Improved Weak Labels using Contextual Cues for Person Identification in Videos

Makarand Tapaswi Martin Bäuml Rainer Stiefelhagen
Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
{makarand.tapaswi, baeuml, rainer.stiefelhagen}@kit.edu

***Abstract*— Fully automatic person identification in TV series has been achieved by obtaining weak labels from subtitles and transcripts [11]. In this paper, we revisit the problem of matching subtitles with face tracks to obtain more assignments and more accurate weak labels. We perform a detailed analysis of the state-of-the-art showing the types of errors during the assignment and providing insights into their cause. We then propose to model the problem of assigning names to face tracks as a joint optimization problem. Using negative constraints between co-occurring pairs of tracks and positive constraints from *track threads*, we are able to significantly improve the speaker assignment performance. This directly influences the identification performance on all face tracks. We also propose a new feature to determine whether a tracked face is speaking and show further improvements in performance while being computationally more efficient.**

I. Introduction

Person identification in TV series (*e.g.*, [2], [3], [8], [9], [11], [13], [15], [17], [18]) enables their indexing and opens a broad area of applications including character retrieval, video recommendation or story summarization.

In their seminal work [11], Everingham *et al*. propose to leverage subtitles and (fan) transcripts to automatically obtain training data for character-specific face models. Subtitles contain the dialogs associated with a timespan, while transcripts provide the speaker name for these dialogs. Matching the dialogs and aligning the two texts provides information about *who* speaks *when*. Lip-motion analysis is performed to associate the speaker identity with the speaking face track in case of multiple co-occurring face tracks.

In a typical TV series, this method is able to assign labels to roughly 20-30% of all face tracks with a precision of 80-90%. Thus, we refer to them as *weak* labels. This is shown to be sufficient for training classifiers to subsequently identify all face tracks, especially when the employed classifier is robust against erroneous training data (*e.g.*, SVMs). Hence, fully automatic character identification can be performed without any manual annotation. Please refer to [11] for a more gradual introduction to the problem.

In this paper, we re-consider the problem of assigning weak labels to tracks. Our contributions are the following. First, in Sec. III we present a detailed analysis of the errors in a recent implementation and application of the approach by [2]. Motivated by the types of errors revealed in this analysis, we propose a novel way to assign speaker identities to face tracks, incorporating both positive and negative constraints between tracks (see Sec. IV). Inspired by a recent application to face track clustering [19], our negative constraints (*not* same label) are obtained from co-occurring tracks and positive constraints (same label) from tracks which are part of a *shot thread*. Our proposed model already improves assignment and identification performance. In addition, we also propose a simplified and improved feature to measure the amount of lip movement. Sec. V presents a detailed experimental evaluation of our proposed approach and the influence of improved weak labels on subsequent face track identification in TV series.

II. Background and Related Work

We present the related work for this paper into two areas: (i) person identification using subtitles and transcripts; and (ii) analysis of lip motion.

A. Textual cues for person identification in videos

Due to some ambiguity in the terms, we first define subtitles and transcripts as used within the scope of this paper.

Subtitles A text which contains dialogs from the video along with the corresponding timestamps qualifies as subtitles. They are also often called *closed captions*. Note that subtitles are similar to the output of a near-perfect automatic speech recognition system which produces the spoken text and timestamps, but not speaker identities.

Transcripts A text which (minimally) contains the character names and their dialog is a transcript. Note that in most cases the real scripts[1] provided to the actors are hard to obtain (*e.g.*, due to copyright restrictions). However, fans often provide various transcriptions of the dialogs on the internet[2]. While the original scripts contain extra information about the scene setting (INT/EXT - interior/exterior, time and day, location, actions, *etc*.), fan transcripts often do not.We restrict ourselves to the information necessary for person identification, namely character names and dialogs.

As briefly discussed in the introduction, [11] forms the basis of most related work in this area. Dialogs from the subtitles and transcripts are matched via dynamic programming. The matching generates pairs {speaker identity, timestamp}. To resolve ambiguities between co-occurring face tracks, a score is computed over the duration of the subtitle to capture the strength of the lip movement (see Sec. IV-A). The highest

[1] https://sites.google.com/site/tvwriting/
[2] *E.g.*, http://bigbangtrans.wordpress.com/

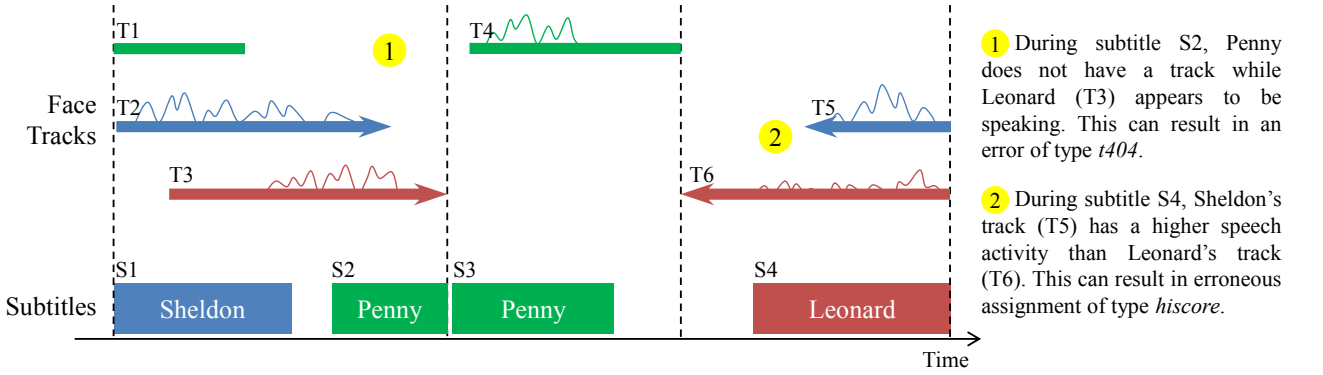

Fig. 1. Overview and error sources in the speaking face assignment of [11]. Subtitles, their associated names and timespan are shown at the bottom of the timeline, while face tracks are shown on the top. Each character is represented with a unique color. The lip activity associated with a face track is shown above each track. The track threading is indicated by arrow heads for the track edges. We highlight the two scenarios of errors and describe them in the neighboring text. This figure is best viewed in color.

