on learned visual embedding

patrick pérez

Allegro Workshop
Inria Rhônes-Alpes
22 July 2015
Vector visual representation

- **Fixed-size image representation** $\mathbf{x} \in \mathbb{R}^D$
  - High-dim (100 $\sim$ 100,000)
  - Generic, unsupervised: BoW, FV, VLAD / DBM, SAE
  - Generic, supervised: learned aggregators / CNN activations
  - Class-specific, e.g. for faces: landmark-related SIFT, HoG, LBP, FV

- **Key to “compare” images** and fragments, with built-in invariance
  - Verification (1-to-1)
  - Search (1-to-N)
  - Clustering (N-to-N)
  - Recognition (1-to-K)
VLAD: vector of locally aggregated descriptors

- \( C \) SIFT-like blocks, \( D = 128 \times C \)

\[
x = (x_1, x_2, \ldots, x_C)
\]

[Jégou et al. CVPR’10]
Face representation

- **Sparse representation**
  - Layout of facial landmarks
  - Multi-scale descriptor of facial landmarks

- **Dense representation**
  - Fixed grid of overlapping blocks
  - SIFT/HOG/LBP block description
  - Fisher and CNN variants
  - Landmarks still useful to normalize

\[ x \in \mathbb{R}^{9 \times 3 \times 128} \]

- e.g., [Cinbis et al. ICCV’11]

\[ x \in \mathbb{R}^{9 \times 9 \times 4 \times 6} \]

- e.g., [Sivic et al. ICCV’09]
Embedding visual representation

- **Further encoding** $\phi(x) \in M$ to
  - Reduce complexity and memory
  - Improve discriminative power
  - Specialize to specific tasks

- **Various types** (possibly combined)
  - Discrete (Hamming, VQ, PQ):
    \[ M = \{c_1, \ldots, c_K\}, \quad K = 2^B \]
  - Linear (PCA, **metric learning**):
    \[ M = \mathbb{R}^E, \quad E < D \]
  - Non-linear (**K-PCA**, spectral, NMF, SC):
    \[ M \subset \mathbb{R}^E \]
Explicit embedding for visual search
[JMIV 2015, with A. Bourrier, H. Jégou, F. Perronin and R. Gribonval]

E-SVM encoding for visual search (and classification)
[CVPR 2015, with J. Zepeda]

Multiple metric learning for face verification
[ACCV 2014, CVPR-w 2015, with G. Sharma and F. Jurie]
Euclidean (approximate) search

- Nearest neighbor (1NN) search in \( x = \{x_n\}_{n=1}^N \subset \mathbb{R}^D \)
  
  \[
  \arg \min_{x \in X} d(q, x) \text{ or } \arg \max_{x \in X} s(q, x)
  \]

- Euclidean case
  
  \[
  \arg \min_{x \in X} \|q - x\|_2^2 \text{ or } \arg \max_{x \in X} \langle q, x \rangle
  \]

- Euclidean approximate NN (a-NN) for large scale
  - Discrete embedding efficient to search with: binary hashing or VQ
  - Product Quantization (PQ) [Jégou 2010]: asymmetric fine grain search

\[
\phi(x) = [\phi_1(x_1), \cdots, \phi_R(x_R)], \quad \phi_r : \mathbb{R}^{D/R} \mapsto \mathbb{M}_r \subset \mathbb{R}^{D/R}
\]

\( B = R \times B_s \) bits code with sub-quantizers on \( 2^{B_s} \) values

\[
\arg \min_{x \in X} \sum_{r=1}^R \|q_r - \phi_r(x_r)\|_2^2
\]

\( D \times 2^{B_s} \) distances and \((R - 1) \times N\) sums for search
Beyond Euclidean

- **Other (di)similarities**
  - $\chi^2$ and histogram intersection (HI) kernels
  - Data-driven kernels
  - Appealing but costly

- **Fast approximate search with Mercer kernels?**
  - Exploiting of kernel trick to transport techniques to implicit space
  - Inspiration from classification with *explicit embedding*
    - [Vedaldi and Zisserman, CVPR’10][Perronnin et al. CVPR’10]
The implicit path

- **Kernelized Locality Sensitive Hashing (KLSH)**
  [Kulis and Grauman ICCV’09]
  - Random draw of directions within RKHS subspace spanned by implicit maps of a random subset of input vectors
  - Hashing function computed thanks to kernel trick

