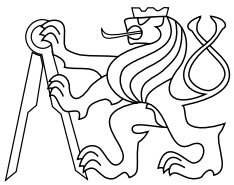
CENTER FOR
MACHINE PERCEPTION



CZECH TECHNICAL
UNIVERSITY IN PRAGUE

BACHELOR THESIS

Gender and age estimation from video

Jan Krček

krcekjan@fel.cvut.cz

BSc Thesis CTU–CMP–2015–03

May 19, 2015

Thesis Advisor: Ing. Vojtěch Franc Ph.D.

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Jan K r č e k

Studijní program: Otevřená informatika (bakalářský)

Obor: Informatika a počítačové vědy

Název tématu: Odhad pohlaví a věku z videa lidské tváře

Pokyny pro vypracování:

Implementujte metodu pro robustní odhad pohlaví a věku z videosekvencí lidské tváře.
Použijte existující knihovnu pro odhad pohlaví a věku ze statických fotografií.
Vytvořte testovací protokol pro měření přesnosti strojového odhadu pohlaví a věku.
Otestujte přesnost Vámi navržené metody a porovnejte ji s existujícím řešením, které se vyvíjí na katedře kybernetiky.

Vedoucí práce dodá knihovny pro odhad pohlaví a věku ze statických fotografií, video sekvence s anotací pohlaví/věku, tracker tváří a nástroje pro učení klasifikátorů z dat.

Seznam odborné literatury:

[1] Ramanathan Narayanan, Chellappa Rama, Biswas Soma: Age progression in Human Faces: A survey. Journal of Visual Languages and Computing. 2009.
[2] Han Hu, Otto Charles, Jain Anil K.: Age Estimation from Face Images: Human vs. Machine Performance. International Conference on Biometrics (ICB). 2013.
[3] Guo Guodong, Mu Guowang: Simultaneous Dimensionality Reduction and Human Age Estimation via Kernel Partial Least Squares Regression. International conference on Computer Vision and Pattern Recognition. 2011.

Vedoucí bakalářské práce: Ing. Vojtěch Franc, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 13. 11. 2014

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

BACHELOR PROJECT ASSIGNMENT

Student: Jan K r č e k

Study programme: Open Informatics

Specialisation: Computer and Information Science

Title of Bachelor Project: Gender and Age Estimation from Video

Guidelines:

Implement a method for robust estimation of gender and age in video sequences of human faces. Use existing library for gender and age estimation from still images.
Create a testing protocol for benchmarking accuracy of gender and age estimators.
Evaluate accuracy of your method and compare it with the solution being developed in the department of Cybernetics.

Thesis advisor supplies a library for gender/age estimation from still images, anotated video sequences and tools for classifier learning from data.

Bibliography/Sources:

[1] Ramanathan Narayanan, Chellappa Rama, Biswas Soma: Age progression in Human Faces: A survey. Journal of Visual Languages and Computing. 2009.
[2] Han Hu, Otto Charles, Jain Anil K.: Age Estimation from Face Images: Human vs. Machine Performance. International Conference on Biometrics (ICB). 2013.
[3] Guo Guodong, Mu Guowang: Simultaneous Dimensionality Reduction and Human Age Estimation via Kernel Partial Least Squares Regression. International conference on Computer Vision and Pattern Recognition. 2011.

Bachelor Project Supervisor: Ing. Vojtěch Franc, Ph.D.

Valid until: the end of the summer semester of academic year 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, November 11, 2014

Anotace Bakalářská práce se zaměřuje na implementaci metody odhadu věku a pohlaví lidských tváří z videa. Využívá k tomu již existující knihovny pro odhad věku a pohlaví ze statických obrázků a sestavuje protokol pro testování přesnosti klasifikátorů věku a pohlaví. Přesnost klasifikátorů je zvýšena použitím lineárního verifikačního klasifikátoru, který z rozhodování vyřazuje problematické obrázky ze vstupní video sekvence. Práce porovnává více způsobů trénování a ladění verifikačních klasifikátorů. Experimenty ukazují, že použitím verifikačního klasifikátoru lze významně redukovat chybovost odhadu pohlaví.

Klíčová slova počítačové vidění, rozpoznávání tváří z videa, odhad věku a pohlaví

Abstract This bachelor project focuses on implementing a method for prediction of gender and age of human faces in video sequences. The project builds on existing libraries for gender and age prediction from still images. In the thesis we created testing protocol for benchmarking accuracy of gender and age predictors working with video. Predictor accuracy is enhanced by usage of linear verification classifiers used to filter out difficult or corrupted frames from the input video sequence. The work compares multiple verification classifiers and different approaches to training and fine tuning their parameters. The experiments show that using the verification classifier can significantly reduce the gender prediction error.

Keywords computer vision, face recognition from video, age and gender prediction

Acknowledgment I wish to express my sicere thanks to Ing. Vojtěch Franc Ph.D., my thesis advisor. I am extremly thankful and indebted to him for sharing expertise, and sincere and valuable guidance and encouragement extended to me.

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl vešekeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Dne ....

