LineIT: Similarity Search and Recommendations for Photo Lineup Assembling

Ladislav Peška\textsuperscript{a} and Hana Trojanová\textsuperscript{b}

\textsuperscript{a} Department of Software Engineering, Charles University, Prague, Czech Republic
\textsuperscript{b} Department of Psychology, Charles University, Prague, Czech Republic
Police Photo Lineup

- **Eyewitness identification** of the suspect / offender during criminal proceedings
- Most common case: select suspect among other persons (fillers)
- Either „in natura“ or based on photographs

Randomly positioned suspect
Three to seven fillers
Witnesses are asked to state the suspect’s number or that the suspect is not present
Errors in Photo Lineup

1) Observe the event

2) Identify the offender
   - among other candidates
Errors in Photo Lineup

1) Observe the event.
   - numerous sources of error (poor lighting, large distance, short time, fear, fatigue…)
   - affects memory and recognition processes
   - decrease witness certainty and reliability

2) Identify the offender
   - decreased certainty may lead to incorrect identification
   - and conviction of innocent persons

   Witnesses may follow the „best available choice“ selection

   Possibly large time-span

Real offender might not be in the lineup
Assembling Fair Photo Lineups

- Lineup process should be set to *eliminate* testimony of *uncertain* witnesses
  - *Don’t go to jail just because you’re (e.g.) big & black*
- Double-blind administration
- *Fair (unbiased) assembling of lineups*
  - Out of the dataset of candidates, lineup administrator should select fillers *similar* to the suspect
    - Currently, only feature-based filtering systems are available
    - Better automatization in content processing is necessary

- Use deep learning techniques (DCNN) and principles of similarity search & recommender systems to simplify the task of assembling fair lineups
Early experiments: Item-based Recommendations

Recommend top-k candidates for each suspect

- **CB-RS**, cosine similarity of candidates’ and suspect’s explicit CB features
  - Nationality, Age, Appearance features
  - *Baseline (mimic feature-based filters)*
- **Visual-RS**, cosine similarity of candidates’ and suspect’s visual features (last fully connected layer of a pre-trained neural network)
  - VGG-Face\(^1\) network

*Let domain experts select the relevant candidates*

Early experiments: Results

- Visual recommendations outperform CB ones *(58% vs 37% of selections)*
  - However, both seems relevant up to some extent
  - Very small intersection of candidates recommended by Visual and CB
    - *Some RS aggregations needed.*

- *Low level of agreement among participants on the selected candidates*
  - Potential for personalized recommendation / long-term preferences?

- Less diversity is better (!)
  - Participants agreed on the need for providing homogeneous lineups
  - *Dynamical recommendation based on the selected candidates?*

- No seemingly too similar candidates
  - *Dataset-dependent, however probably common in real-world settings*

Peska, Trojanova: Towards Recommender Systems for Police Photo Lineup; DLRS (RecSys) 2017.
Early experiments: Follow-up

Evaluation of:
- Combined Visual+CB recommendations
- Expert-assembled lineups (with only CB filtering available)

- **Automatically assembled lineups are in average as good as expert-based**

**However, both automated & expert-based solutions often do not meet the criteria for unbiased lineups**
- „Mock-witnesses“ too often select the suspect based only on a short textual description

**→ Combined „human-in-the-loop“ approach needed**

Peska, Trojanova: Towards Similarity Models in Police Photo Lineup Assembling Tasks; SISAP 2018.
LineIT Tool

- Information Retrieval tool tailored for lineup assembling tasks
- Based mainly on:
  - Recommendations (Visual & CB aggregation with lineup uniformity constraints)
  - Similarity search (Query-by-multiple-examples & regions of interest selection)
  - (CB filtering)

- Currently proof-of-concept tool serving for user studies & research
  - Evaluation of developed IR model, utilized ML algorithms & GUI
  - Currently static dataset, no authentication, no authorization etc.
  - Python/Flask + jQuery
LineIT Tool: Front-end

A) Attribute-based filtering

B) QBE search

Query examples

Local search weight

Selected ROIs (local search)

C) Search results

D) Current lineup

Selected candidates

Suspect

E) Recommended candidates
LineIT Tool: Recommendations

- Recommendations upon a change in assembled lineup
  - Similarity to both suspect (high weight) and already selected candidates is aggregated
  - CB and Visually similar candidates evaluated separately (two lists)
  - Aggregate recommendations via Fuzzy D’Hondt’s election algorithm

\[
\sum w_m \times sim_v(c_m, c_i) \quad \text{Visual similarity}
\]

\[
\sum w_m \times sim_{CB}(c_m, c_i) \quad \text{CB similarity}
\]

- Visual vs. CB votes learned on-line w.r.t. candidates selection
- CB attribute weights adjusted on-line w.r.t. its utilization in filtering

² Peska, Balcar: Fuzzy D’Hondt’s Algorithm for On-line Recommendations Aggregation; ORSUM (RecSys) 2019
LineIT Tool: Search Engine

- Query by visual similarity of selected examples
- Pre-filtering via explicit CB filters
- Optionally, select regions of interest (ROIs) within the image to focus similarity search
  - Evaluated against the same or nearby regions of other candidates
  - Bald, distinctive chin, protruding ears etc.
  - Important, if the witness mentioned such a feature
LineIT Tool: Evaluation

- Usage evaluation of individual GUI elements
  - Which of the GUI elements were mostly utilized?
  - Which lead to the acceptable results?

- Task: Assemble full lineup for pre-selected mock suspects

  - 11 participants, 75 assembled lineups, 371 selected candidates
  - 274 submitted search queries
  - Mean time to assemble lineup: 3 min
    - Compared to 10-20 minutes with only explicit CB filters

- Selected candidates from
  - Search results: 165 (44%)
  - Recommended: 206 (56%)
LineIT Tool: Evaluation

Which of the GUI elements were mostly utilized?

- CB conditions specified in 65%, QBE in 87% and local search in 52%
  - However, only a few queries with multiple examples (7% of queries)
  - Sole CB filtering had low convergence (13% queries vs. 7% selections)
  - Otherwise, no observed differences between utilized vs. successful strategies

Volumes of elements set within a query
LineIT Tool: Evaluation

Which of the GUI elements were mostly utilized?

- Local queries mostly focused on the center of the face
- Recommended candidates were more utilized in the later stages, i.e., users gained trust into recommendations over time
Conclusions & Future Work

- Prototype tool providing *human-in-the-loop* IR support for lineups assembling
- **Selected IR GUI** seems capable enough & simplifies the task
  - However, iterative improvements & research necessary (background ML models, RS tuning, **improved GUI** etc.)
  - Mock witness based evaluation of created lineups in the future

- Better datasets needed – perhaps data augmentation (e.g., GAN) can help?
- **Psychology-related:**
  - Effect of ethnicity? Prejudices? Is there a limit for too uniform lineups?

- **Develop ready-to-use software (in progress)**
Thank you!

Questions, comments?

Would you like to participate in evaluations? Just e-mail us!

peska@ksi.mff.cuni.cz