An Analysis of Data Association Strategies for Multi-Object Tracking in Crowded Scenes

Jana Nikolovska
Master's Degree in Artificial Intelligence, University of Bologna
jana.nikolovska@studio.unibo.it

February 12, 2026

Abstract

This project analyzes the impact of different data association strategies for multi-object tracking on the DanceTrack dataset, a challenging benchmark characterized by crowded motion and visually similar targets. A tracking-by-detection pipeline is evaluated using three progressively complex configurations: (1) an IoU-only baseline (B0), (2) a ByteTrack-style confidence-aware tracker (B1), and (3) an appearance-augmented tracker with re-identification features (B2). All trackers share identical YOLOv8n detections to isolate the effect of association mechanisms.

Detector evaluation is performed independently, and an IoU threshold of $\tau = 0.25$ is selected based on F1 performance. Experimental results show that the largest performance gain comes from improving the association strategy (B0 to B1), significantly reducing identity switches and improving IDF1. In contrast, adding appearance-based re-identification (B2) yields only marginal or inconsistent improvements. The findings suggest that, for DanceTrack, robust association logic plays a more critical role than appearance cues in maintaining identity consistency in crowded scenes.

Introduction

Multi-object tracking

Multi-object tracking (MOT) is the problem of detecting multiple objects in a video sequence and maintaining their identities consistently over time. The goal is not only to localize objects in each frame, but also to correctly associate detections across frames despite motion, occlusions, and appearance changes.

Multi-object tracking is used in many real-world applications such as video surveillance, autonomous driving, robotics, and sports analysis, where it is important to follow multiple objects over time. The problem became more popular with the introduction of tracking-by-detection methods in 2009, notably with the work of Breitenstein et al., who proposed detecting objects independently in each frame and then linking these detections across frames to form object trajectories.

Today, most tracking systems use both motion information and visual appearance to decide which objects belong to the same track. However, tracking is still difficult in crowded scenes, where objects often overlap, look similar to each other, and move unpredictably, leading to lost tracks or identity errors.

Multi-object tracking on DanceTrack

To be more precise, in this project, we will try to solve the problem of multi-object tracking in the domain of dancing videos using the DanceTrack dataset. This domain is particularly challenging due to frequent and prolonged overlaps between people, similar clothing and visual appearance, and fast, non-linear movements. These factors make it difficult to reliably associate detections across frames and often lead to identity switches or fragmented tracks.

The purpose of this project is to compare how different data association strategies perform in this challenging setting. In particular, we study the difference between motion-only association and association based on visual appearance using re-identification features. The project aims to better understand the strengths and limitations of common tracking approaches when applied to complex and highly dynamic videos.

Model Architecture

This project evaluates three multi-object tracking configurations, referred to as B0, B1, and B2.

All configurations share a common detection backbone and tracking framework, differing only in their data association strategy.

Object Detector

All models employ YOLOv8n as the object detector. The detector produces bounding boxes and confidence scores for each frame and provides them to the tracking pipeline. YOLOv8n is selected due to its favorable trade-off between accuracy and computational efficiency, making it suitable for near real-time multi-object tracking applications. In our experiments, the pretrained model already achieved sufficiently strong detection performance on the DanceTrack dataset. For this reason, and to keep the focus of the study on data association strategies rather than detection performance, we did not fine-tune the detector on the target dataset.

Tracking Framework

Tracking is performed using a tracking-by-detection paradigm. For each frame, the following steps are executed:

- Objects are detected using YOLOv8n.
- Existing tracks are propagated using motion information.
- Detections are associated with existing tracks using a matching strategy that varies by model configuration.
- Unmatched detections initialize new tracks, while unmatched tracks are terminated after a predefined age.

Model B0: Motion-Only Tracking

The B0 configuration represents the baseline tracking approach without appearance-based cues. Data association relies exclusively on geometric overlap (IoU) between predicted track positions and current detections with no ReID features extracted. As a result, track matching is intentionally conservative, prioritizing correct associations over long-term continuity. While this reduces the risk of incorrect matches, it often prevents identities from being maintained through occlusions or rapid motion. Consequently, this approach is expected to achieve high precision but low recall, as tracks are more likely to be terminated than incorrectly associated.