Contents

1 Intro

This work is documentation of an IT experimental project from the computer vision and recognition fields. It is highly focused on experimentation and comparing results of those experiments. In this work I will try to describe the task of creating a classifier that allows more accurate age and gender prediction from video sequences.

A common approach used for prediction from video sequences processes each frame separately and aggregates the individual per-frame predictions to a final decision. Predicting age and gender of person from video frames is harder than doing so from still image. Reasons for that are motion blur, lower resolution of input images, people not looking in the camera and other disturbing effects. Trying to deal with each of those issues separately would be too complicated and might not yield better results at all. The approach taken in this work centers around training a verification classifier used to check quality of each frame in the input video sequence. The verification classifier is used to select from the video sequence only those frames quality of which is good for reliable prediction of age or gender. The other frames from the sequence are discarded an not used to make the final decision.

The content of this work is structured into the four main parts:

1. (Section 2). Designing benchmark for evaluation of systems predicting age and gender from videos. The input was a database of videos downloaded from the Internet which was provided to me by the thesis supervisor. My work involved designing a tool for annotation of video tracks and designing a suitable evaluation protocols.

2. (Section 3). Design of a verification classifier which is trained from examples by the Support Vector Machines algorithm [3].

3. (Section 4). Design of several strategies for prediction of age and gender from video sequences. We describe both baseline methods as well as new strategies which make use of the verification classifier.

4. (Section 5). Experimental evaluation of the proposed prediction strategies on the developed benchmark and comparison to the baseline methods.

Programming language used for all the scripting and experiments was Matlab. The annotated database of video sequences, a library for prediction of age and gender on static images, a toolbox for training a linear SVM classifier were provided for purposes of this work by my thesis advisor.

2 Benchmark

2.1 Source database

The input database is composed from video sequences downloaded from the Youtube. Each sequence contains a single person speaking to a webcamera. The videos are all medium to low resolution and of different lenghts but on average they are several minutes long. The database comes with annotation of age, gender and other attributes described below in details.

For purposes of this work it was necessary to split the database into training and testing part. To get reliable estimate of the prediction errors the training and testing sets were generated randomly in three different versions (splits). We make sure that the same identity does not simultaneously appear in the training and the testing set. This was achieved by using very a convenient field in the original annotations called foldID. Thus splits used in this project respect the original assignment of the videos to folders. Example of a frame from a video sequence in this database is shown in Figure 1. Age distribution is shown in Figure 2.

Figure 1: An example frame extracted from one of the video sequences.

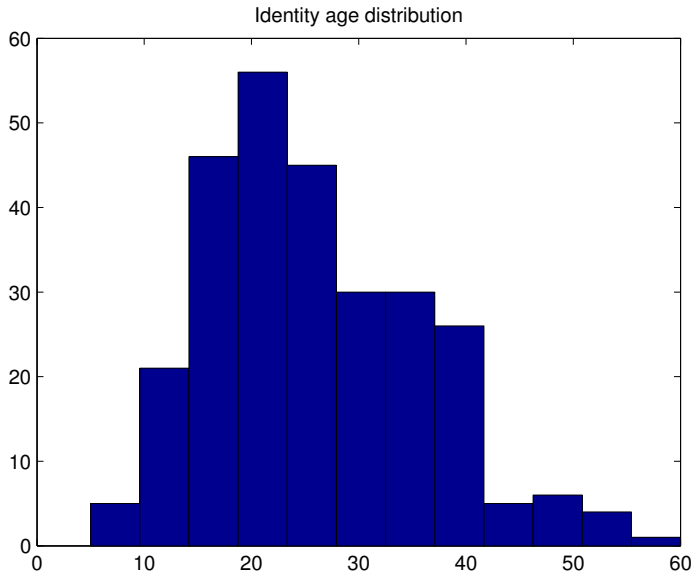


Figure 2: Age distribution among all data.

2.2 Finding tracks

Each video sequence from the input database was processed by a face tracker and a facial landmark detector [1]. The result was a set of face tracks. Typically, several face tracks were found in a single video but each containing the same identity. In the sequel, we will use the term "**face track**" to denote a sequence of image frames of fixed size each depicting a single face. The faces were cropped from the video sequence and consequently aligned by an affine transform based on the position found by the tracker and the facial landmark detector. Due to a complicated background clutter and non-static subjects some face tracks do not contain faces or the tracks are extremly short. These false or short tracks were manually removed from the benchmark.

Finally, each face track in the benchmark was annotated by a set of attributes which were extracted from the input database:

1. aviFile - Contains the name of the videofile with this track.
2. srcTrackFile - Contains name of the file with facebox coordinates.
3. personID - Contains number representing identity of this person.
4. gender - Contains M or F for Male or Female.

[1] Provided by courtesy of Eyedea Recognition s.r.o. `www.eyedea.cz`

5. age - Contains age rounded to multiple of five.

6. fold - Contains number of the folder this track belongs to.

2.3 Definition of splits

Every track is assigned to a folder from 1 to 3. The folder definition originates from the input database. In order to generate different splits into training and testing parts, we use a permutation of folders as described in Table 1.

Split	Trn	Tst
1	fold1+fold2	fold3
2	fold1+fold3	fold2
3	fold2+fold3	fold1

Table 1: The assignment of folders to splits. Each face track with the given folder number goes to the corresponding training or testing split.