scoring face track – if the score further passes a threshold – is assigned the name obtained from the subtitle-transcript alignment. Finally, all other tracks (which are not assigned a speaker identity) are assigned an identity by determining the nearest neighbor to the speaker-assigned tracks.

Since [11], the area of person identification has branched into two parallel streams. One stream follows [11] and performs weak label assignment followed by a distinct subsequent identification stage [2], [17]. The focus of this paper is to improve the weak label assignment for such a scheme. The other stream does not explicitly split the weak labeling and person-specific model training into two parts and uses techniques such as multiple instance learning [13] or ambiguous label learning [8]. Both streams have their advantages and disadvantages, however the focus of this work is on improving the former. We thus compare our weak label assignment performance against [11] and identification against [2] (shown to be better than [11]).

Among the other works in this area, [9] does not use transcripts; and [18] uses manually labeled training data, and thus are not directly comparable. Recently, there is rising interest in jointly solving vision tasks. *E.g.* [3] uses transcripts and performs person and action identification jointly, while [15] proposes to solve person identification in vision along with co-reference resolution in the transcript. However [15] does not use the dialog matching scheme and thus obtains fairly bad identification performance.

In all of the above methods the actual speaking face assignment is either not addressed specifically or performed in a very simple manner (compute score and assign name to max-score co-occurring track), and the focus lies rather on improving the subsequent identification. In this paper, we explicitly address the improvement of the speaker assignment (and thus the quality of the generated weak labels) in the presence of both transcripts and subtitles. We motivate the need to revisit this problem in Sec. III.

B. Lip motion analysis

In the speech recognition literature, lip movement analysis has long been investigated as a means to localize audio sources and improve speech recognition (*e.g.*, [5], [10]), for example motivated by the McGurk effect [14]. Different features have been employed to describe the mouth region, from simple gray level pixels (possibly after a 2d-DCT), motion-based analysis, to detailed models of the lips such as Active Appearance Models. These features are then combined with audio features, typically Mel Frequency Cepstral Coefficients, and are used to train a Hidden Markov Model (*e.g.*, [1], [12]). In a different approach, [10] determine the mutual information between the audio and video signal and localize pixels that match the audio signal.

However, to the best of our knowledge the problem of improving the speaker assignment by taking into account contextual cues such as uniqueness constraints and track threads has not been addressed.

III. Motivation

In this section, we analyze the performance and possible error sources of the character name assignment to face tracks, using the implementation of Everingham's method by [2]. Together with the experiments from Sec. V, we will show that there are many missed opportunities in terms of the assignment and needless errors that can be corrected.

A. Evaluation metrics

Consider a set of N tracks $\mathcal{T} = \{(t_i, s_i, y_i)\}, i = 1 \dots N$. Every track t_i is associated with a speaking score s_i and a ground truth identity y_i (which also includes an "unknown" class for unnamed guest characters). In [2], [11], s_i is computed for every pair of frames of the face track and is the minimum pixel-wise distance between the mouth regions. The speaker identity of each subtitle is assigned to the simultaneously occurring face track with the maximum speaker score. In this way, weak labels are assigned to M face tracks with identities $\hat{y}_k$, $k = 1 \dots M$.

The performance of the weak labeling can be analyzed via two metrics:

$$\text{spk precision} = \frac{1}{M} \sum_k \mathbb{1}\{y_k = \hat{y}_k\} \quad (1)$$

$$\text{spk assigned} = \frac{M}{N} \quad (2)$$

TABLE I

EPISODE STATISTICS, ANALYSIS OF THE SPEAKING FACE ASSIGNMENT AND SUBSEQUENT IDENTIFICATION PERFORMANCE OF OUR IMPLEMENTATION OF [2]. BBT AND BF ARE ACRONYMS FOR THE TWO TV SERIES WE ANALYZE. PLEASE REFER TO SEC. III FOR A DESCRIPTION.