Tracks are updated using a single-stage greedy IoU-based matching strategy, and unmatched tracks are removed after a fixed age. This design choice further reinforces the emphasis on precision at the expense of sustained identity tracking.

This configuration serves as a lower-bound reference for evaluating the importance of appearance information in crowded and dynamic scenes.

Model B1: ByteTrack-Based IoU Tracking

The B1 configuration extends the baseline by adopting a ByteTrack-style data association strategy. Rather than relying on a single IoU-based matching stage, detections are partitioned into high- and low-confidence sets according to their detection scores, following the approach proposed by Zhang *et al.* in ByteTrack (Zhang et al. 2022) .

Data association is performed in two stages. First, existing tracks are matched to high-confidence detections using an IoU-based criterion. Subsequently, unmatched tracks are associated with low-confidence detections using a more permissive IoU threshold. This two-stage strategy allows additional associations to be recovered when detection confidence degrades due to factors such as motion blur or partial occlusions.

As a result, B1 is designed to improve recall by preserving track continuity in challenging scenarios, while still maintaining strong precision compared to more aggressive matching strategies.

Model B2: Tracking with Re-Identification

The B2 configuration further extends B1 by incorporating appearance-based re-identification (ReID) to handle cases where IoU-based matching fails. Each detection is embedded into a feature space using a ResNet-50 backbone pretrained on ImageNet, employed as a generic feature extractor. The final classification layer is removed, and the network outputs a 2048-dimensional embedding for each detected bounding box. Cropped image regions are resized using ROIAlign and normalized using ImageNet statistics before feature extraction. The ReID backbone is used in evaluation mode without additional fine-tuning on the target dataset.

Data association in B2 follows the two-stage ByteTrack IoU matching strategy used in B1, operating on high- and low-confidence detections. After this IoU-based association, ReID is applied only to tracks that remain unmatched for a predefined number of frames. In this final stage, cosine similarity between appearance embeddings is used to reassociate tracks and detections, subject to a fixed similarity threshold.

Unlike B1, which relies exclusively on motion and detection confidence, B2 can recover associations across longer occlusions or abrupt motion by leveraging appearance information. This design is intended to improve recall and long-term identity continuity, at the potential cost of increased identity switches depending on the chosen ReID similarity threshold.

Figure 1: Example frames from the DanceTrack dataset, showing crowded dance scenes with frequent occlusions and visually similar targets, adapted from the original dataset publication (Sun et al. 2022).

Model Comparison

Table 1 summarizes the main differences between the three tracking configurations, highlighting their data association strategies and expected behavior.

Experimental Setup

Dataset

Experiments are conducted on the DanceTrack dataset, a large-scale benchmark for multi-object tracking in videos of group dancing with uniform appearance and complex motion (Sun et al. 2022). DanceTrack contains 100 annotated sequences divided into training, validation, and test splits, and is characterized by frequent occlusions, highly similar visual appearances among targets, and diverse movement patterns that challenge association models (Fig. 1). The dataset has its own project website (`https://dancetrack.github.io/`) and official GitHub repository (`https://github.com/DanceTrack/DanceTrack/`).

In this work, we use only the training split of the DanceTrack dataset for evaluation and analysis. This split contains

Aspect	B0	B1	B2
Association paradigm	IoU-only	ByteTrack (IoU + scores)	ByteTrack + ReID
Number of matching stages	1	2	3
Detection confidence used	No	Yes	Yes
High/low score separation	No	Yes	Yes
Primary association cue	IoU	IoU	IoU + appearance
Appearance features (ReID)	No	No	Yes
Similarity metric	–	–	Cosine similarity

Table 1: Comparison of the three tracking configurations. B0 uses a single-stage IoU-based association. B1 adopts a ByteTrack-style two-stage IoU matching strategy that incorporates detection confidence to improve recall. B2 further extends B1 by adding an appearance-based re-identification stage to recover associations that cannot be resolved using IoU alone.