Training parts have roughly twice as many tracks as the testing parts. The age and gender distributions for each training split is shown in Table 2, Figure 3 and Figure 4. The same statistics for the testing splits are shown in Table 3, Figure 5 and Figure 6. In generally it holds that in all three splits there are more male than female tracks. The splits are based on number of identities rather than number of tracks in them. In that regard they are more or less balanced.

Tracks	Identities	Male	Female	Tracks
All	275	174	101	676
Split1	176	108	68	405
Split2	175	114	61	486
Split3	175	114	61	415

Table 2: Identity, age and gender distribution among training splits.

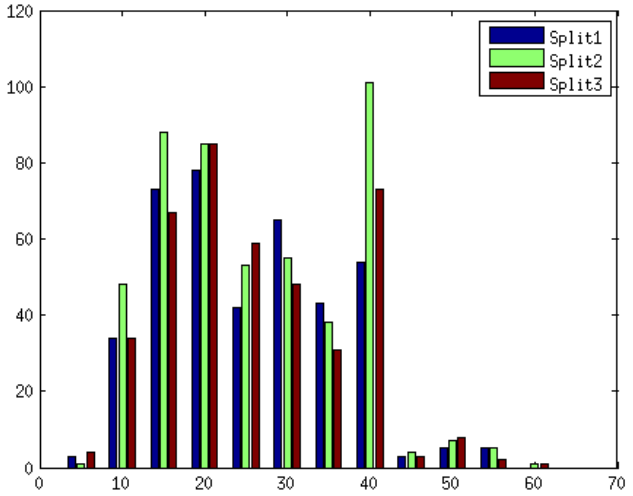


Figure 3: Age distribution among training splits.

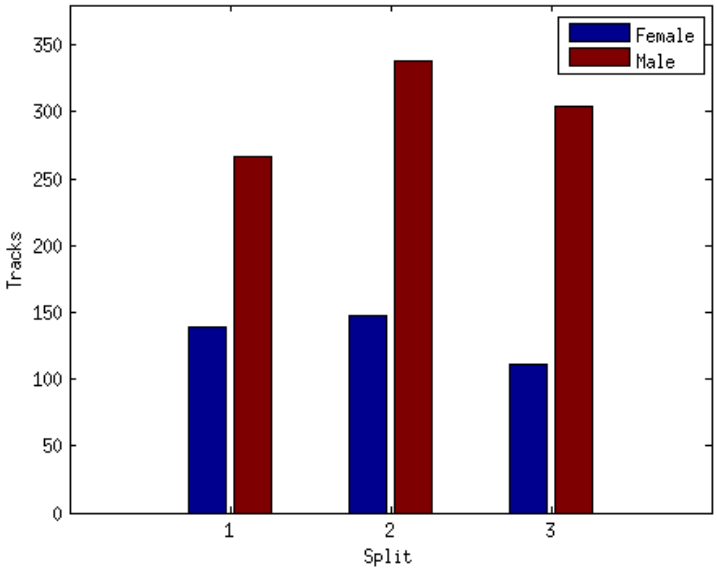


Figure 4: Gender distribution among training splits.

Tracks	Identities	Male	Female	Tracks
All	275	174	101	676
Split1	87	60	27	248
Split2	88	54	34	167
Split3	88	54	34	238
Unused	12	-	-	23

Table 3: Identity, age and gender distribution among test splits.

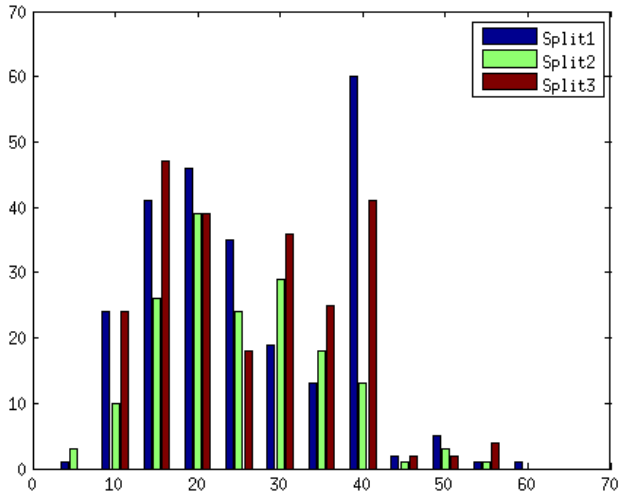


Figure 5: Age distribution among test splits.