	BBT-1	BBT-2	BBT-3	BBT-4	BBT-5	BBT-6	Total	BF-1	BF-2	BF-3	BF-4	BF-5	BF-6	Total
#tracks	657	615	660	613	524	851	3920	796	1004	1194	900	840	1127	5861
#subtitle lines	620	542	560	521	495	503	3241	683	858	823	786	700	758	4608
#assigned	147	104	132	126	78	116	703	158	192	177	186	174	211	1098
#characters	7	6	8	9	7	7	12	12	13	14	15	14	18	27
spk precision	89.1	87.5	93.2	88.9	92.3	87.9	89.8	89.2	82.8	81.4	87.1	88.5	86.7	85.9
spk assigned	22.4	16.9	20.0	20.6	14.9	13.6	17.9	19.8	19.1	14.8	20.7	20.7	18.7	18.7
err total	16	13	9	14	6	14	72	17	33	33	24	20	28	155
err *t404*	2	2	3	1	3	2	13	5	10	11	7	7	6	46
err *hiscore*	14	11	6	13	3	12	59	12	23	22	17	13	22	109
err *uniq*	5	1	1	4	0	3	14	1	5	2	4	2	0	14
err *thread*	2	0	1	0	1	2	6	5	2	4	4	2	4	21
id accuracy	91.0	91.1	75.0	80.9	81.1	61.8	80.2	81.0	70.8	76.3	76.3	73.8	71.3	74.9
iderr all ft	59	55	165	117	99	325	820	151	293	283	213	220	324	1484
iderr named ft	54	53	67	72	38	138	422	144	198	281	165	130	258	1176
iderr named+unassg. ft	47	46	63	62	36	130	384	131	175	253	148	113	233	1053
iderr KNN = 1	22	14	13	11	2	25	87	18	24	39	18	8	19	126
iderr KNN = 5	33	27	35	22	13	49	179	26	45	66	32	20	46	235
iderr KNN = 20	42	35	45	41	20	74	257	47	70	123	68	53	101	462

where $\mathbb{1}\{\cdot\} = 1$ when the condition is true and 0 otherwise, and "speaking face assignment precision" is abbreviated as "spk precision". *spk assigned* is an important metric since this is the fraction of tracks used later for training the face models.

The obtained weak labels generally do not have a precision of 100%. Fig. 1 presents a scenario where we might encounter errors. For Everingham's method [11], the errors can be categorized into two types:

(i) *Track not found* (***t404***): This type of error occurs when the speaker is not visible on screen or does not have a face track. In such a scenario, we may erroneously assign a different on-screen face track as speaking.

(ii) *Speaking score higher for wrong person* (***hiscore***): This error can be attributed to the quality of the speaking score s_i, the feature used to determine whether a face track is speaking. For example (in Fig. 1), consider that *Leonard* (T6) is speaking and both *Leonard* (T6) and *Sheldon* (T5) appear on screen. We wrongly obtain a higher speaking score for Sheldon ($s_{Sheldon} > s_{Leonard}$), for example owing to failure in facial landmark detection or rapid head motion. Thus, the label *Leonard* is incorrectly assigned to *Sheldon*'s face track.

B. Speaking face analysis

Table I presents an in-depth analysis of the performance of a speaking face assignment method from [2] followed by its implications on identification. In the first section of the table, we provide an overview of the data set – number of tracks, subtitles, assignments and characters. The second section presents the speaking assignment performance (spk precision, spk assigned in %) using the metrics discussed above. Note that an operating point of high precision is favorable to train good face models.

We then list the number of errors made during the assignment (err total), and categorize them into *t404* or *hiscore*. Some of these errors can be corrected by leveraging additional information from the video. For example, we can assume that tracks that co-occur at the same time should be assigned a different identity. We also obtain links between tracks which should be assigned the same identity via shot threading and subsequent track threading (please see Sec. IV-B for a description of shot and track threading). Such cues can be used to correct some of the above errors.

In the Table I, we display the number of erroneous track pairs (err *uniq*), essentially counting the number of track pairs which are assigned the same identity and co-occur. Further, the labeling of track threads is said to have an error when more than one character name is assigned to tracks in a single thread (err *thread*). While at first these may seem to be small, the benefits of including both uniqueness and threading are evident in Sec. V.

C. Implications for identification

Using the weakly labeled face tracks (roughly 18% of all tracks) we train person-specific SVMs with a polynomial kernel of degree 2. Following [2], we extract block-wise Discrete Cosine Transform (DCT) features on each face of a track first aligned using the eyes and mouth centers. We obtain SVM scores for all tracks, select the highest scoring model and report the identification accuracy in Table I (id accuracy). We also present the number of errors over all face tracks (iderr all ft) and a breakup for all named tracks (iderr named ft). Note that [2] has difficulties dealing with unknowns as the background characters have very limited number of dialogs resulting in a weak unknown model. We further consider the erroneous tracks which were not assigned a name in the first weak labeling phase (iderr named+unassg. ft).

We propose a simple experiment to analyze the impact

Fig. 2. SDM-based facial landmarks on faces from a track. The landmark points are presented in black, while the points used to measure the amount of mouth opening – upper lip lower center and lower lip upper center are marked in white.

of weak labeling errors. For each episode, we first compute Euclidean distances between the unassigned and erroneous test tracks against all the training tracks (703 for BBT, 1098 for BUFFY). We consider how far – in k-Nearest-Neighbor sense – is the first error in the weak label which is likely to have caused the identification error.

We present the number of errors at three different distances in the kNN space. Roughly 22% (BBT) and 12% (BUFFY) of erroneous tracks are nearest neighbors to wrong weak labeled samples at $k = 1$. This number goes up to 46% (BBT) and 22% (BUFFY) at $k = 5$ and 67% (BBT) and 44% (BUFFY) at $k = 20$. Note that while we stop here, 20 tracks constitute less than 3% of all the training data for either series.

This suggests that reducing errors in the weak labeling phase of the identification process can directly help improve identification performance.

IV. Speaking Face Assignment Model

In this section, we describe our proposed joint assignment of the weak labels as a constrained function minimization problem. We begin by first introducing the employed features and constraints.