- **Random Maximum Margin Hashing (RMMH)**
  [Joly and Buisson CVPR’11]
  - Each hashing function is a kernel SVM learned on a random subset of input vectors (one half labeled +1, the other -1)

\[ h(x) = \text{sign}\left( \sum_{m=1}^{M} y_m \alpha_m K(x, z_m) + b \right) \]

- Outperforms KLSH
Explicit embedding

- **Data-independent**
  - Truncated expansions or Fourier sampling
  - Restricted to certain kernels (e.g., additive, multiplicative)

- **Generic data-driven**: Kernel PCA (KPCA) and the like
  - Mercer kernel $K$ to capture similarity
  - Learning subset $\mathcal{Z} = \{z_1 \cdots z_M\}$
  - Low-rank approximation of kernel matrix $K = [K(z_i, z_j)] \succeq 0$

\[
K = U \Lambda U^\top \approx U_E \Lambda_E U_E^\top, \quad D < E \ll M
\]

\[
\phi(z_m) = \Lambda^\frac{1}{2} E U_E^\top
\]

\[
\phi(x) = \Lambda^\frac{-1}{2} E U_E^\top k, \quad k = [K(x, z_m)]_{m=1}^M
\]

\[
\phi_e(x) = \lambda e^\frac{-1}{2} \langle u_e, k \rangle = \lambda e^\frac{-1}{2} \sum_{m=1}^M K(x, z_m) \phi_e(z_m)
\]
NN and a-NN search with KPCA

- **Exact search**
  - KPCA encoding
  - Exact Euclidean 1NN search
  - Bound computation
  - Most similar item is in short list truncated with bounds

- **Approximate search**
  - KPCA encoding
  - Euclidean a-kNN search with PQ
  - Similarity re-ranking of short list
Experiments

- **1NN local descriptors search**

  - $N=1M$ SIFT ($D=128$), $K=\chi^2$, $M=1024$, $E=128$,
  - Tested also: KPCA+LSH (binary search in explicit space)
Experiments

- **1NN image search**
  - $N=1.2M$ images BoW ($D=1000$), $K=\chi^2$, $M=1024$, $E=128$
  - Tested also: KPCA+LSH (binary search in explicit space)
Discriminative encoding with E-SVM

- **Boost discriminative power of representation**
  - Extract what is “unique” about image (representation) relative to all others

- **Method**
  - Exemplar-SVM (E-SVM) [Malisiewicz 2012] to encode visual representation
  - Symmetrical encoding even for asymmetric problems
  - Recursive encoding

- **Application**: search and classification
Method

- Large “generic” set of images $\mathcal{Z} = \{z_m\}_{m=1}^M \subset \mathbb{R}^D$

- Exemplar-SVM
  $$\tilde{w} = \arg \min_{w \in \mathbb{R}^D} \left[ \frac{1}{2} \|w\|_2^2 + \alpha_+ \max(0, 1 - x^T w) + \alpha_- \sum_{m=1}^M \max(0, 1 + z_m^T w) \right]$$

- Final encoding
  $$\phi(x; \mathcal{Z}) = \frac{\tilde{w}}{\|\tilde{w}\|_2}$$
Method

- **E-SVM learning**: stochastic gradient (SGD) with Pegasos
- **Recursive encoding (RE-SVM)**
  
  \[
  \begin{align*}
  \mathbf{w}^{(1)} &= \phi(\mathbf{x}, \mathbf{z}) \\
  \mathbf{z}^{(1)} &= \phi(\mathbf{z}, \mathbf{z}) \\
  \mathbf{w}^{(k+1)} &= \phi(\mathbf{w}^{(k)}, \mathbf{z}^{(k)}) \\
  \mathbf{z}^{(k+1)} &= \phi(\mathbf{z}^{(k)}, \mathbf{z}^{(k)})
  \end{align*}
  \]

- **Image search**: symmetrical embedding
  - Query and database codes: \( \mathbf{w}_0 \) and \( \{\mathbf{w}_n\}_{n=1}^N \)
  - Cosine similarity: \( \langle \mathbf{w}_0, \mathbf{w}_n \rangle \)

