Online Identities Re-assignment for Tracking Recovery

Cao Khanh Nguyen[1], Taylor Mordan[1,2], and Alexandre Alahi[1,2]

[1]École Polytechnique Fédérale de Lausanne (EPFL)
[2]Visual Intelligence for Transportation (VITA)

June 17, 2022

Abstract

In this project, we want to extend the OpenPifPaf [7] to effectively address the issue of tracking identities mis-matches. Given the current implementation of OpenPifPaf, we aim to develop a plug-in framework that can be easily integrated into the pose tracking pipeline. The final deliverable for this project is a standalone package that reduced 37.8% of the number of switches in identities matching.

1 Introduction

Pose tracking is a combined task for pose estimation and articulated tracking in video. On the scope of this project, we would mainly focus on improving the articulated tracking performance of pose tracking frameworks.

Specifically, we want to extend OpenPifPaf [7], which currently loosely addresses the person identification in its architecture. Our extended framework combined OpenPifPaf with Deep Visual Re-identification with Confidence model [1] to reassign the tracking identifications (IDs) in an online manner.

We also made experimentation to evaluate different ways to improve our framework. For effective evaluation, we extended the `poseval` [2] package to include more relevant metrics for IDs reassignment task, namely `accuracy`.

Our final framework showed a 37.8% decrease in identity switches and 2% increased in identities matching accuracy. The work is developed as a standalone package and can be integrated into any detection method.

2 Related Work

There have been existing methods aim at effectively fusing the person re-identification (re-ID) task into multiple object tracking (MOT). For example, *Zhang et al.* proposed a methods that balanced the importance of tracking and re-ID tasks in an one-shot approach, i.e. detecting objects and re-ID feature in a single network [10]. Another school of approaches is to treat the re-ID as a separated process and trained two separate models for tracking and re-ID respectively [8].

Since the objecttive of the project is to build upon OpenPifPaf, we approach the problem in a similar way to the second school of thoughts, i.e. to combine a re-ID model to the detections and tracking of OpenPifPaf.

2.1 OpenPifPaf

OpenPifPaf was originally developed as a pose estimation framework [6] and then extended to handle pose tracking tasks [7]. Currently, OpenPifPaf utilizes *Temporal Composite Association Fields* to form association in images sequence, resulting in *temporal poses*. The method then assigns track ID if there is an association in the temporal pose between frame t_i and frame t_{i-1} and assigns a new track ID otherwise. Therefore, the framework would assign a new track ID for an existing identity if this identity is not tracked in certain frames. Possible reasons for this problem might be because of occlusion, mis-tracked, or object going out-of-view. We would want to extend OpenPifPaf to handle these exceptions and improve the tracking performance to be more robust.

For this framework, we adopted the pre-trained tracking model from OpenPifPaf on the Posetrack2018 dataset [2], with the ShuffleNetv2 [9] as the baseline.

2.2 Deep Visual Re-Identification with Confidence

The new framework also makes use of a re-ID model, which is Deep Re-Identification with Confidence model [1]. We utilized this model as it has been developed in VITA lab and thus is easier to get access to. The model used in this framework is re-trained using confidence loss on the Market-1501 dataset [11] with Resnet-50 baseline [5].