A. Speaking score

Mouth region block matching In order to make a fair comparison, we employ the same feature for determining the amount of lip movement as [2], [11], that is the minimum difference between regions around the mouth as determined via block matching. The raw difference is dual-thresholded into three classes of speaker scores for each frame: not-speaking (0.0), unsure (0.5) and speaking (1.0). Finally, the mean of the thresholded speaker scores in the duration of a subtitle is used as speaking confidence. A drawback of this method is that the block matching is computationally expensive.

Amount of mouth opening Recent improvements in facial landmark localization (*e.g.*, [4], [16], [20]) enable fast and highly accurate localization of facial landmarks in videos. In this paper, we employ the Supervised Descent Method (SDM) [20] to estimate facial landmarks. Fig. 2 shows the landmark points provided by SDM. We are especially interested in the upper lip lower center and lower lip upper center (marked in white). We use the spatial distance between these two points (normalized by the face height) to indicate the amount of lip motion in any given frame.

We filter the spatial lip distance using a band-pass filter at the expected frequency of lip motion with which a human speaks (we use 3.75 – 7.5 Hz), reducing noise and amplifying the true lip motion signal. We present the impact of filtering in Fig. 4.

In contrast to the block matching, this feature is computationally very efficient. Its computation time is mainly dominated by the facial landmark estimation, which has to be performed nevertheless for determining the mouth region for the block matching. We are thus able to compute this feature in real-time.

Speaking score for a track We denote the speaking score vector of each track t_i by $\boldsymbol{s}_i \in \mathbb{R}^{C\times 1}$, where C is the number of characters. Let c be the speaker (as determined from the subtitle-transcript matching) for subtitle u. We accumulate the speaking scores across all overlapping subtitles

$$\boldsymbol{s}_i = \sum_u \hat{\boldsymbol{e}}_c \cdot \psi_u^c \quad , \tag{3}$$

where $\hat{\boldsymbol{e}}_c$ is the unit vector in dimension c, *i.e.* a vector where the value of dimension c is 1 and all other values are 0. ψ_u^c is the sum of the amount of mouth opening (or mouth region difference) within the duration of the subtitle-track overlap.

B. Pair-wise track constraints

Positive constraints like track threading and negative constraints have been recently applied to face track clustering [19]. We now use them for weak label assignment.

Negative constraints In order to avoid assigning the same name to co-occurring face tracks, we form negative constraints between all such pairs of tracks. Let the set of negative constraints be $\mathcal{N} = \{(t_i, t_j)\}$, where track t_i and t_j overlap in time by at least one frame. We will describe how these constraints can be employed to induce a penalty when co-occurring tracks are assigned the same identity. This helps reduce errors and admits tolerance in the speaking score extraction method. Such constraints have been used before, albeit for the actual *identification*, not during the speaker assignment (*e.g.*, [2], [6]).

Positive constraints We obtain positive constraints between face tracks across shots from *shot threading* [7]. Due to the video editing, shots often alternate between two (or more) camera angles without much camera motion between corresponding shots. Following [19], we find these shot-threads by matching local features between the end-frame of one shot and start-frames of the following shots. We assume that shots in a thread do not have much camera motion and that characters appear in the same spatial position in the frame. Thus, two tracks which are found at the same location

in consecutive shots of a thread are likely to be of the same character.

Consider a shot thread $\mathcal{S}^q = \{z_i^q\}$, consisting of a set of shots stemming from the same camera angle or viewpoint. Let the face tracks in each shot z_i^q of the thread be denoted by $t_{i,j}^q$. The positive constraints (track pairs) are obtained by linking together tracks across shots of a thread based on the amount of overlap of their bounding boxes.

$$\mathcal{P} = \{(t_{i,j}^q, t_{i+1,k}^q) \;\; \forall q, i, j, k \text{ and } \phi(t_{i,j}^q, t_{i+1,k}^q) > 0.5\} \tag{4}$$

where the function $\phi(\cdot,\cdot)$ is the intersection over union overlap measure of the mean locations of the bounding boxes of the two tracks under consideration.

Given a set of pairs, we compute the transitive hull and form cliques of tracks. Such cliques were used for clustering in [19]. Each such clique is called a *track thread*. We present a brief evaluation of track threads in Sec. V-D.

C. Joint speaking face assignment

We now describe our proposed framework to encompass all the available information for a joint assignment of speaker names to tracks. For each face track t_i, we define a random variable (vector) $\boldsymbol{x}_i \in \mathbb{R}^{C\times 1}$, where C is the number of characters including the *unknown* class. $\boldsymbol{x}_i$ serves as the probability of assigning one of the C names to the track t_i and is constrained such that $\sum_{c=1}^{C} x_i^c = 1$.

The positive and negative constraints from the previous section are treated as soft constraints since they are obtained automatically and can be wrong (although very rarely). For example, a mirror in the scene can result in two co-occurring tracks of the same person (thus violating the uniqueness assumption), while a swap in the spatial position of characters within a shot thread can cause errors in track threads.

To obtain the values of $\boldsymbol{x}$ over all tracks, we define an energy function involving four terms. They are either maximized (↑) or minimized (↓) in the optimization.