- **Classification**: learn and run classifier on E-SVM codes
Image search

- *Holiday* dataset, VLAD-64 ($D=8192$)

\[
\begin{align*}
M &= 60,000 \\
T &= 10^5
\end{align*}
\]
Image search

- Holiday and Oxford datasets

<table>
<thead>
<tr>
<th>Method</th>
<th>Holidays</th>
<th>Oxford</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD-64 [1]</td>
<td>72.7</td>
<td>46.3</td>
</tr>
<tr>
<td>VLAD-64 + RE-SVM-1</td>
<td>77.5</td>
<td>55.5</td>
</tr>
<tr>
<td>VLAD-64 + RE-SVM-2</td>
<td>78.3</td>
<td>57.5</td>
</tr>
<tr>
<td>CNN [2]</td>
<td>68.2</td>
<td>40.6</td>
</tr>
<tr>
<td>CNN [2] + RE-SVM-2</td>
<td>71.8</td>
<td>44.6</td>
</tr>
</tbody>
</table>

---

47 → 4

92 → 29
Face verification

- Given 2 face images: **Same person?**
  - Persons unseen before

- **Various types of supervision for learning**
  - Named faces (provide +/− pairs)
  - Tracked faces (provide + pairs)
  - Simultaneous faces (provide − pairs)

- **Labelled Faces in the Wild (LFW)**
  - +13,000 faces; +4,000 persons
  - 10-fold testing with 300 +/− pairs per fold
  - Restricted setting: only pair information for training
  - Unrestricted setting: name information for training
Linear metric learning

- **Powerful approach** to face verification
- **Learning Mahalanobis** distance in input space $\mathbb{R}^D$, via $M \succeq 0$
  \[
  d_M^2(x, x') = (x - x')^\top M (x - x')
  \]
- **Typical training data**: $\mathcal{T} = \{(x, x', y_{xx'})\} \subset \mathbb{R}^{2D} \times \{-1, +1\}$
  - +/- pairs should become close/distant
- **Verification of new faces**: $y_{xx'} = \text{sign}(1 - d_M^2(x, x'))$
- **Several approaches**
  - Large margin nearest neighbor (LMNN) [Weinberger et al. NIPS’05]
  - Information theoretic metric learning (ITML) [Davis et al. ICML’07]
  - Logistic Discriminant Metric Learning (LDML) [Guillaumin et al. ICCV’09]
  - Pairwise Constrained Component Analysis (PCCA) [Mignon & Jurie, CVPR’12]
Low-rank metric learning

- **Very high dimension** (in range $1,000 \sim 100,000$)
  - Prohibitive size of Mahalanobis matrix
  - Scarcity of training data
- **Low-rank Mahalanobis metric** learning: $M = L^\top L, \ L \in \mathbb{R}^{E \times D}, \ E \ll D$

  $$d^2_L(x, x') = (x - x')^\top M (x - x')$$
  $$= ||Lx - Lx'||_2^2$$

- Learn linear projection (dim. reduction) and metric
- **Minimize loss** over training set

$$\min_{L,b} \sum_{(x, x', y_{xx'}) \in T} \text{loss}[d^2_L(x, x'), y_{xx'}; b]$$

- Rank fixed by cross-validation
- **Proposed**: extension to **latent variables** and **multiple metrics**
# Losses

- **Probabilistic logistic loss**

\[ 1 + y_{xx'} \tanh\left[-\frac{1}{2}(d_L^2(x, x') - b)\right] \]

- **Generalized logistic loss**

\[ \frac{1}{\beta} \log(1 + \exp[\beta y_{xx'}(d_L^2(x, x') - b)]) \]

- **Hinge loss**

\[ \max\left[0, 1 - y_{xx'}(b - d_L^2(x, x'))\right] \]
Expanded parts model

- Expanded parts model [Sharma et al. CVPR’13] for human attributes and object/action recog.

- Objectives
  - Avoid fixed layout
  - Learn collection of discriminative parts and associated metrics
  - Leverage the model to handle occlusions
Expanded parts model

- **Mine** $P$ **discriminative parts** and learn associated metrics $\mathcal{L} = \{L_p\}_{p=1}^P$
- **Dissimilarity** based on comparing $K < P$ best parts

\[
d^2_L(x, x') = \min_{\alpha \in \{0, 1\}^P} \sum_{p=1}^P \alpha_p \|L_p(x_p - x'_p)\|_2^2
\]

sb.t. $\|\alpha\|_0 = K$, and overlap($\alpha$) $< \theta$

- **Learning**
  - Minimize **hinge loss**: greedy on parts + gradient descent on matrices
  - Prune down to $P$ a large set of $N$ random parts
  - Projections initialized by whitened PCA
  - Stochastic gradient: given annotated pair $(x, x', y_{xx'})$

\[
\text{if } y_{xx'}(b - d^2_L(x, x')) < 1
\]