40 annotated sequences, and we limit each sequence is limited to a maximum of 1000 frames as we are limited in computational resources

Evaluation Metrics

Detector Evaluation The detector is evaluated independently from the tracking module to ensure that the tracking results are not limited by poor detection quality. Evaluation is performed at the frame level using IoU-based matching, without considering object identities over time. For this purpose the metrics **Precision**, **Recall**, and **F1 score** were used.

Multi-Object Tracking Evaluation Tracking performance is evaluated using the MOTChallenge evaluation framework, implemented through the `motmetrics` library. The framework performs frame-wise IoU-based matching while preserving object identities across time, enabling the evaluation of both localization accuracy and identity consistency.

- **Recall**: Reflects how well the tracker maintains coverage of ground-truth trajectories over time. The values range from 0 to 1, where higher values indicate better tracking coverage. Very low recall suggests frequent track termination or failure to re-associate objects after occlusions.
- **IDF1**: Measures how consistently object identities are preserved across frames. The values range from 0 to 1, with higher values indicating more stable identity assignment.
- **Identity Switches**: Count the number of times a tracked identity is incorrectly reassigned. This metric is unbounded and sequence-dependent, with lower values always being better. A small number of identity switches indicates reliable data association, while large values suggest unstable identity assignment, especially in crowded scenes.
- **Fragmentation**: Measures how often a ground-truth trajectory is interrupted during tracking. This metric is also unbounded, and lower values indicate more temporally stable tracks. High fragmentation typically results from missed detections or overly conservative association strategies.

Implementation Details

All experiments were developed and executed using Google Colab, leveraging an NVIDIA A100 GPU for acceleration.

Execution Setup Running the detector for every tracker configuration would require repeatedly computing YOLO detections on the same sequences. Due to limited computational resources, this would significantly increase execution time.

To avoid redundant computation, we first run the YOLO detector once on all sequences and save the resulting detections to disk. These precomputed detections are then reused by all tracker variants during evaluation. This ensures consistent detector inputs across experiments while substantially reducing overall runtime.

Tracker Parameters and Configuration All tracking configurations use the same object detector and detection post-processing settings, including fixed detection confidence threshold parameter. The trackers differ only in their data association strategy and the associated hyperparameters, which control the trade-off between recall, identity stability, and fragmentation.

B0: Motion-Only Tracking The B0 configuration relies exclusively on IoU-based association. The parameter we try to tune is:

- **IoU matching threshold**: controls the minimum spatial overlap required to associate a detection with an existing track. Higher values enforce stricter matching, favoring precision, while lower values allow more permissive associations.
- **Maximum track age**: defines the number of consecutive frames a track can remain unmatched before being removed. Larger values allow tracks to survive short occlusions, while smaller values lead to faster termination.

Other tunable parameters that are not the focus of this project include **minimum hits**.

B1: ByteTrack-Based IoU Tracking The B1 configuration extends B0 using a ByteTrack-style two-stage association strategy based on detection confidence. In addition to the B0 parameters, the following tunable parameters are introduced:

- **High confidence threshold**: separates reliable detections from uncertain ones. Increasing this threshold reduces false positives but may reduce recall.
- **Low confidence threshold**: defines the minimum confidence for detections considered in the second association stage. Lower values increase recall by recovering difficult detections.
- **High-stage IoU threshold**: IoU threshold used when matching tracks to high-confidence detections, enforcing conservative associations.
- **Low-stage IoU threshold**: more permissive IoU threshold applied when matching unmatched tracks to low-confidence detections, improving track continuity.

B2: ByteTrack with Re-Identification The B2 configuration further incorporates appearance-based re-identification to recover associations that cannot be resolved using motion cues alone. In addition to all B1 parameters, B2 introduces:

- **ReID age**: minimum number of consecutive unmatched frames before appearance-based matching is attempted. Larger values restrict ReID usage to longer occlusions.
- **ReID similarity threshold**: cosine similarity threshold used to accept or reject ReID-based matches. Stricter thresholds favor identity stability, while more permissive thresholds increase recall at the risk of identity switches.

