SmartSquare – Motion Paths Extraction

HCU | HafenCity Universität Hamburg

CityScienceLab | HafenCity University Hamburg

Visualize and Analyze

- **Input:** Tracks with geographical coordinates
- **Task:** Visualize the tracks on maps, compute and visualize indicators on Dashboards, analyze tracks and regions
- The observed plaza has been divided into tiles
- **Counting, statistics and indicators:** computed on basis of:
  - Frequency over time
  - Exposure time per track, per tile and over time
  - Change over time
- **CityScopes:** e
  - coll
- **Analytics:** Visualize tracks, statistics and indicators
  - Load anonymized and geo-referenced trajectories into our Data Analytics Pipeline (see our other poster) to analyze and visualize motion paths over time
  - Find and track pattern in motion paths
  - Provides diagrams, figures and maps
  - Shows change over time

Map to Real World

- **Input:** Tracks over all frames resp. over time
  - with pixel coordinates
- **Task:** Transform the pixel-coordinates from all detections and camera perspectives into geographical coordinates
- Thin plate splines (TPS) is a spline-based technique for data interpolation and smoothing
- Based on: "Principal Warps: Thin-Plate splines and the decomposition of deformations" from F.L. Bookstein in IEEE Transaction on Pattern Analysis and Machine Intelligence (Vol. 11, No. 6, June 1989)

Transformation Approach:

Step 1: Transform images to georeferenced point-clouds (geoinformation-matrices) through landmark definition based on thin-plate splines warping

a. Compute thin-plate splines
b. Warping
c. Save as geoinformation-matrix

Step 2: Transform the result of the detection algorithms (pixel-coordinates for each detection) to geographical coordinates via former generated (step 1) geoinformation-matrices

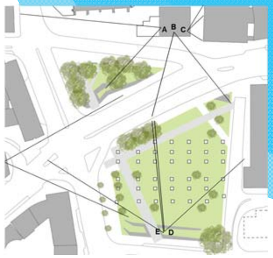
- geoinformation-matrices has to be generated for all camera perspectives

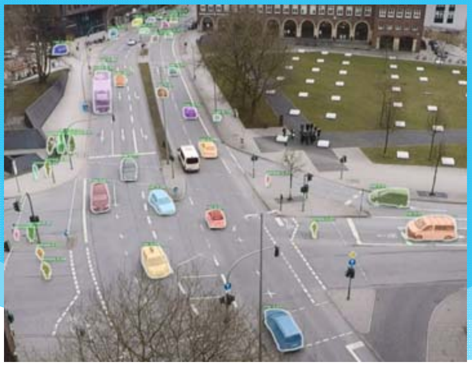
Track Paths

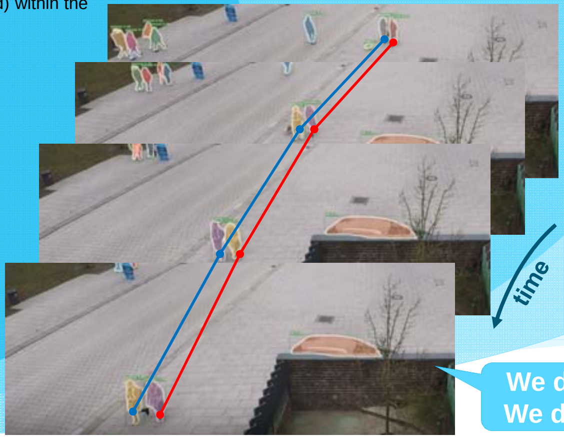
- **Input:** Frames and its detections per frame
- **Task:** Associate the detections of former frames with the detections of later frames in order to generate tracks of pedestrians and vehicles movements over time
  - Amongst others we are using Deep SORT (https://github.com/nwojke/deep_sort) as Simple Online and Realtime Tracking algorithm utilizing a Deep Association Metric
  - Rudimentary combination of familiar techniques such as the Kalman Filter (state estimation) and Hungarian algorithm (data association) for the tracking components (SORT)
  - Extended to integrate appearance information based on a deep appearance descriptor (Deep SORT)
- **Multi-camera approach:** Tracking results from all camera perspectives have to be merged by connecting its tracks (if pedestrians move from one camera perspective to another)
  - Multi-cameras are not used to observe the same scene from different perspectives but rather to observe a bigger space (means that cameras angles of view do not overlap much)

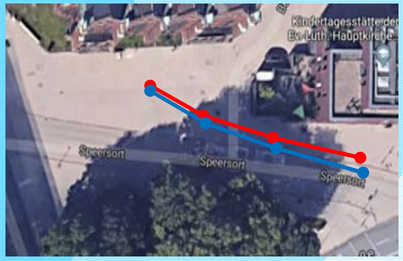
Detect Pedestrians

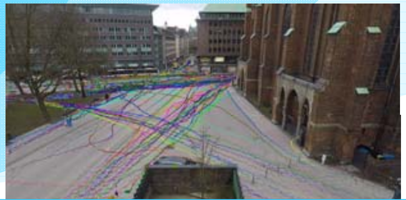
- **Input:** Frames
  - Records from 6 different fixed camera perspectives (Offline approach; using records from 3 weeks, 3 recorded days each and around 5 hours of recorded HD videos per camera and day ~ around 270 hours of video recordings)
  - From 1 permanent mounted "intelligent camera" module (NVIDIA Jetson TX2 with FLIR camera) which does the detection job on-the-fly and just sends anonymized detection results (privacy-by-design) and features to a server where the tracking and mapping tasks can be performed server side and in real-time (bigger hardware)
- We are using Detectron (https://github.com/facebookresearch/Detectron) and its Mask R-CNN (https://arxiv.org/abs/1703.06870) implementation for detection and segmentation as well as YOLOv3 (https://pjreddie.com/darknet/yolo/) and SSD (https://github.com/weiliu89/caffe/tree/ssd) within the "intelligent camera" module
- Instant segmentation and masking enables instant anonymization (privacy-by-design)

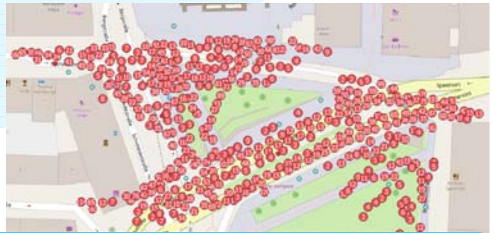












We do not track or save information or images about or from individuals!
We do extract anonymized motion paths in order to explore the square …

June 2018 - Marc-André Vollstedt