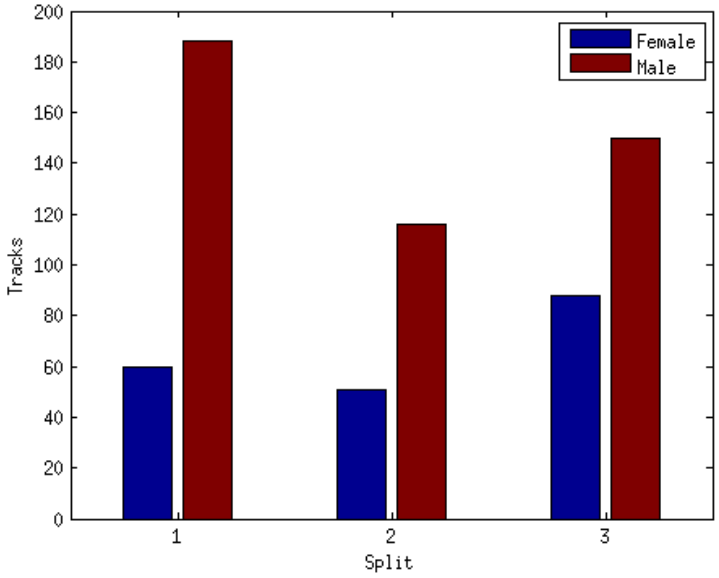


Figure 6: Gender distribution among test splits.

2.4 Benchmark protocols

The training and testing splits of the face tracks were further organized to three different scenarios. The scenarios should correspond to possible applications of the age/gender prediction systems. In particular, we designed the following scenarios:

Scenario 1. All found tracks are used in the benchmark. This scenario simulates the situation when age and gender needs to be predicted from many shorter tracks some of them containing the same identity. The tracks are around 5 to 20 seconds long. This was the default scenario used in most of the experiments.

Scenario 2. Only first track of each video sequence is used. This simulates situations like live demonstrations or surveillance systems which require fast predictions. For example, the prediction needs to be made in one or two second since the subject appeared in the scene.

Scenario 3. All tracks for each video sequence are merged into one long record. In this scenario the classifier is intended to be used on longer sequences with unrestricted processing time. For example, this might be useful for offline data analysis when the main goal is a reliable prediction not the decision time.

The distribution of identities, age and gender in terms of the number of tracks is shown in Table 4.

Scenario I

Tracks	Identities	Male	Female	num of trn tracks	num of tst tracks
All	275	174	101	676	-
split 1	87	60	27	405	248
split 2	88	54	34	486	167
split 3	88	54	34	415	238

Scenario II

Tracks	Identities	Male	Female	num of trn tracks	num of tst tracks
All	275	174	101	275	-
split 1	87	60	27	176	87
split 2	88	54	34	175	88
split 3	88	54	34	175	88

Scenario III

Tracks	Identities	Male	Female	num of trn tracks	num of tst tracks
All	275	174	101	275	-
split 1	87	60	27	176	87
split 2	88	54	34	175	88
split 3	88	54	34	175	88

Table 4: The table shows a distribution of identities, age and gender in each scenario and its corresponding training/testing splits.

3 Verification classifier

3.1 The idea

The ultimate goal of this work is to design a predictor of age and gender working on top of face tracks. The track predictor is composed from age/gender predictors working on static images. The question is how to aggregate predictions on individual frames to a single prediction assigned to the whole track. In Section 4, we will design several aggregation strategies which use a verification classifier the purpose of which is to filter out corrupted or difficult frames from the decision process. This section describes a method used to train the verification classifier from examples. The idea is illustrated in Figure 7.

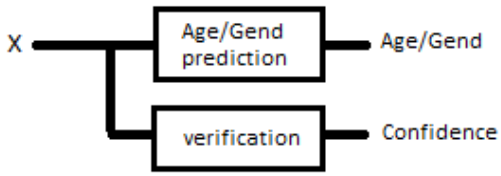


Figure 7: The x are features computed on a static-frame. The verification classifier augments the age/gender prediction by a confidence.

3.2 Verification classifier

The verification classifier is a function which for an input frame returns either a confidence value (real valued score) or a sharp decision (valid /non-invalid). We implement the verification classifier by a linear decision rule $V : \mathbb{R}^n \to \{-1, +1\}$ defined as follows:

$$V(x) = \begin{cases} +1, & v(x) \geq \psi \\ -1, & v(x) < \psi \end{cases}$$

where $v(x) = \langle w, x \rangle + w_0$ is a linear score the value of which serves as the confidence. The parameters w, w_0 are trained from example of classifier inputs and desired labels $\{(x_1, y_1), ..., (x_m, y_m)\} \in (\Re^n \times \{+1, -1\})$ by SVM algorithm[3]. The SVM algorithm has a single parameter to be tuned, so called reguralization constant C. The value of C is tuned on validation data as will be described below.

The verification classifier is trained for a given static frame predictor, in our case either predictor of gender or age, based on training examples which

capture the inputs where the static-frame predictor works or does not work. More details on how to construct the training set are given in section 3.4.

The input of the verification classifier are features extracted from the frame of the face track. Recall, that the frames are affinely aligned facial images. We experimented with the following two feature descriptors:

1. Pyramid of Local Binary Patterns (LBPPYR) proposed in [2]. The descriptor is a high-dimensional binary sparse vector the value of which are LPB codes computed in several resolutions of the input frame and stack to a single vector.

2. Histograms of Locally Binary Patterns. The image is split to windows. A histogram of LBP codes in each window is computed. The histograms of all windows stacked to a single vector form the final feature descriptor.