1) *Speaking score term* (↑): incorporates the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{s}_i$. Note that $\boldsymbol{s}_i \in \mathbb{R}^{C\times 1}$ is usually a sparse vector with a single non-zero entry at the dimension of the character obtained via subtitle-transcript matching. To assign track t_i to the speaker, we wish to maximize $\boldsymbol{x}_i^T \boldsymbol{s}_i$ subject to $\sum_c \boldsymbol{x}_i^c = 1$ constraint. In the absence of the other energy terms, the ideal value of $\boldsymbol{x}_i$ is a vector of zeros with a single 1 at the location of non-zero $\boldsymbol{s}_i$.
2) *Uniqueness term* (↓): applies to pairs of tracks (t_i, t_j) and is calculated as $\boldsymbol{x}_i^T \boldsymbol{x}_j, \forall (t_i, t_j) \in \mathcal{N}$. Minimizing this term promotes assignment of different labels to the track pair $\hat{y}_i \neq \hat{y}_j$.
3) *Threading term* (↑): applies to all possible pairs of tracks within a track thread. Similar to the uniqueness, this is calculated as $\boldsymbol{x}_i^T \boldsymbol{x}_j, \forall (t_i, t_j) \in \mathcal{P}$. To label the tracks in a thread with the same identity, we maximize this term.
4) *Regularization term* (↓): our final term regularizes the values of $\boldsymbol{x}_i$ and makes sure that a small speaking score does not have a large effect on the values of $\boldsymbol{x}_i$. Since $\boldsymbol{x}_i$ is constrained to sum to 1, the regularization essentially promotes values of $\boldsymbol{x}_i$ closer to $1/C$ (uniform prior). This term is calculated for all tracks as $\boldsymbol{x}_i^T \boldsymbol{x}_i, \forall t_i$ and is minimized.

The optimal values for the assignment are obtained by minimization of a weighted combination of the above terms, posed as a constrained minimization problem:

$$\begin{aligned} \mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \;\; & -w_S \sum_i \boldsymbol{x}_i^T s_i + w_U \sum_{(t_i,t_j)\in\mathcal{N}} \boldsymbol{x}_i^T \boldsymbol{x}_j \\ & - w_T \sum_{(t_i,t_j)\in\mathcal{P}} \boldsymbol{x}_i^T \boldsymbol{x}_j + w_R \sum_i \boldsymbol{x}_i^T \boldsymbol{x}_i \quad , \\ & \text{subject to } \sum_{c=1}^{C} x_i^c = 1 \quad . \end{aligned} \tag{5}$$

Terms to be maximized appear with a negative sign. The weights w_* steer the relative importance of the respective feature/constraint.

Assignment confidence Given the best value of $\mathbf{x}^*$, we now compute the assigned label and confidence for each track. The assigned label for track t_i is

$$\hat{y}_i = \underset{c}{\operatorname{argmax}}\, x_i^c \tag{6}$$

where $c \in \{1 \ldots C\}$, the number of characters. The assignment confidence is the difference between the highest and the second highest score in $\boldsymbol{x}_i$. That is, a uniform assignment of $1/C$ to all identities yields a confidence of 0 whereas an assignment of 1 to one of the characters and 0 to all others yields a confidence of 1. The precision vs. assigned curves shown in Fig. 3 are based on this confidence.

D. Implementation details

Constrained function minimization We use an interior-point algorithm as implemented in MATLAB to minimize Eq. 5. For all tracks, $\boldsymbol{x}_i$ is initialized as $x_i^c = 1/C$. The optimization is constrained as $\sum_c x_i^c = 1$ and $0 \leq x_i^c \leq 1$. When not specified otherwise, the weights for the optimization are $w_S = 1, w_U = 1, w_T = 1$ and $w_R = 3$.

Fast and efficient Not all tracks are connected via a series of a negative or positive constraints. Thus, we form cliques of tracks which are connected by either positive or negative constraint and optimize each clique independently. This makes the optimization efficient since the problem size reduces compared to a joint assignment of all tracks in an episode. For an episode of 20 minutes with roughly 600-700 face tracks we typically require less than 2 minutes to solve the optimization for all cliques.

Subtitle offsets The duration of a subtitle need not represent the amount of time for which a person actually speaks. We notice that there are four typical ways in which subtitles are created: (i) A character has a long dialog which is split into multiple subtitles: the first subtitle appears about 0.1

seconds before the dialog begins, the subtitles in between are well segmented, and the last subtitle stays on screen for roughly 0.2 seconds after the end of the dialog. (ii) Different characters have a dialog without a break: since the dialog is very fast, the subtitles usually follow each other without intermediate buffers. (iii) Only one dialog from one character: such a subtitle has a buffer of 0.1 seconds at the beginning, and 0.2 seconds at the end. (iv) Multiple characters have dialogs in the same subtitle: if a subtitle contains text from multiple characters it is very common to add a "–" at the beginning and split the dialog into separate lines. Taking these buffer timings into account and reducing the duration of the speaking score provides a small further improvement for the baseline and our proposed method.

We will make the code publicly available at http://cvhci.anthropomatik.kit.edu/projects/mma

V. Experiments

We evaluate our proposed speaker assignment approach both in terms of the spk precision/assignment, and the subsequent identification performance when using the generated weak labels for training face models and identifying all tracks. To distinguish between the two contributions, we first present results of using the speaking face assignment model on the original speaking score feature. We then introduce the new speaking score based on the amount of mouth opening, compare the weak labeling performance and show that the identification yields improved results.

A. Data set

We use the updated KIT TV data set from [2] which comprises two TV series: (i) BBT: The Big Bang Theory (season 1, episodes 1-6) and (ii) BF: Buffy the Vampire Slayer (season 5, episodes 1-6).

B. Joint assignment model evaluation

We evaluate the performance of the joint speaker assignment in multiple stages.

Model vs. [2] In Fig. 3 we compare the speaker precision vs. the fraction of tracks which are assigned a name.