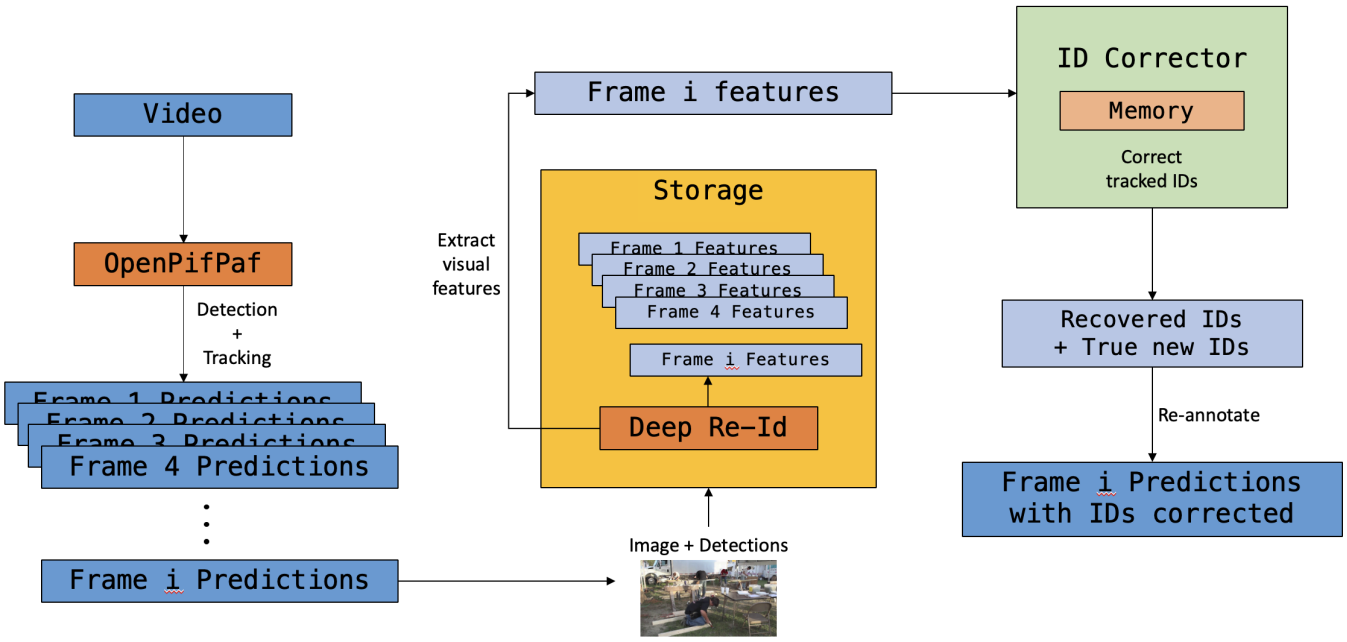

Figure 1: Framework architecture

3 Proposed Method

The main components of all architectures are shown in Figure Figure 1. As OpenPifPaf processes the detection and tracking frame-by-frame, we developed our system in an online manner. An online processing pipeline requires inputs to be processed at each sequential step, which corresponds to 'frame' in this case. Our framework, thus, will take the predictions at each frame as an input, and output the predictions of the same frame with corrected IDs assignments. The high overview architecture consists of two main components: Storage and ID Corrector.

3.1 Storage

The storage takes the image and predictions of each frame as an input. Its main purpose is to extract visual features from the input. The storage contains the deep re-ID model. Upon receiving the image and predictions of each frame, the storage would extract the bounding boxes of each identity within the frame, alongside with their metadata. Then those bounding boxes are fed into the deep re-ID model to extract the visual features vectors that correspond to each identity. The feature vectors is of size `1 x 2048` and is stored as a `dictionary` that corresponds to each identity in the frame.

Note that the feature extractor, which can be passed as an argument, is not limited the Deep Visual Re-Identification with Confidence model, but can also be an extractor from other Person Re-Identification methods. The feature dimension of `n x 2048` is not a strict requirement either. This approach is suitable for future extension of the framework. Besides, the storage would all the metadata and features of the whole video. While this auxiliary steps might cause an overhead in memory, it is essential for the debugging process. Optimization can be made to skip this step and ignore the storage of old frames. In that case, the storage would simply act as a feature converter.

3.2 ID Corrector

The ID Corrector would take the feature vectors of each frame as an input. On a high level overview, it will compare the query (input) to a short-term memory, and then match the new IDs to a potential candidate ID in the past. If the distance between the query and the candidate is lower than a certain threshold, it would declare a match and reassign the new ID into the ID of the matched candidate. The high overview of the ID Corrector pipeline is shown in Figure 2

We further explain the construction of the ID Corrector. The component would consists of a memory which would stores the feature vectors of all the IDs in a short-term manner, which means that the system would not store every appearances of the each unique identity in the previous frame. From the input feature vectors of the current frame t_i, the Corrector will extract the newly tracked IDs and set the vectors of these new IDs as the `query`. The Corrector also extract the IDs that was continuously tracked during the previous frame t_{i-1}.

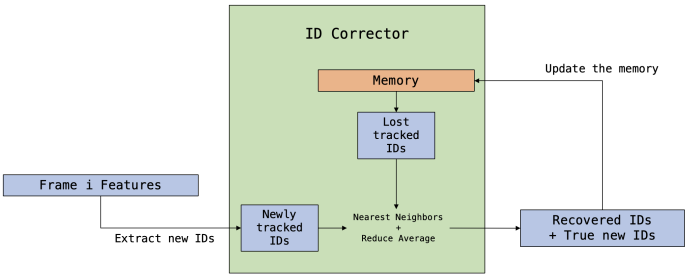

Figure 2: ID Corrector Pipeline

These newly and continuously tracked IDs would then be temporarily discarded from the memory. The remaining IDs from the memory would be set as the `gallery`. At this point, the Corrector will run a Nearest Neighbors algorithm between the `query` and the `gallery` to select the candidates with the best visual similarities. We then performed a Reduce Average to increase the robustness of the selected candidates. Finally, we select the unique ID with the lowest average distance from the query, if that distance is lower than a certain threshold, as the recovered ID. If there is no match or the distance is larger than the threshold, the `query` is deemed as an true new ID. The Corrector would then update the memory with the recovered and true new IDs.