3.3 Age/Gender predictor from static images

For prediction of age and gender form static frame a Matlab library LIBFACE provided by the thesis advisor was used. The gender classifier outputs a real-value score the sign of which defines the predicted class (+1 male, -1 female). The age predictor returns a sharp estimate of the age. These static-frame predictors do not provide a reliable estimate of the confidence of their prediction. The output of the verification classifier described above can be seen as a substitute for the missing confidence of the prediction.

3.4 Training the verification classifier

The training set contains examples on which the static-frame predictors work (label +1) and do not work (label -1). We trained two types of the verification classifier:

1. Combined classifier trained on age and gender examples. In this case, the positive examples are those where both the gender and the age predictor work correctly and negative where at least one of them fails. In particular, the frame is assigned to be positive when the gender prediction corresponds to the ground truth gender and simultaneously the age prediction is within a defined range ($|Age_{prediction} - Age_{annotation}|$ is below a threshold).

2. Specialized classifier trained only on gender examples. In this case, the positive examples are those where the gender predictor estimates the frame correctly and negative otherwise.

We used the worst $N_{max}/2$ and the best $N_{max}/2$ examples of the positive and the negative class. The worst and the best are determined based on the linear score of the gender predictor. The generated training sets were always balanced so that they contained the same amount of positive and negative examples. Training itself was done using Matlab SVMOCAS toolbox [1].

There were several other parameters involved in creating the training set for the SVM algorithm. The number of examples found for negative and positive classes greatly varies depending on values of these constants:

1. N_{max} - Represents the threshold on the number of examples used in the whole training set.

2. $A_{threshold}$ - Represents the maximum allowed difference of predicted and the ground truth age to consider the frame correctly classified.

3. F_{max} - Represents the threshold on the maximal number of frames taken from each track. Raising this value allows creation of bigger training sets at the cost of lesser variability. On smaller sets lower values usually yield better results.

4. F_{tested} - Represents the maximal number of frames from beginning of the track that are used. Allows for limiting the amount of seconds from the start of the face track which are used for training.

The procedure used to create the examples for training the verification classifier was as follows:

1. The values for constants $N_{max}, A_{threshold}, F_{max}$ and F_{tested} were chosen.

2. The negative examples are found by going through all frames and checking which have bigger age error than $A_{threshold}$, or have incorrectly predicted gender.

3. For each frame chosen in last step, a check is done if no more than F_{max} frames from that track are already slected.

4. Positive examples are found the same way, only selecting frames without any errors.

5. $N_{max}/2$ negative examples and $N_{max}/2$ positive examples together form the whole training set.

4 Age and gender estimation from video sequences

In this section we define several predictors of age and gender based on face tracks. That is, the input is a sequence of static-frame predictions along with the output of verification classifier. The face-track gender predictor is a function

$$H_G((g_1, K_1), ..., (g_T, K_T)) \rightarrow \begin{cases} \text{Male} \\ \text{Female} \\ \text{Don't know}^2 \end{cases}$$

where g_t, $t \in \{1...T\}$ are the outputs of the static-frame gender predictor on T frames and K_t, $\in \{1...T\}$ are the confidence values estimated by the verification classifier. The face-track age predictor is a function

$$H_A(a_1, K_1), ..., (a_T, K_T)) \rightarrow \begin{cases} \text{Age form } \{1, 2, \ldots, A\} \\ \text{Don't know}^2 \end{cases}$$

where a_t, $t \in \{1, \ldots, T\}$ are responses of the static-frame age predictor.

4.1 Baseline 1

The face-track gender predictor decides based on the sign of the average (linear) score returned by the static-frame gender predictor:

$$H_{gB1}((g_1, K_1), ..., (g_T, K_T)) = \text{sign}\Big(\frac{1}{T}\sum_{t=1}^{T} g_t\Big)$$

The face-track age predictor outputs the average of the static-frame age predictions

$$H_{aB1}((a_1, K_1), ..., (a_T, K_T)) = \text{sign}\Big(\frac{1}{T}\sum_{t=1}^{T} a_t\Big)$$

4.2 Baseline 2

The second baseline method is defined only for the gender prediction which was the main focus of the thesis. It outputs the gender (male or female) which occurs most frequently in the sequence of static-frame gender prediction:

$$H_{gB2}((g_1, K_1), ..., (g_T, K_T)) = \begin{cases} \text{male} & \text{if} \quad \sum_{t=1}^{T} \delta(g_t \geq 0) \geq \sum_{t=1}^{T} \delta(g_t < 0) \\ \text{female} & \text{if} \quad \sum_{t=1}^{T} \delta(g_t \geq 0) < \sum_{t=1}^{T} \delta(g_t < 0) \end{cases}$$

[2]Option specific to Strategy 2 in Section 4.3

where $\delta(A)$ is 1 if the statement A holds and 0 otherwise.

4.3 Face-track predictors using the verification classifier

Strategy 1 If more than θ frames are classified as correct by the verification classifier then the face-track gender predictor H_1 outputs the signed of the average of the gender scores of the static-frame gender predictor computed over the correct frames. If the number of correct frames is less then θ, then the H_1 outputs the average computed over θ frames with the highest score of the verification classifier.