For a fair comparison, we use the mouth-region block-matching-based speaking score from [2] as the underlying feature. The baseline assignment method [2] is denoted as *Baseline* in Fig. 3. For our proposed approach, we report results including different combinations of the terms of Eq. 5 to show their relative potential. Incorporating the speaking score only with the regularization term (Fig. 3 *Model*) performs similar to the baseline. This is not surprising since this results in an independent optimization problem for each track due to the absence of any constraints, which is very similar to the old method.

Adding the uniqueness term (*Model+U*) consistently improves performance. On the other hand, adding the threading term (*Model+T*) initially introduces more errors. This can be explained by the fact that an erroneous assignment can be propagated as easily as a correct assignment by threading.

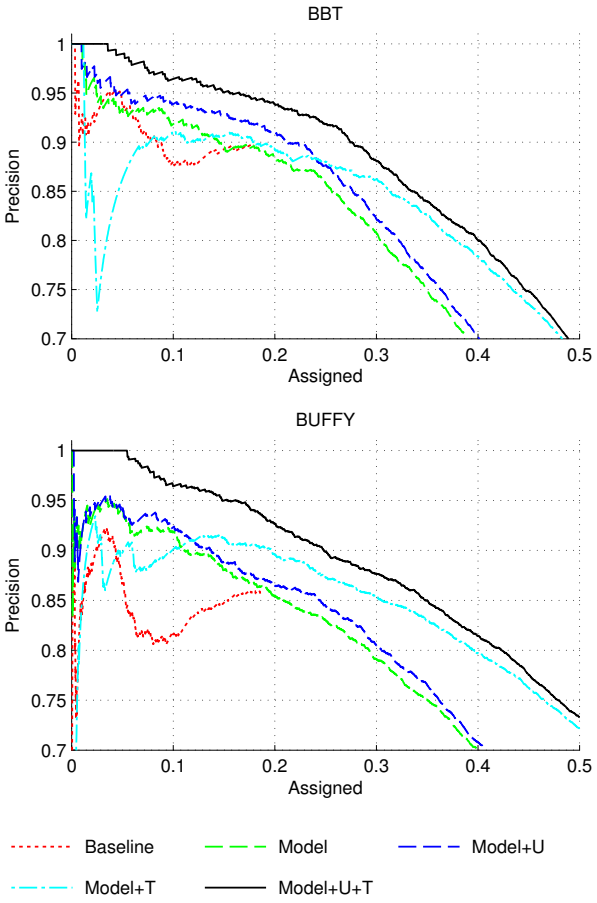

Fig. 3. ASSIGNMENT MODEL: Comparison of speaking face precision (spk precision) vs. fraction of assigned tracks (spk assigned) for BBT (top) and BUFFY (bottom). In comparison to the baseline, the usage of the model with uniqueness and threading improves the overall performance significantly. This figure is best viewed in color.

Using all terms – regularization, uniqueness and threading (Fig. 3 *Model+U+T*) – we obtain the best performance. Compared to the baseline, we consistently gain about 5% in precision or assign names to 5% more tracks depending on the choice of the operating point.

Apart from the improved precision *and* number of assigned tracks, our model also simplifies the choice for an operating point. The precision-assignment curves of the final complete model are much smoother and do not behave as erratic as the baseline.

Error analysis We compare the errors in the assignment of names to tracks in Table II. As in Table I, we consider the two types of errors: *t404* and *hiscore*. To account for additional errors introduced by the threading (and which are not of the above types), we add a new error type – err *other*.

We compare the baseline (denoted as *Old*) against the usage of the model with all the terms (*M+U+T*). We choose the operating point such that the number of assigned tracks is roughly the same. With this, we achieve a 5-8% improvement in precision while assigning the same number of tracks.

Secondly, a drastic reduction in the total number of errors (err total) can be observed. While a few new errors are introduced, overall about 45-50% of the errors are mitigated.

TABLE II

ERROR ANALYSIS OF THE SPEAKING FACE ASSIGNMENT.

	BBT		BUFFY	
	Old	M+U+T	Old	M+U+T
spk precision	89.8	94.3	85.9	93.5
spk assigned	17.9	18.0	18.7	18.8
err total	72	40	155	72
err *t404*	13	7	46	11
err *hiscore*	59	27	109	50
err *other*	0	6	0	11
err *uniq*	14	8	14	30
err *thread*	6	0	21	5

The number of track threads with an error (err *thread*) reduces significantly, while the number of unique pair errors are relatively stable. The weights w_U and w_T play an influential role in the optimization and typically trade off between the two types of errors.

C. Speaking score feature evaluation

In Fig. 4, we compare the baseline feature against our proposed SDM-based mouth opening feature with and without filtering. Here *Model+U+T* refers to full model with mouth region matching. *no BPF* denotes the use of the new feature without filtering. Applying the band pass filter results in a small improvement which is seen in Fig. 4 *with BPF*.

The block matching based feature [11] requires 3.6ms per face image on a parallelized implementation which uses 8 cores. On the other hand, our feature including band-pass filtering takes on average 0.15ms per face image on a single core.

D. Track threading evaluation

Track threading is essentially a form of track clustering. Errors in the track threading can lead to erroneous links in the final joint optimization. As an evaluation measure, we use the purity of a track thread, *i.e.* the fraction of tracks in a thread which have the same ground truth identity against all tracks. Similar to clustering purity, we combine the purity of all track threads by weighting the purity by the number of tracks it contains.

On the BBT data set we are able to form 381 track threads containing on average 3.07 tracks. The weighted track thread purity is 99.83%. On the other hand, we form 601 track threads on the BUFFY data set, containing 3.38 tracks on average and having an overall purity of 99.39%.