Due to limited computational resources, a full grid search was not performed. Therefore, only a limited set of configurations was tested to analyze different association behaviors.

Runtime Evaluation Runtime performance is measured on full-length sequences.

Results

Detector Performance

We evaluated the YOLO detector in a detector-only setting (no tracking), using per-frame matching between predicted and ground-truth bounding boxes. The base detection confidence threshold was initially set to 0.1 to operate in a high-recall regime. For the tracking experiments, the confidence threshold was increased to 0.25 to reduce noise and false positives; however, some tracker configurations internally override this value during processing depending on their association strategy.

With a confidence threshold of 0.1 and = 0.25, the detector achieves an F1 score of 0.749, with precision of 0.771 and recall of 0.727, indicating a good balance between detection coverage and localization accuracy.

This configuration serves as the baseline detection setting for all subsequent tracking experiments to ensure fair comparison across tracker variants. A qualitative comparison between tracker predictions and ground truth annotations is shown in Fig. 2.

Tracking Performance Comparison

The performance of each model under the selected configurations is presented in Tables 3–8, together with the corresponding configuration details and association parameters.

Run	Prec.	Rec.	MOTA	IDF1	IDs	Frags
B0_permissive	0.879	0.591	0.484	0.258	8977	15576
B0_balanced	0.879	0.591	0.483	0.243	9123	15580
B0_conservative	0.879	0.591	0.460	0.177	17337	15605

Table 2: DanceTrack performance of B0 tracker variants (IoU evaluation threshold = 0.5).

Run	Configuration Summary
B0_permissive	IoU=0.2, max_age=60
B0_balanced	IoU=0.3, max_age=30
B0_conservative	IoU=0.6, max_age=15

Table 3: Configuration parameters for B0 tracker variants. IoU denotes the global association threshold.

Trade-off Between Recall and Identity Consistency Recall and identity consistency often conflict in multi-object tracking. High recall encourages the tracker to detect and follow as many targets as possible, but this increases the risk of identity switches when detections from different individuals are incorrectly associated. In contrast, prioritizing identity consistency leads to more conservative associations, which reduces identity switches but may cause tracks to terminate early, lowering recall. A tracker with high recall but poor identity consistency may follow targets continuously while frequently assigning incorrect identities. Conversely, a tracker that emphasizes identity consistency may lose tracks more often but maintains correct identities while tracks remain active.

The preferred behavior depends on the application. Tasks that require detecting as many targets as possible may favor higher recall, even at the cost of identity errors. In contrast, applications such as behavior analysis or long-term person tracking benefit more from stable identities than from maximum recall.

B0 Results The B0 variants achieve high recall but perform poorly in terms of identity consistency, exhibiting a very large number of identity switches and track fragmentations. Among the B0 configurations, the permissive variant performs best; however, its IDF1 score remains significantly lower than those of higher-level models. These results indicate that relying primarily on IoU-based matching leads to frequent identity errors in crowded scenes. This behavior is particularly evident in the DanceTrack dataset, where individuals often move close together and experience frequent overlap.

B1 Results

The B1 variants show a clear improvement in identity consistency compared to B0, reflected by higher IDF1 scores and fewer identity switches. The long-occlusion configuration performs best, suggesting that maintaining tracks during short detection gaps is beneficial. However, more aggressive recovery strategies increase identity switches, highlighting