Strategy 2 The strategy predicts the gender only if the number of correct frames estimated by the verification classifier is at least θ. In this case, the strategy returns the sign of the average over the gender scores of the static-frame predictor computed over the correct frames. If the number of the correct frames is less then θ it returns "don't known" decisions.

Strategy 3 This strategy returns the sign of the average of the gender scores of θ frames with the highest score of the verification classifier.

All of the strategies depend on the constant θ. This constant represents a threshold of the minimal number of correct frames used for decision. The strategies are defined only for the gender prediction. The face-track age predictors are defined analogically when the average is taken over the static-frame age predictions.

5 Experiments

In the first three experiments the main aim was to create combined classifier for both age and gender error minimization. This proved to be not very efficient approach. Reduction of gender misclassifications has shown much better results than age mean absolute error(MAE). Thus in the later experiments classifiers were trained only on gender examples.

5.1 Definition of evaluation statistics

MAE - Mean absolute error is the average over absolute deviations between the predicted and the ground truth age.

FaM - Female as male ratio is the ratio of the female tracks predicted to be male tracks.

MaF - Male as female ratio is the ratio of the male tracks predicted to be female tracks.

DR - Decision rate is the ratio of the tracks for which the decision is made. That is, $1 - DR$ is the ration of tracks where the strategy (namely, the strategy 2) returns "don't know" decision.

CLS - Classification error.

The goal of the age predictor is to minimize the MAE. The goal of the gender predictor is to minimize the maximum of MaF and FaM which is the upper bound of the classification error invariant to the a priory probability.

5.2 Evaluation protocol

In the experiments we used the training/testing splits of the Scenario 1 defined in Section 2. The face tracks in the training split were used to create a training and a validation set necessary to train and tune parameters of the verification classifier. Once the verification classifier was trained it was evaluated on test split. In particular, the evaluation procedure had the following steps:

1. Selecting values for the constants defined in Section 3.4 which control the positive and negative examples selected for training the verification classifier.

2. Splitting the frames in the training tracks into validation part and training part. We used 20% of identities in training examples for tuning the parameters like the SVM regularization constant C and adjusting the decision threshold w_0. Remaining 80% were used for training the weights w of the verification classifier.

3. Cross-validation based selection of the C and the decision threshold w_0 on the training the weights w.

4. Training the final classifier with best combination of C and w_0 on the a set obtained by merging the validation and training data.

5. Evaluation of the performance of the face-track classifier on the testing part.

The procedure has been repeated for the 3 splits. The resulting errors are averages and the standard deviations computed over the 3 splits.

5.3 Tuning the SVM based verification classifier

Tuning was done separately for two constants C and w_0. Both were tuned to reach minimal $\max(FaM, MaF)$ on validation data. The optimal values were selected from a fixed set of candidates of C and w_0. We also considered tuning a set of different values of C for each of the classes separately. So that $C_{negative}$ and $C_{positive}$ represent weight of those examples while training the classifier. Unfortunately, this method turned out to be excessively slow especially with large training datasets and the high-dimensional LBPPYR feature vector. For this reason we used the same C for both classes.

Table 5 summarizes the classification accuracy of the verification classifier trained on example sets of different sizes and using different feature descriptors (LBPPYR and histogram of LBPs). The best verification classifier correctly predicts around 77% of frames. It is seen that the LBPPYR features significantly outperforms the histograms of LBPs.

Classif	trnCls	valFP	valFN	valCls	tstFP	tstFN	tstCls
8k	0.1419	0.3529	0.4766	0.4148	0.3887	0.3600	0.3744
10k	0.0000	0.2610	0.0632	0.1709	0.3083	0.4093	0.3588
20k	0.0000	0.1993	0.2641	0.2317	0.1861	0.5054	0.3458
30k	0.0000	0.0846	0.1102	0.0974	0.1648	0.5208	0.3428
50k	0.0000	0.1781	0.1525	0.1653	0.1687	0.4991	0.3339
h30k	0.0404	0.1980	0.2721	0.2350	0.4391	0.3415	0.3903
h50k	0.0569	0.2579	0.2334	0.2456	0.5027	0.2678	0.3853

Table 5: Classification accuracy of the verification classifier trained on different training sets. The number in the first column denotes the total number of training examples. The letter "h" denotes the results using the histogram of LBP features. The other results (without "h") use the LBPPYR features.

5.4 Results

Table 6 summarizes the test performance of different face-track prediction strategies. The results are show for different version of the verification classifier as described in the previous section.

—	MAE	FaM	MaF	Decision rate
Baseline 1	7.6969/0.34	0.0061	0.1499	100%
Baseline 2	—	0.0061	0.1526	100%
Strategy 1 (8k)	7.4725/0.35	0.0075	0.1741	100%
Strategy 1 (16k)	7.5021/0.7	0.0075	0.1599	100%
Strategy 2 (8k)	6.9340/0.35	0.0218	0.0717	49.22%
Strategy 2 (16k)	7.2896/0.6	0.0080	0.1459	93.18%
Strategy 3 (8k)	7.4522/0.35	0.0096	0.1437	100%
Strategy 3 (16k)	7.3331/0.65	0.0080	0.1442	100%
Strategy 2 (10k/gender)	—	0.0023	0.0182	60.64%
Strategy 2 (20k/gender)	—	0.0113	0.0554	43.32%
Strategy 2 (30k/gender)	—	0.0141	0.0327	49.46%
Strategy 2 (50k/gender)	—	0.0087	0.0418	48.67%

Table 6: Results of running recognition on the test splits.

