Detecting and Naming Actors in Movies using Generative Appearance Models
Vineet Gandhi, and Remi Ronfard
vineet.gandhi@inria.fr, remi.ronfard@inria.fr

INTRODUCTION
We introduce a generative model for learning person and costume specific detectors from labeled examples. We demonstrate the model on the task of localizing and naming actors in one long movie \cite{3}, sampled every 10 seconds.

Contributions
• A complete framework to learn view-independent actor models using MSCR \cite{1} features with a novel clustering algorithm
• Two stage detection framework, a search space reduction using the k-nearest neighbours and a sliding window search for the best localization of the actor in position and scale

GENERATIVE MODEL
• Head AND Shoulder representation, each as a constellation of optional MSCR regions
• Arbitrarily complex actor models
• Each Actor associated with visual vocabulary of cluster centers C_i and respective frequencies H_i
• C_i is represented in a 9 dimensional space (position, color, size and shape)

Formally, our generative model for each actor consists of the following three steps:
1. Choose screen location and window size for the actor on the screen, using the detections in the previous frame as a prior
2. Choose visible features C_i in the "head" and "shoulder" regions independently, each with a probability H_i
3. For all visible features C_i, generate color blob B_i from a gaussian distribution with mean C_i and covariance \Sigma_i, then translate and scale to the chosen screen location and size

P(B_i, m_{ij}, a) = \sum_i H_i m_{ij} \exp \left\{-\frac{(C_i - B_i)^T \Sigma_i^{-1} (C_i - B_i)}{2} \right\}

Overview of Detection Process

TRAINING ACTOR MODELS
• Constrained agglomerative clustering
• Each blob in the first training image is initialized as a singleton cluster. We then compute pairwise matching with set of blobs from other training images (one by one)
• At each step, for each cluster, we assign at most one more blob. Blobs not assigned to an existing cluster are assigned to their own singleton cluster

EXPERIMENTAL RESULTS
Precision-recall curves compared to a state of the art generic detector \cite{2} on a movie with 8 actors sampled every 10 seconds.

Detection using sliding window search
Each actor detection score is based on the likelihood that the image in the sliding window was generated by the given actors model.

Given
• Actors Models (C^n, \Sigma^n, H^n) and image features B
for each actor a do
• for each scale s do
  Normalize image features w.r.t scale.
  [IDX, D_T] = kNN-SEARCH(B, C^n, k)
  Build inverted index i.e. for each unique blob B' in the Kn
  refined set, store corresponding clusters in C^n and respective
  distances using IDX and D_T.
  for each position (x, y) do
    Find blob indices J_{head} and J_{shoulders}
    Compute m_{ij} using blob indices and inverted indices
    score(x, y, s, a) = \sum_{k \in J(x, y)} P(B_i, m_{ij}, a)
  end for
end for
end for

[\hat{x}(a), \hat{y}(a), \hat{a}(a)] = \arg \max \sum_a (score(x, y, s, a) - t_0)

Conclusions and Future Work
• View-independent models from few training examples
• Simultaneous detection and identification
• Fast matching using kNN-search
• 80% recall with 90% precision in both detection and identification
• Good candidate for tracking-by-detection and floor-plan view reconstruction

References
\cite{1} Per-Eric Forsen. Maximally Stable Colour Regions for Recognition and Matching In CVPR’07, 2007.
\cite{3} Alfred Hitchcock. Rope. Warner Brothers, 1948.