 10-scale output class, visual inspection on distributions and data correlation was done with no evident conclusions. Different algorithms were run for out attribute selection: Attribute Correlation Evaluation, Information Gain, and Wrapper subset eval. Different models were tested then, with two different attribute sets chosen based on the results of the correlation evaluation. After running Naïve Bayes, Support Vector Machines and Trees 48 on different attribute configurations, the maximum accuracy we obtained was 56%.

In an aim to improve the model, outputs on the 7-mark were reassessed, and the 10-output scale class was rescaled into a binary output class. All the previous steps above were run again resulting into an accuracy of 94%.

5. Results

The tables on the sections above show how different models can perform with 2 different sets of attributes, on two different configurations of output class. The 10-scale output class was discarded with a maximum accuracy of 56%.

After rescaling the output class into a binary variable, it can be observed how accuracy varies from 91% to 94% with two different attribute sets. It is very important to remark on this case that more accuracy does not mean a better model, as sensitivity and specificity are different both cases meaning with this, each of the algorithms is better depending on what is looking for. The table below shows a summarized view of the results obtained with the 2-scale output class:

Id	Attribute Set	Algorithm	Accuracy	Sensitivity	Specificity	Comments
1	1	Naïve Bayes	91%	**100%**	90%	Better for positives
2	1	Trees.J48	89%	30%	**95.5%**	Better for negatives
3	2	Naïve Bayes	**94%**	90%	94.4%	Best accuracy
4	2	Trees.J48	89%	30%	**95.5%**	Same as id=2

"a class": full

This table shows important information about the data. Accuracy column shows an average percentage of how the model would be able to predict “fully” and “partially” blurred videos. On this context, the third configuration has the highest accuracy.

On this case, Sensitivity focuses on predicting the “fully” blurred videos. Based on this value, the first configuration will recognize 100% of the “fully” blurred videos.

At the same time, J48 three shows 95.5% of Specificity, meaning with this, the model would predict 95.5% of the times when a video would be “*partially*” blurred, therefore the second option is better when looking for negative patterns.

It is also important to remark the distribution of our outputs when having a 2-scale output class may not be the best one. Simplicity of the tree algorithm in this case confirms our first assumptions when processing the videos. That is, videos from close distances with static cameras tend to have a high positive rate. We can observe the pruned tree generated by the J48 algorithm:

```
CamDistance = close
|   CamMovement = static
|   |   In-Out = in: full (5.0/1.0)
|   |   In-Out = out: partial (2.0)
|   CamMovement = steady: full (8.0/2.0)
|   CamMovement = dynamic: partial (7.0)
CamDistance = medium: full (40.0)
CamDistance = far: partial (30.0)
CamDistance = mixed: partial (8.0)
```

Number of Leaves: 7
Size of the tree: 10

This tree also emphasizes the importance of the CamDistance and CamMovement attributes and how they are mostly determinant to recognize outputs with high face recognition accuracy.

6. Discussion and Conclusion

6.1. Overview and Recommendations

The system described in the sections above tried to solve two specific problems:

- Anonymizing sport videos.
    - In this phase many limitations arose when using EmguCv and Accord Libraries. The initial guess was, close ranges and static videos had a better success rate.

- Predicting whether these anonymizations were fully or partially blurre
    - On a second phase, outputs were tackled from a Data Analytics perspective aiming to provide a prediction of the quality of the results based on properties from the inputs. Research and tests on this area confirmed the first assumptions and highlighted the impact that some attributes had on the produced outputs.

The points below overlay the steps followed with a production scenario, in order to identify areas open for improvement and automation:

a. Inputs Gathering face

Manual process followed in this project would be easily replaced by an agent uploading them to the system. System space could be a constraint on a production scenario depending on the amount and frequency of inputs.

b. Input Properties-Attributes extraction

This step was carried out manually. For a full automated system, this could be one of the most challenging problems to solve, although no more than a quick video inspection is required to fill this information.

On a full integrated system, Microsoft Cognitive Services can provide this information for specific frames, although it is not granted any snapshot data from the video will remain constant.

c. Processing Video Inputs

System Design

The component in charge of this has been presented as an executable file as briefly detailed in first sections. Recommendation here is a system that will allow uploading these inputs to the server file system. Basic UI is recommended for this step. Once the file has been successfully uploaded, a service "*listening*" to this folder should automatically process the video and produce the output on a different folder.

Library Limitations

The points below elaborate a set of limitations detected when processing the videos on the first phase of the project:

- It has been observed height camera angle has a high impact on accuracy, as higher angles present lowest accuracy when detecting faces. This can be observed on video 26. The video is recorded using different heights. E.g. second 2:09 (higher) vs 2:27(medium). Both sequences show how the referee has a lower accuracy ratio when the camera is higher than when the camera is recording at the same height than the head.