The best results were achieved using the prediction strategy 2 with the gender verification classifier trained on 10K training examples and using the LBPPYR features. The best configuration of constants (from 3.4 and 4.3) used to train this classifier were $N_{max} = 10000, F_{max} = 30, F_{tested} = 150$ and $\theta = 10$. Best values for C and w_0 were 1 and 0. In particular, the

best face-track gender predictor achieves the error $\max(FaM, MaF) = 1.8\%$ when classifying 60.6% of the test face tracks while on the remaining 39.4% returns the "don't known" decision. This results constitutes a significant error reduction if compared to he baseline methods 1 and 2 both achieving the gender error around $\max(FaM, MaF) = 15\%$. On the other hand, it is seen that the none of the decision strategies utilizing the verification classifier is able to significantly reduce the MAE.

The gender prediction strategy 2 allows to control the trade-off between the prediction accuracy and the decision rate by varying the parameter θ (see the description in Section 4.3). The effect of tuning the parameter θ is shown in Figure 8 which displays the gender error $\max(FaM, MaF)$ as a function of the decision rate. It is seen that the prediction error grows with the increasing decision rate, however, it remains significantly below the prediction error of the other competing strategies.

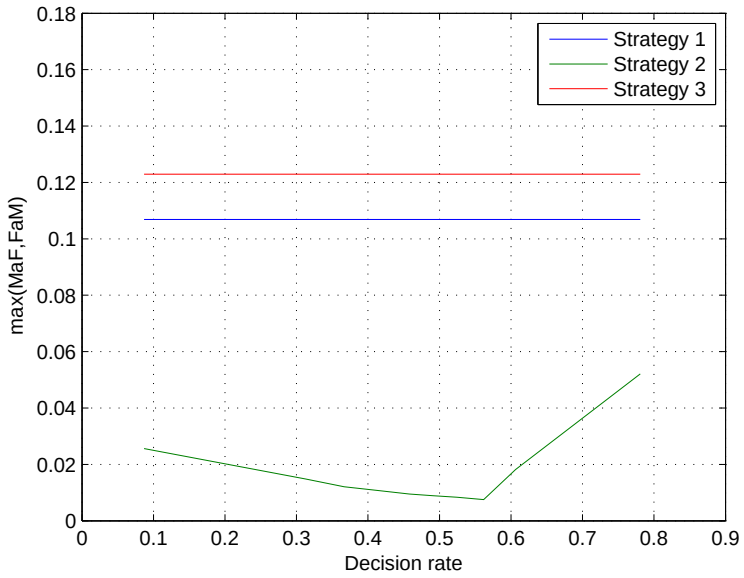


Figure 8: max{FaM,MaF} values for different strategies.

On Figure 9 we can see the MAE results for the combined classifiers compared with those of an ideal classifier using the prediction strategy 2. The ideal classifier represents the best possible results of this experiment. It shows that there is quite a lot of space for improvement.

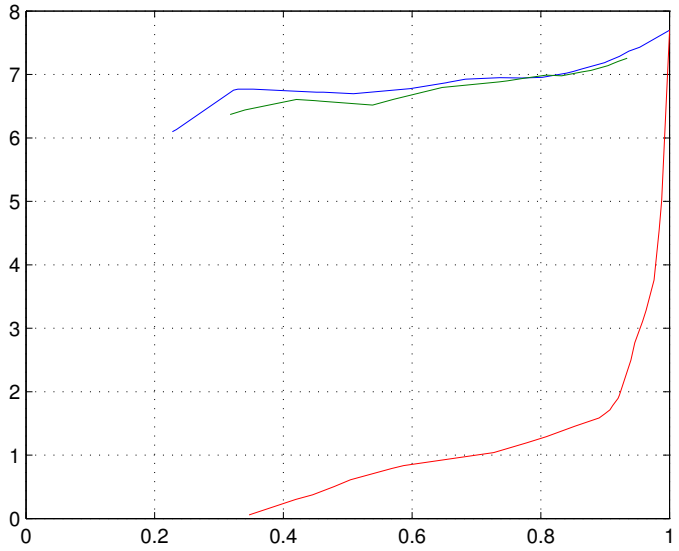


Figure 9: Number of tracks used vs. MAE for 8k(Blue), 16k(Green) and ideal(Red) classifiers.

On the next page there are two more detailed figures of the specific gender errors. Figure 10 and Figure 11 compare results of the three developed strategies and display respectively MaF and FaM as a functions of the minimum correct frames threshold θ.