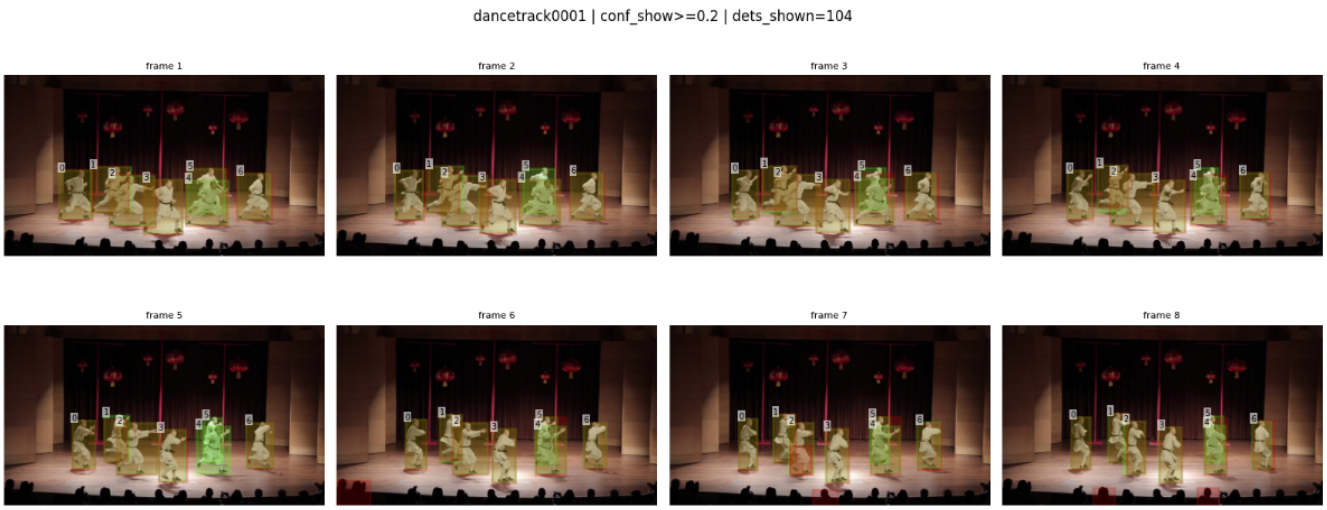


Figure 2: Qualitative comparison between YOLO detector predictions at IoU threshold $\tau = 0.25$ and ground truth annotations on sequence `DanceTrack0001`. The detector achieves a good balance between precision and recall while maintaining stable detections in crowded scenes.

Run	Prec.	Rec.	MOTA	IDF1	IDs	Frags
B1_crowd_safe	0.945	0.549	0.506	0.299	3623	12521
B1_long_occlusion	0.929	0.571	0.513	0.323	4924	14085
B1_recall_recovery	0.919	0.578	0.512	0.303	5290	14518
B1_conservative_ID	0.946	0.540	0.499	0.319	3440	12212
B1_balanced	0.934	0.567	0.515	0.318	4256	13733

Table 4: DanceTrack results for B1 tracker variants (IoU evaluation threshold = 0.5). IDs = identity switches; Frags = track fragmentations.

Run	high	low	IoU$_h$	IoU$_l$	max_age	match IoU
B1_crowd_safe	0.65	0.10	0.35	0.20	15	0.30
B1_long_occlusion	0.60	0.10	0.30	0.20	60	0.30
B1_recall_recovery	0.50	0.05	0.25	0.15	30	0.30
B1_conservative_ID	0.70	0.10	0.35	0.25	30	0.30
B1_balanced	0.60	0.10	0.30	0.20	30	0.30

Table 5: Key association parameters for B1 tracker variants.

the trade-off between recall and identity accuracy. Overall, the B1 results demonstrate that improved association logic alone can substantially enhance tracking performance.

B2 Results

Incorporating appearance-based re-identification in B2 does not consistently improve performance relative to B1. Conservative use of appearance information results in stable behavior, whereas aggressive re-identification increases both identity switches and fragmentations. This suggests that appearance features are not always reliable in crowded DanceTrack scenes, likely because many individuals have similar visual appearances and are frequently partially occluded.

Run	Prec.	Rec.	MOTA	IDF1	IDs	Frags
B2_crowd_safe	0.948	0.540	0.500	0.307	3704	12295
B2_long_occlusion	0.923	0.575	0.511	0.318	5593	14455
B2_aggressive_recovery	0.915	0.581	0.509	0.314	6138	14798
B2_conservative	0.939	0.558	0.510	0.317	4143	13276
B2_balanced	0.929	0.572	0.514	0.314	5088	14075

Table 6: DanceTrack results for B2 tracker variants (IoU eval threshold = 0.5). IDs = identity switches; Frags = track fragmentations.