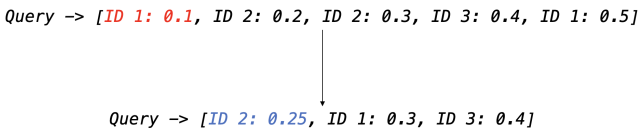

Figure 3: Reduce Average. The original best candidate is ID1 but after reducing average, the selected candidate is ID2 as there are more instances of it with low distance from the candidates pool

There are also hyperparameters that determine how the memory would be constructed. Those hyperparameters are describeb below.

Mode: the mode in which the memory is constructed, there are currently supports for three modes:

- `recent`: the memory would store only the most recent appearances of each ID.
- `sparse`: the memory would store only the appearances of each ID intermittently, with each appearance 5-frame away from the the others.
- `first`: the memory would store only the first appearances of each ID.

Note that, there are potentially many algorithms on how to effectively construct the memory, which can also make use of poses information from each identity. Due to the time constraint of the project, this feature is not yet developed but would be of interesting topic for further research.

Memory length: The length of tracklet which the memory would store for each ID.

Max rank: The number of candidates that would be selected from the `gallery` before reduce average

Threshold: The threshold to determine whether the selected candidate is an eligible ID to recover.

After the recover IDs and true new IDs are determined, the ID Corrector would re-annotate the predictions of the current frame and continue onto the next frame. The information from future frames bear no impacts on the ID re-assignment of the current frame. Thus, it is considered an online identity recovery.

4 Evaluation

4.1 Experimental Setup & Metrics

For the evaluation of the framework, we evaluated its performance on the Posetrack2018 dataset [2]. We did our local evaluation on the validation set of Posetrack2018.

For the metrics, we initially used **Multiple Objects Tracking Accuracy** (MOTA) [3] to evaluate the performance, as it is the standard metric evaluate multiple objects tracking. However, a deeper analysis of the evaluation metric suggested us to only focus on one component of the metric, which is the mismatch errors (MME). The MME only recorded the number of time an ID is switched from its matching ID in the ground truth. There are two reasons for this decision: First, MME is often dominated by the other components and therefore a significant change in MME might not be reflected properly on the overall MOTA. Second, this metric is directly relevant to our problem, as the reassignment of the IDs does not have any impact on the object detection task.

Furthermore, we extended our metrics based on the MOTA pose evaluation package for Posetrack2018. The original metric only detect the tracking performances of the *Joints*, which might be misleading as the matching of *Joints* is based on distance and not associated with the identity of the *Person*. We therefore extended the package to also evaluate the MME of *Person* IDs. More importantly, we found that MME on its own does not address cases where the ID of a new person got assigned to the ID of a different identity previously appeared, which means that excessive IDs reassignment (assigning IDs of true new identity to an old one) will decrease the number of switches and thus decrease MME.

Therefore, following the inspiration from evaluating the number of time in the sequence where the matching IDs is incorrect [4], we developed the **True Accuracy**, which measures the accuracy of the tracked ID against the original matched ID from ground truth, and **Recovered Accuracy**, which evaluation the accuracy in which the tracked IDs belong to any IDs matched to the identity in the past. The comaprison between the two accuracies is shown in Figure 4 and Figure 5. The dictionary matches the ground truth ID and the corresponding tracked ID from a certain sequence. In the Recovered Accuracy, each frame would have its own ground truth, as the history of IDs matched to the ground truth grows over time.

```
{0: 2, 1: 1, 2: 5, 3: 4, 4: 8, 5: 9, 6: 3, 7: 7}
```

Figure 4: True Accuracy matches of IDs

```
{0: {0: [2], 1: [1], 2: [5], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 1: {0: [2], 1: [1], 2: [5], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 2: {0: [2], 1: [1], 2: [5], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 3: {0: [2], 1: [1], 2: [5], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 4: {0: [2], 1: [1], 2: [5], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 5: {0: [2], 1: [1], 2: [5, 10], 3: [4], 4: [8], 5: [9], 6: [3], 7: [7]},
 6: {0: [2], 1: [1], 2: [5, 10], 3: [4], 4: [8], 5: [9, 12], 6: [3], 7: [7]}
```

Figure 5: Recovered Accuracy on each frames. From frame 5 onward, both ID5 and ID10 from tracking is considered an accurate match for ID2 in ground truth

4.2 Experimentation

We performed a manual grid search experiment to detect the best performance combination of hyperparameters. First, we conducted experiments on selecting the potentially best threshold. As the threshold of the framework increases, there are more reassignments of the IDs, which lead to the previously mentioned issue in the MME metrics. Figure 6 shows that a steep decrease in MME does not necessarily means that the tracking performance is improved. While there is a decrease in number of switches, the system definitely performs worse, as new identities is assigned to IDs of existing person.