$\forall p \in \text{support of } \alpha^*$:

\[
\partial_{L_p} \text{loss} = y_{xx'} L_p(x_p - x'_p)(x_p - x'_p)^	op
\]
Experiments with occlusions

- **LFW, unrestricted setting**
  - \( N = 500, \ P \sim 50, \ K = 20, D = 10k, E = 20, 10^6 \) SGD iterations
  - Random occlusions (20 – 80%) at test time, on one image only

- **Focused occlusions**
Experiments with occlusions

<table>
<thead>
<tr>
<th></th>
<th>Left eye</th>
<th>Right eye</th>
<th>Both eyes</th>
<th>Nose</th>
<th>Mouth</th>
<th>Nose + mouth</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>75.5</td>
<td>73.4</td>
<td>61.7</td>
<td>78.0</td>
<td>77.3</td>
<td>73.5</td>
</tr>
<tr>
<td>EPML</td>
<td>78.9</td>
<td>77.0</td>
<td>69.2</td>
<td>79.1</td>
<td>78.5</td>
<td>75.5</td>
</tr>
</tbody>
</table>
Comparing face sets

- Given groups of single-person faces
  \[ x = \{x_n\}_{n=1}^{N} \subset \mathbb{R}^D \]
e.g., labelled clusters, face tracks

- Comparing sets
  - Based on face pair comparison, i.e.
    \[ D_L^2(x, x') = \min_{x \in X} \min_{x' \in X'} d_L^2(x, x') \]
  - For face tracks: a single descriptor per track [Parkhi et al. CVPR’ 14]

[Everingham et al. BMVC’06]
Learning multiple metrics

- Metrics associated to $L$ mined types of cross-pair variations

$$D_\mathcal{L}^2(x, x') = \min_{(\ell,p,q)} \| L_\ell(x_p - x'_q) \|_2^2$$

- Learning from annotated set pairs $\mathcal{T} = \{(x, x', y_{xx'})\}$

$$\min_{\mathcal{L}, b} \sum_{(x, x', y_{xx'}) \in \mathcal{T}} \text{loss}[D_\mathcal{L}^2(x, x', y_{xx'}); b]$$
Learning multiple metrics

- **Stochastic gradient**: given annotated pair \((x, x', y_{xx'})\)
  - Subsample the sets (to ensure variety of cross-pair variations)
  - Dissimilarity: \(D^2_{\mathcal{L}}(x, x') = \min_{(\ell, p, q)} \|L_\ell(x_p - x'_q)\|^2\)
    \[= \|L_{\ell^*}(x_{p^*} - x'_{q^*})\|^2\]

- Sub-gradient of pair’s **hinge** loss: if \(y_{xx'}(b - \|L_{\ell^*}(x_{p^*} - x'_{q^*})\|^2_2) < 1\)
  \[\partial L_{\ell^*} \text{loss} = y_{xx'}L_{\ell^*}(x_p - x'_q)(x_p - x'_q)^\top\]

- Projections initialized by whitened PCA computed on random subsets
New dataset

- From 8 different series (inc. Buffy, Dexter, MadMen, etc.)
- 400 high quality labelled face tracks, 23M faces, 94 actors
- Wide variety of poses, attributes, settings
- Ready for metric learning and test (700 pos., 7000 neg.)
Comparing face tracks

- Parameters: $D \sim 14000$, $K = 3$, $10^6$ SGD iterations

<table>
<thead>
<tr>
<th>Method</th>
<th>Subspace dim. $E$</th>
<th>Aver. Precision known persons</th>
<th>Aver. Precision unknown persons</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA+cosine sim + min-min</td>
<td>1000</td>
<td>24.8</td>
<td>20.4</td>
</tr>
<tr>
<td>PCA+cosine sim + min-min</td>
<td>100</td>
<td>21.4</td>
<td>20.2</td>
</tr>
<tr>
<td>Metric Learning + min-min</td>
<td>100</td>
<td>23.7</td>
<td>21.0</td>
</tr>
<tr>
<td>Latent ML (proposed)</td>
<td>(3X)33</td>
<td>27.9</td>
<td>22.9</td>
</tr>
</tbody>
</table>
Learn embedding of visual description

- Unsupervised learning of $\phi$
- Task-dependent supervised learning of $(\phi, f)$

Also for deep learning

- 1-layer adaptation of CNN features for classification with linear SVM
- Ad-hoc dim. reduction or learned with L1 regularization (Kulkarni et al. BMVC15)
- Same performance as VGG-M 128 [Chatfield 2014], with 4x smaller codes