Run	high	low	IoU$_h$	IoU$_l$	max_age	reid_age	cos_thr	match_IoU
B2_crowd_safe	0.70	0.10	0.35	0.25	20	3	0.25	0.30
B2_long_occlusion	0.60	0.10	0.30	0.20	60	10	0.32	0.30
B2_aggressive_recovery	0.55	0.05	0.25	0.15	45	8	0.38	0.30
B2_conservative	0.65	0.10	0.35	0.20	30	5	0.22	0.30
B2_balanced	0.60	0.10	0.30	0.20	30	5	0.30	0.30

Table 7: Key configuration parameters for each B2 run.

Comparison Across B0, B1, and B2

The largest performance gain is observed when moving from B0 to B1, particularly in terms of identity consistency. The transition from B1 to B2 yields only minor or inconsistent improvements. These results indicate that association strategy plays a more critical role than appearance cues for this dataset. Overall, the findings suggest that the primary challenge of DanceTrack lies in crowded motion patterns rather than insufficient appearance information.

Runtime Analysis

Runtime is reported separately for YOLO detection and for the tracking module, since detections are precomputed and tracking runs independently. The YOLO runtime reflects the cost of generating detections per frame, while the tracker runtime reflects only the association stage. For the tracker,

Component	ms / frame
YOLO Detection	17.78
B0 Tracker (avg.)	6.41
B1 Tracker (avg.)	6.44
B2 Tracker (avg.)	8.74

Table 8: Average runtime per frame for detection and tracking. Tracker values are averaged across configurations.

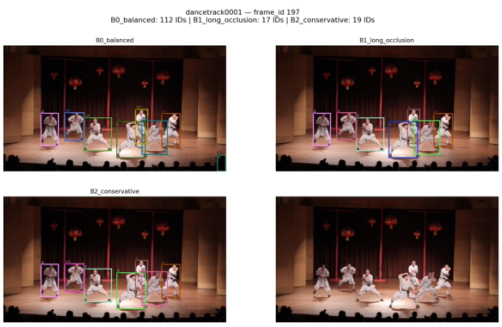


Figure 3: Short descriptive caption of the image.

we report the average runtime across all B0, B1, and B2 configurations (see Table 8).

If detection and tracking are combined, a rough estimation of the end-to-end runtime per frame is 40FPS

Qualitative Comparison via Parallel Visualization

We generate side-by-side visualization videos to qualitatively compare B0, B1, and B2 on identical frames. Each bounding box is color-coded by identity and annotated with its track ID, enabling direct inspection of identity stability and switches.

An example screenshot is shown in Fig. 3.

Ablation study

Effect of Appearance-Based Re-Identification

We ablate the contribution of appearance-based re-identification in B2 by varying how strongly embeddings are used during association (conservative vs. aggressive). For selected frames, we crop up to K detections, extract embeddings, and compute a pairwise cosine similarity matrix (including the same IDs from the previous/next frame when available). Figure 4 illustrates two examples: the crops and the corresponding similarity matrices. In crowded DanceTrack scenes, different individuals often produce high similarity values due to similar clothing and partial occlusions, which explains why aggressive ReID increases identity switches and fragmentations, while conservative usage is more stable (Tables 6–8).

Conclusion

The results show that B1 clearly improves over B0, especially in terms of identity consistency. While B0 achieves high recall, it produces many identity switches, showing that simple IoU-based matching is not reliable in crowded scenes. B1 reduces these errors by using a better association strategy, making it more stable overall.

B2 does not consistently improve over B1. In some cases, appearance-based re-identification helps, but in others it increases identity switches, especially when used aggressively. This suggests that appearance information is not always reliable for the DanceTrack dataset. Overall, configuration changes within each model affect performance only slightly, while the main improvements come from moving between model levels.