E. Face identification evaluation

We evaluate the impact of the improvements on the accuracy of labeling all face tracks in the episode. For a fair comparison, we extract DCT features (as in [2]) and train person-specific SVMs with polynomial kernel. We present the results in Table III. The accuracy of the baseline is presented in the first row (baseline [2]). In the second row, (model) we present the accuracy using the new model. We obtain a consistent improvement across all episodes and an overall boost of 1.0% on BBT, and 0.7% on BUFFY. The last row presents the identification accuracy using the new

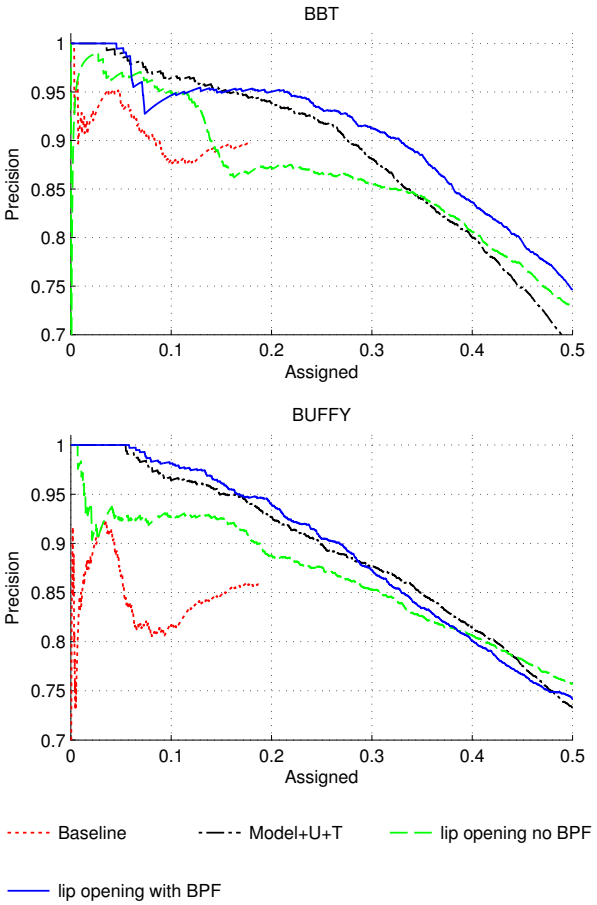

Fig. 4. LIP OPENING FEATURE: Comparison of the speaking face precision (spk precision) vs. the fraction of assigned tracks (spk assigned) for BBT (top) and BUFFY (bottom). While the new speaking score is very easy to compute, its performance after band pass filtering is very similar to the old block matching based technique.

mouth opening based feature. Compared to the baseline, the feature is very simple, fast to compute, and provides an additional improvement of 1% and 2.2% on BBT and BUFFY respectively.

Finally, Fig. 5 presents the variation in identification accuracy across different operating points. Firstly, please note that the baseline is operated at the best possible value since the heuristic method cannot assign names to more than 18-20% of the face tracks. Our model with the new lip opening feature performs better than the baseline at all points.

For the best accuracy, we observe that we need to shift the operating point to a higher percentage of assigned tracks. This is partly due to threading which inherently assigns names to more tracks. At a low percentage of assigned tracks, more characters have zero training data which results in a lowered accuracy.

VI. CONCLUSION

We present a comprehensive analysis of the current state-of-the-art speaking face assignment method providing an insight into the types of errors made. We further analyze the influence of these erroneous assignments on the identification performance. An underlying cause of these errors is that tracks are treated independent from one another.

TABLE III

FACE IDENTIFICATION PERFORMANCE ACROSS ALL TRACKS OF THE EPISODES, MEASURED AS IDENTIFICATION ACCURACY.

	BBT-1	BBT-2	BBT-3	BBT-4	BBT-5	BBT-6	Total	BF-1	BF-2	BF-3	BF-4	BF-5	BF-6	Total
baseline [2]	91.0	91.1	75.0	80.9	81.1	61.8	80.2	81.0	70.8	76.3	76.3	73.8	71.3	74.9
model	91.5	92.0	76.7	80.9	83.2	**62.9**	81.2	82.0	70.0	77.3	75.8	76.8	**71.6**	75.6
model + lip opening	**92.9**	**92.7**	**78.6**	**83.0**	**84.7**	61.5	**82.2**	**83.7**	**75.7**	**79.3**	**77.7**	**80.0**	70.4	**77.8**