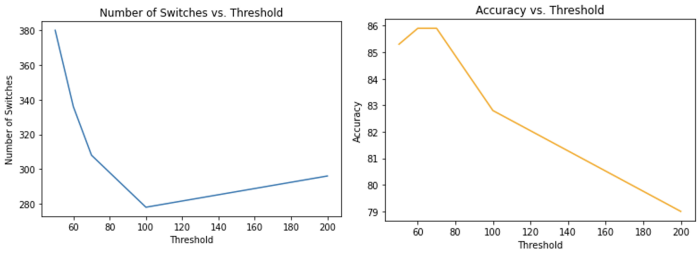

Figure 6: Trade- off between MME and Accuracy metrics as the number of reassignments increases (increased threshold)

From the result, we chose the 70.0 as our threshold, as it provides the best trade off between MME and True Accuracy. We then test the different modes of our framework with the following setting: memory length - 5, max rank - 5, threshold - 70.0. The result can be found in Table 2

	original	`recent`	`sparse`	`first`
Person MME	498	308	323	338
True Acc	83.9	85.9	86.3	85.8
Recovered Acc	_	92.9	93.1	93.4

Table 2: The mode *recent* seems to perform most consistent in terms of both MME and accuracy

	ML-1	**ML-5**	**ML-10**
MR-1	361	288	280
MR-5	360	308	292
MR-10	360	310	311

Table 3: A manual grid search to select the best parameters - MME

	ML-1	**ML-5**	**ML-10**
MR-1	94.5	91.8	91.6
MR-5	94.4	92.9	91.8
MR-10	94.4	93.3	93.0

Table 4: A manual grid search to select the best parameters - Recovered Accuracy

	Original Prediction	Modified Prediction
Joints MME	5420	3658
Person MME	498	310
True Accuracy	83.9	86.0
Recovered Accuracy	_	93.3
Posetrack MOTA	65.069	65.415

Table 1: Final result

Finally, we performed a grid search over the ranges [1, 5, 10] for memory length and max rank, of which the result can be found in Figure 3 and Figure 4. As a final method, we selected our parameters setting to be: memory length - 5, max rank - 10, threshold - 70.0. This selection, while being based on experiments, was acknowledgely heuristic as we have to find the optimum point between MME and Accuracy. With this final method, we achieved the following benchmark. The framework show a 37.8% decrease in number of IDs switches and 2% increase in True Accuracy. The detailed result can be found in Table 1.

5 Qualitative Results

We select a number of qualitative results to show. The samples are chosen from the Posetrack2018 validation set in which the framework has reduced at least one number of ID switches. The color code for these sample results are detailed in the following:

- Green: prediction ID matching ground truth ID
- Red: prediction ID not matching ground truth ID
- Dark Green: ID that has been recovered and match ground truth ID
- Orange: ID that has been recovered but does not match ground truth ID
- Grey: False Positive

Figure 7: A sample of a person recovered the ID after being occluded. Images on the first line are original predictions from OpenPifPaf for frame 0 and frame 5, from left to right. Images on the second line are the modified predictions for the same frames.

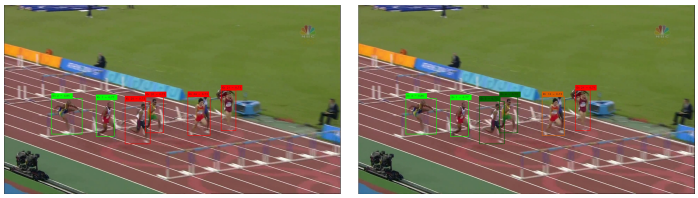

Figure 8: The framework does not work perfectly. There are cases when it assigns new IDs for the identities yet these new IDs does not matches the ground truth IDs. However, in general there is an improvement in the tracking performance.

6 Known Limitations

There are some limitations that is known when designing the framework. This section details some of them and potential adjustments that can be made to alleviate those limitations.