- Faces close to 45 degrees angle or close to this angle show high rates of error. Therefore, another technique is recommended for these scenarios, such as full body recognition and partial anonymization.

- The library also shows difficulty to recognize faces when arms or other elements are partially interfering with the head. This can be observed on tennis videos when players are serving. This indicates this specific problem should be addressed using a different approach.

- The system also shows difficulty to recognize faces on some scenarios where people are wearing head protections.

- A face looking down won't be detected depending on how tilted the head is. E.g. video 48, second 00:39. When played at slow motion it can be appreciated the face of this player is only recognized when he raises his head up facing the camera.

- Focus is obviously another element impacting the accuracy. Faces stop being recognized when faces are shown as unfocussed or blurred.

Main recommendation here is adding body recognition that will allow the system to partially anonymize faces based on size calculations. This would positively affect the accuracy of the outputs and minimize the scenarios above happening, although would also have a negative impact on the time required to process one single file.

Performance

The presented tool iterates each of the frames and process them one by one. On these premises, there is also potential room for speed optimization by parallelizing frames computation.

d. Predicting outputs accuracy

This step is done through WEKA, although there is a wide range of options for full integration. Python or C# are recommended options to build a component that would complete the automated pipeline.

6.2. Conclusions

Given our initial problem, this document sets the base ground for its resolution, and provides an architectural option while analyses the outputs generated. It also identifies areas with limitations and other points open for improvement and research. It is important to remark this project does not seek to develop a software system that anonymizes faces, but to demonstrate the feasibility to develop one based on the provided specifications, limitations and time constraints.

It also has been detected room to improve the accuracy obtained on the anonymization process. Recommendations section above provides guidelines and specific instruction in this area.

The model tries to predict accuracy on video outputs based on properties extracted from them. The library however, has defined limitations (exposed on the “Library Limitations” sub-section) that are not factored into these attributes. Therefore, whether there is any pattern affecting the accuracy of face recognition algorithms can get diffused with boundaries and limitations of this library. When overlaying this point with a real production case, a business analyst should draw a line at some point between library limitations and patterns that may have an impact on accuracy. Further open discussion on this topic is suggested.

High accuracy, sensitivity and accuracy obtained from the model generated may suggest there is a clear problem to be tackled and analysed in depth, but also confirms the importance that basic properties, such as closeness or movement of the camera have when working with face recognition algorithms.

6.3. Ethics

Face recognition is a global field where word wide governments invest millions in an attempt to isolate the perfect model, and in many cases aiming to have full control of the data. Although this project tackles a problem from an identity preservation point of view, it has a wide range of applications out of the NCCA´s specific business case, as nowadays identity is one of our the most valuable assets we have as members of our society.

7. Appendix

7.1. Images extracted from Weka

Image	Naïve Bayes
S Scale Output Class Naïve Bayes	Correctly Classified Instances 91 Incorrectly Classified Instances 9 Kappa statistic 0.6429 Mean absolute error 0.0923 Root mean squared error 0.2228 Relative absolute error 49.3687 % Root relative squared error 74.2407 % Total Number of Instances 100 === Confusion Matrix === a b <-- classified as 10 0 | a = good 9 81 | b = bad
2 Scale Output Class Trees.J48	J48 pruned tree ------------------ CamDistance = close | In-Out = in | | CamMovement = static: good (5.0/1.0) | | CamMovement = steady: good (8.0/2.0) | | CamMovement = dynamic: bad (5.0) | In-Out = out: bad (4.0) CamDistance = medium: bad (40.0) CamDistance = far: bad (30.0) CamDistance = mixed: bad (8.0) Number of Leaves : 7 Size of the tree : 10 Correctly Classified Instances 89 Incorrectly Classified Instances 11 Kappa statistic 0.2949 Mean absolute error 0.1579 Root mean squared error 0.3046 Relative absolute error 84.4734 % Root relative squared error 101.5067 % Total Number of Instances 100 a b <-- classified as 3 7 | a = good 4 86 | b = bad

7.2. 10-scale Output Class Table

1	no accuracy, high noise
2	no accuracy, noise
3	extremely low accuracy, noise (any)
4	low accuracy, high noise
5	low accuracy, noise
6	good accuracy, high noise
7	good accuracy, low noise
8	almost no failures, noise (any)
9	no failures, noise
10	no failures, no noise