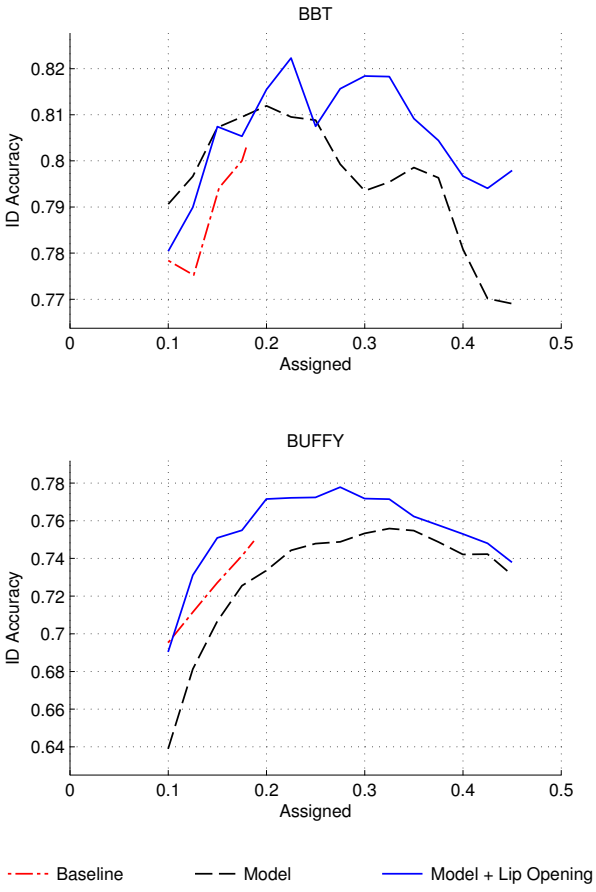

Fig. 5. Variation in identification accuracy of all tracks for different choices of operating points.

In order to mitigate these errors, we propose to include uniqueness constraints between co-occurring pairs of tracks and positive constraints between tracks of a *track thread* in a joint speaker assignment procedure. We present results on two existing TV series data and show that our method improves assignment performance, while also providing a boost to the identification accuracy over all tracks. In addition, based on recent advances in facial landmark localization, we employ a simple and efficient feature to detect which face track is speaking. This further enhances both the weak label assignment and identification of all tracks.

Outlook As noted before, different subsequent identification schemes have been explored using the method from [11] for generating weak labels (*e.g.*, [2], [17]). In this paper, we employ simple one-vs-all SVM classifiers in our experiments which can be replaced with more sophisticated methods such as Multiple Kernel Learning [17] or a semi-supervised framework [2] which further incorporates unlabeled data and constraints into the learning. A similar boost in performance can be expected for these identification schemes. An interesting avenue for future work might be to perform the speaker assignment and identification jointly, using for example the appearance of the potential speaker as further cue on his/her identity.

Acknowledgments This work was funded by the Deutsche Forschungsgemeinschaft (DFG - German Research Foundation) under contract no. STI-598/2-1. The views expressed herein are the authors' responsibility and do not necessarily reflect those of DFG.

REFERENCES

[1] I. Arsic and J.-p. Thiran. Mutual information eigenlips for audio-visual speech recognition. In *European Signal Processing Conference (EUSIPCO)*, 2006.
[2] M. Bäuml, M. Tapaswi, and R. Stiefelhagen. Semi-supervised Learning with Constraints for Person Identification in Multimedia Data. In *CVPR*, 2013.
[3] P. Bojanowski, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic. Finding Actors and Actions in Movies. In *ICCV*, 2013.
[4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face Alignment by Explicit Shape Regression. In *CVPR*, 2012.
[5] H. Çetingül, E. Erzin, Y. Yemez, and a.M. Tekalp. Multimodal speaker/speech recognition using lip motion, lip texture and audio. *Signal Processing*, 86(12):3549–3558, Dec. 2006.
[6] R. Cinbis, J. Verbeek, and C. Schmid. Unsupervised metric learning for face identification in TV video. In *ICCV*, 2011.
[7] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. Movie/script: Alignment and parsing of video and text transcription. In *ECCV*, 2008.
[8] T. Cour, B. Sapp, C. Jordan, and B. Taskar. Learning from ambiguously labeled images. In *CVPR*, 2009.
[9] T. Cour, B. Sapp, A. Nagle, and B. Taskar. Talking Pictures : Temporal Grouping and Dialog-Supervised Person Recognition. In *CVPR*, 2010.
[10] T. Darrell, JW Fisher, P. Viola, and W. Freeman. Audio-visual segmentation and "the cocktail party effect". In *International Conference on Multimodal Interfaces (ICMI)*, 2000.
[11] M. Everingham, J. Sivic, and A. Zisserman. "Hello ! My name is ... Buffy" Automatic Naming of Characters in TV Video. In *British Machine Vision Conference (BMVC)*, 2006.
[12] M. Gurban and J.-P. Thiran. Information theoretic feature extraction for audio-visual speech recognition. *IEEE Transactions on Signal Processing*, 57(12):4765–4776, 2009.
[13] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Learning to Recognize Faces from Videos and Weakly Related Information Cues. In *IEEE Intl. Conference on Advanced Video and Signal-based Surveillance (AVSS)*, 2011.
[14] H. McGurk and J. MacDonald. Hearing lips and seeing voices. *Nature*, 264:746 – 748, 1976.
[15] V. Ramanathan, A. Joulin, P. Liang, and L. Fei-Fei. Linking people in videos with "their" names using coreference resolution. In *ECCV*, 2014.
[16] S. Ren, X. Cao, Y. Wei, and J. Sun. Face Alignment at 3000 FPS via Regressing Local Binary Features. In *CVPR*, 2014.
[17] J. Sivic, M. Everingham, and A. Zisserman. "Who are you?" – Learning person specific classifiers from video. In *CVPR*, 2009.
[18] M. Tapaswi, M. Bäuml, and R. Stiefelhagen. "Knock! Knock! Who is it?" Probabilistic Person Identification in TV-Series. In *CVPR*, 2012.
[19] M. Tapaswi, O. M. Parkhi, E. Rahtu, E. Sommerlade, R. Stiefelhagen, and A. Zisserman. Total Cluster: A person agnostic clustering method for broadcast videos. In *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, Dec. 2014.
[20] X. Xiong and F. D. l. Torre. Supervised Descent Method and its Applications to Face Alignment. In *CVPR*, 2013.