Figure 9: Partial detection would affect the visual features as well as robustness in IDs assignments

First, the framework is vulnerable to inaccurate detections from OpenPifPaf (Figure 9). The inaccurate detections might led to unexpected IDs switching. At the same time, detecting only a portion of pose would lead to dissimilar visual features from the same identity. As a result, the framework, relying on a deep re-ID model, might not be able to detect that these visual feature are from the same person. The only way to alleviate this limitation is to improve the detection performance of OpenPifPaf.

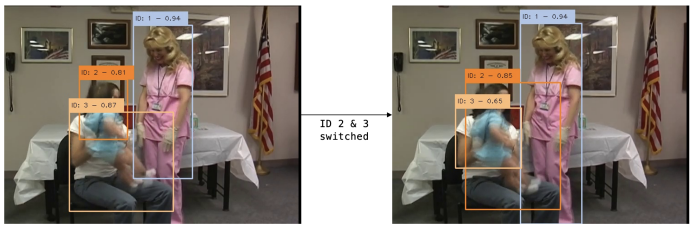

Figure 10: The random switching of two persons' ID in two continuous frames directly contribute to the MME metric. Besides, these cases also have a negative impacts on the way the short-term memory is constructed

Second, the framework currently does not handle cases when

there are ID switches within continuous frames (Figure 10). These cases occupy a large percentage of MME within the Posetrack2018 dataset. Furthermore, in cases when these IDs are lost in future frames, the inherent switches would have a detrimental impact on the short-term memory, as the same ID might have several different visual features. We can extend the framework to address this problem by appending a re-ID process on the continuously tracked IDs to correct potential switchings. Yet, further optimizations for robust performances is expected.

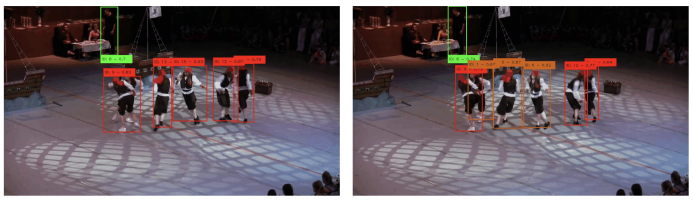

Figure 11: A video consists of identities that look similar to each other might pose a challenge to our framework

Finally, the framework heavily depends on the performances of the deep re-ID model, which is particularly relying on visual cues. Therefore, it cannot handle very well inputs with visually similar identities (Figure 11). To alleviate this limitations, it is essential to improve the performances of the deep re-ID model and to experiment with various re-ID approaches.

There are potentially many different aspect of the project not fully discovered. For example, evaluation on the Posetrack2018 dataset might not be the best way to determine the performance of the framework, as the ground truth annotation ignore cases when people going out-of-view and return. On a similar manner, a lot of other potential improvements can be made to make use of OpenPifPaf's pose information to construct the memory system, which, as in the feedbacks, is an ongoing interesting research topic in the re-ID community.

7 Conclusion

In this work, we present an extension to the OpenPifPaf framework. We developed an online identities recovery system to address OpenPifPaf's drawbacks in assigning tracking IDs task. The framework developed was a combination of OpenPifPaf's predictions and a deep re-identification model, which is responsible for recovering IDs based on visual cues. We also extended the framework as a standalone package with a set of adaptable APIs, thus can be generalized to any other detection methods.

The project was a steep learning curve for us, both in the fields of engineering and deep learning. At the very end, we managed to provide a deliverable that alleviated around 40% of the original problem. However, we believe the project still have rooms for improvement and are open for future changes.

References

[1] George Adaimi, Sven Kreiss, and Alexandre Alahi. Deep visual re-identification with confidence. *Transportation Research Part C: Emerging Technologies*, 126:103067, 2021.

[2] M. Andriluka, U. Iqbal, E. Ensafutdinov, L. Pishchulin, A. Milan, J. Gall, and Schiele B. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018.

[3] Keni Bernardin, Alexander Elbs, and Rainer Stiefelhagen. Multiple object tracking performance metrics and evaluation in a smart room environment. 2006.

[4] François Fleuret, Horesh Ben Shitrit, and Pascal Fua. *Re-identification for Improved People Tracking*, pages 309–330. Springer London, London, 2014.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society.

[6] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[7] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. OpenPifPaf: Composite Fields for Semantic Keypoint Detection and Spatio-Temporal Association. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–14, March 2021.

[8] Chen Long, Ai Haizhou, Zhuang Zijie, and Shang Chong. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *ICME*, 2018.

[9] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[10] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129:3069–3087, 2021.

[11] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification:

A benchmark. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1116–1124, 2015.