7.3. Attribute Selection Comparisons

	10-Scale Output Class	**2-Scale Output Class**

Correlation Attribute Evaluation	Attribute Evaluator (supervised, Class (nominal): 18 Class): Correlation Ranking Filter Ranked attributes: 0.226 10 PeopleNumber 0.2194 16 Referee 0.1973 15 Crowd 0.1764 1 Id 0.1592 8 CamAngle 0.1587 9 CamDistance 0.1467 3 In-Out 0.1327 14 BackgroundColor 0.1308 13 FieldColor 0.128 12 FieldType 0.125 6 CamQuality 0.1234 2 Sport 0.1193 7 CamMovement 0.1094 5 CamType 0.079 17 GroundPosition 0.079 11 HeadProtector 0.06 4 Light Selected attributes: 10,16,15,1,8,9,3,14,13,12,6,2,7,5,17,11,4 : 17	Attribute Evaluator (supervised, Class (nominal): 18 Class): Correlation Ranking Filter Ranked attributes: 0.3242 15 Crowd 0.3203 9 CamDistance 0.3203 3 In-Out 0.2958 7 CamMovement 0.2695 8 CamAngle 0.261 16 Referee 0.2504 14 BackgroundColor 0.244 10 PeopleNumber 0.1896 12 FieldType 0.1644 13 FieldColor 0.1351 1 Id 0.1298 2 Sport 0.0782 6 CamQuality 0.0765 5 CamType 0.0765 4 Light 0.0586 17 GroundPosition 0.0107 11 HeadProtector Selected attributes: 15,9,3,7,8,16,14,10,12,13,1,2,6,5,4,17,11 : 17
Information Gain Attribute Evaluation	Attribute Evaluator (supervised, Class (nominal): 18 Class): Information Gain Ranking Filter Ranked attributes: 1.1366 2 Sport 0.6423 13 FieldColor 0.5992 14 BackgroundColor 0.5789 12 FieldType 0.5647 9 CamDistance 0.4636 1 Id 0.2898 15 Crowd 0.2888 7 CamMovement 0.2521 8 CamAngle 0.2452 16 Referee 0.2417 6 CamQuality 0.1665 3 In-Out 0.072 5 CamType 0.0543 11 HeadProtector 0.0315 17 GroundPosition 0.0255 4 Light 0 10 PeopleNumber Selected attributes: 2,13,14,12,9,1,15,7,8,16,6,3,5,11,17,4,10 : 17	Ranked attributes: 0.343726 1 Id 0.137386 9 CamDistance 0.123167 14 BackgroundColor 0.118877 10 PeopleNumber 0.101851 3 In-Out 0.085085 7 CamMovement 0.08282 15 Crowd 0.077048 16 Referee 0.073872 8 CamAngle 0.058433 2 Sport 0.056315 12 FieldType 0.055622 13 FieldColor 0.027262 5 CamType 0.027262 4 Light 0.023837 17 GroundPosition 0.007999 6 CamQuality 0.000168 11 HeadProtector Selected attributes: 1,9,14,10,3,7,15,16,8,2,12,13,5,4,17,6,11 : 17
Wrapper Subset Evaluation with Naïve Bayes configuration	Selected attributes: 2,3,6,9,10,12,13,14,16 : 9 Sport In-Out CamQuality CamDistance PeopleNumber FieldType FieldColor BackgroundColor Referee	

7.4. Video Manual Ranking Comments

Video Id	Comments
5	Although there are parts that have clear higher success rate, the stats picked are the worst ones (min > 5:16), average results are taken for computation
6	Only the 40 first secs of the video have been used for computation
19	High angles discarded for computation
47	Only measured first 40 seconds of the video. Only mid distance scenes have been evaluated.
51	Medium cam angle as the scenes to be anonymized are taken at this angle. Far scenes are discarded
53	This video cannot be used as input as most of the scenes are from a distance out of range.
57	Only last 8 seconds from the video taken, where the camera is close to them in low angle.
75	Only close range scenes used for accuracy
91	Only lifting scenes taken for voting
	Heads out of the frame won´t be taken into account when determining video result.

References

TEAM, T. B. D. 'Kodak Facebook Collage Project'.

MOHAMMED, S. A. A. (2015) 'RECOGNITION OF FACE DETECTION SYSTEM BASED ON VIDEO'.

Omar, S. H. A. (2016) 'BIOMETRIC SYSTEM BASED ON FACE RECOGNITION SYSTEM'.

Emgu CV: OpenCV in .NET (C#, VB, C++ and more). Available at: http://www.emgu.com/wiki/index.php/Main_Page.

Weka 3 - Data Mining with Open Source Machine Learning Software in Java (no date). Available at: https://www.cs.waikato.ac.nz/ml/weka.