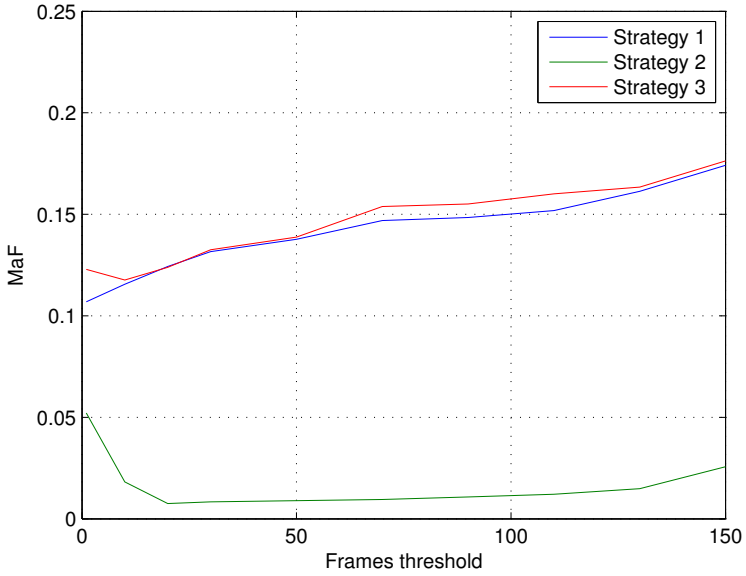


Figure 10: MaF values for the 10kGender classifier.

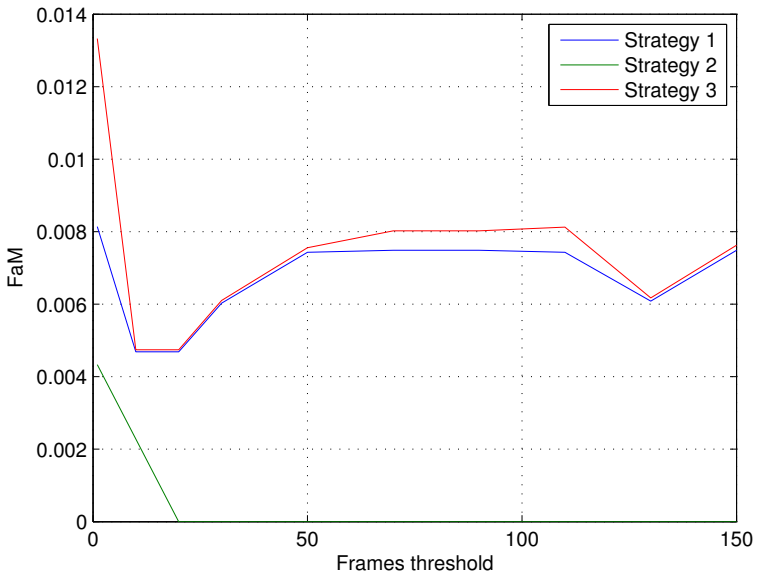


Figure 11: FaM values for the 10kGender classifier.

6 Conclusions

In this work we created a new benchmark for evaluation of methods for age and gender prediction from video sequences. The benchmark has been created from a set of videos downloaded from the Youtube. The benchmark defines a set of face tracks and their splits to training and testing part. All face tracks are annotated with the age and the gender. The splits are further organized to three different scenarios simulating different deployment of the age/gender prediction systems in the practice.

The second goal of the thesis was to develop a new age/gender prediction strategies working on top of the face tracks. The standard methods are based on predicting age/gender from each frame of the face track independently and making the final decision by averaging the individual predictions. We improved these method by training a verification classifier which for each frame in the track predicts whether the static-frame age/gender predictor will work or not. The frames identified by the verification classifier to be difficult are then removed from the final decision. We tested several variants of the decision strategies using the verification classifier. The best turned out to be a decision strategy which rejects prediction of very difficult face tracks when the trade-off between the prediction error and the decision rate is tuned by a single parameter.

We used the developed benchmark to evaluate the new decision strategies and to compare them with several baseline methods. The results show that utilizing the verification classifier can significantly reduce the gender prediction error at the cost of not classifying difficult face tracks. On the other hand, using the same strategy for the age prediction does not yield a significant improvement. The reason is probably the used training set of positive and negative examples capturing simultaneously behavior of the age and the gender predictor. It seems that training the verification classifier independently for both task (age and gender prediction) yields better results. Due to lack of time, we have not managed to verify this hypothesis experimentally. That is, we implemented this strategy only for the gender prediction and left the age prediction for the future work.

The other finding is that the verification classifier works significantly better with the high dimensional LBPPYR feature descriptor rather than with the low dimensional histogram of LBPs. Interestingly, the same LBPPYR features are used as the input of the static-frame age/gender predictors.

References

[1] V. Franc and S.S. Sonnenburg. Optimized cutting plane algorithm for support vector machines. In Andrew McCallum and Sam Roweis, editors, *Proceedings of the 25th Annual International Conference on Machine Learning (ICML 2008)*, pages 320–327, New York, USA, July 2008. ACM. electronic.

[2] Sören Sonnenburg and Vojtěch Franc. Coffin: A computational framework for linear svms. In *Proceedings of the 27th Annual International Conference on Machine Learning (ICML 2010)*, pages 999–1006, Madison, WI 53704 USA, June 2010. Omnipress.

[3] Vladimir N. Vapnik. *Statistical learning theory*. Adaptive and Learning Systems. Wiley, New York, New York, USA, 1998.