Overall, multi-object tracking on DanceTrack requires more advanced approaches to handle frequent occlusions and visually similar targets. Recent state-of-the-art methods use stronger appearance models, temporal context, or joint detection-and-tracking frameworks to better preserve identities in crowded scenes.

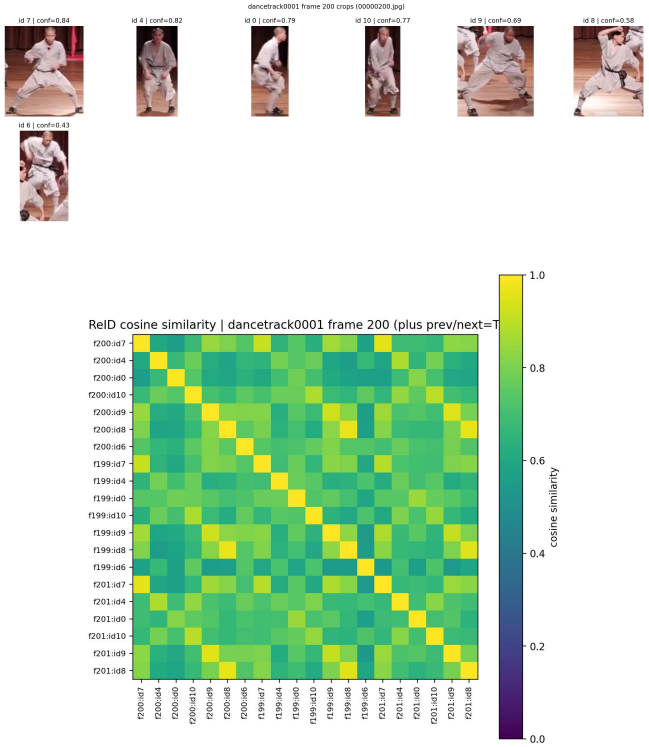
Links to external resources

- link to a GitHub repository containing the code (`https://github.com/jananikolovska/mot_dance_track`)

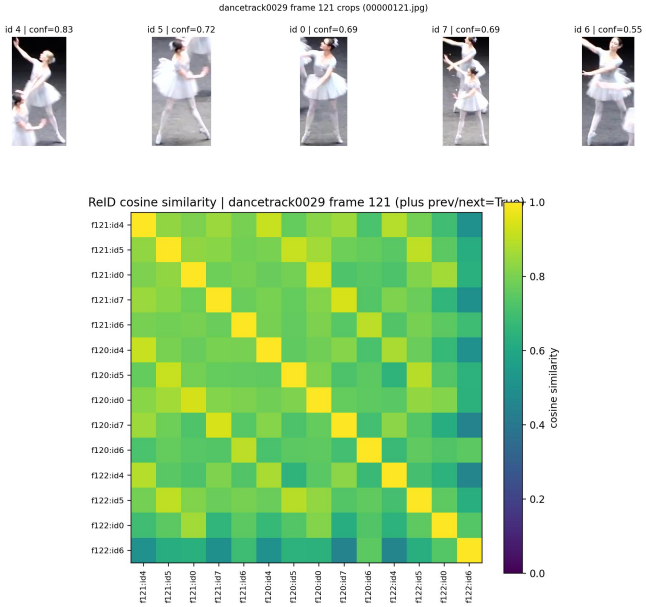
References

Sun, P.; Cao, J.; Jiang, Y.; Zhang, Y.; and Luo, P. 2022. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; and Wang, X. 2022. Bytetrack: Multi-object tracking by associating every detection box. In *Proceedings of the European Conference on Computer Vision (ECCV)*.



(a) Example 1 (frame t): detection crops and cosine similarity matrix.



(b) Example 2 (frame t): detection crops and cosine similarity matrix.

Figure 4: ReID debugging examples shown in parallel. For each example, detection crops are displayed above the corresponding cosine similarity matrix. High off-diagonal similarities indicate limited appearance discriminability in crowded scenes.