Learning A Stroke-Based Representation for Fonts

E. Balashova\textsuperscript{1}, A. H Bermano\textsuperscript{1}, V. G. Kim\textsuperscript{2}, S. DiVerdi\textsuperscript{2}, A. Hertzmann\textsuperscript{2}, T. Funkhouser\textsuperscript{1}

\textsuperscript{1} - Princeton University
\textsuperscript{2} - Adobe Research
Goal: Stroke-Based Font Representation Suitable for Learning

Examples:

Stroke-Based Parametrization

Manifold Learning:
Motivation

[https://www.fontlab.com]
Motivation
Motivation
Desired Font Representation Characteristics
Desired Font Representation
Characteristics

• Detail-preserving
Desired Font Representation
Characteristics

• Structure-aware
Desired Font Representation Characteristics

- Scales without artifacts
Desired Font Representation
Characteristics

- Suitable for learning
Font Representations

- Raster - Based
- Contour - Based
Font Representations

Raster-Based

Limitations

• Scales with artifacts
• Not detail-preserving
• Not structure-aware
Font Representations

Contour-Based
Font Representations

Bezier-Based

Limitations

- Not suitable for learning
Font Representations

Consistent Outline-Based

Limitations

• Not structure-aware

[Campbell '14]
Font Representations

Proposed Representation

Characteristics
- Detail-preserving
- Structure-aware
- Scales without artifacts
- Suitable for learning

Part-Aware
Skeleton-Based Representation
Our Approach

Examples:

Consistency:
Our Approach

Template Fitting

Examples:

Consistency:

Template Definition

\( \theta_t \)

\( (1,1,1,0,1,0) \)

Connectivity Constraints

\( O_a^+ - O_a^+ - O_a^- - O_a^- \ldots \)
Our Approach

Template Fitting

Examples:

<table>
<thead>
<tr>
<th>a</th>
<th>a</th>
<th>a</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

Consistency:

Template Definition

<table>
<thead>
<tr>
<th>a</th>
<th>a</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>
Our Approach

Template Fitting

Examples:

aaaa
bbb
cccc

Consistency:

Skeleton Optimization

Initial Skeleton
Our Approach

Template Fitting

Examples:

Consistency:

Skeleton Optimization

Initial Skeleton

Registration
Our Approach

Template Fitting

Consistency:

Skeleton Optimization

Examples:

Initial Skeleton

Registration

Initial Segmentation

E. Balashova et al. / Learning A Stroke-Based Representation for Fonts

Given a collection of fonts, we set the template skeleton using Coherent Point Drift method. To define the unary term, we use CRF-based segmentation to obtain the final consistent segmentation of the skeleton to the template, and the pairwise term favors cuts at skeleton segmentations - points where more than two skeleton paths meet. Both terms are smoothed with the aforementioned Gaussian kernel. Sharp features and intersections are optimized:

\[ \min_{\mathbf{s}, \mathbf{c}} \sum_{u} \mu_u(x_u) + \sum_{u} \sum_{l} \delta_{l}(x_u) \]

where \( \mu_u(x_u) \) is the unary term, \( \delta_l(x_u) \) is the pairwise term, and \( x_u \) are the points of the skeleton. The unary term is defined as:

\[ \mu_u(x_u) = \lambda_u \cdot \text{dist}(x_u, \text{template})^2 \]

where \( \lambda_u \) is a weight and \( \text{dist}(x_u, \text{template}) \) is the average distance between the curve and the closest point on the outline of the template. The pairwise term is defined as:

\[ \delta_l(x_u) = \lambda_l \cdot \mathcal{A}(x_u, x_{u+1}) \]

where \( \lambda_l \) is a weight and \( \mathcal{A}(x_u, x_{u+1}) \) is a penalty for a pair of adjacent points. To enforce learning correlations in the data, we set the latent dimension to be lower-dimensional, i.e., experimental to work best for our setup. This representation will implicitly adhere to common design principles in font design.
Our Approach

Template Fitting

Examples:

```
| aaaa | bbbb | cccc |
```

Consistency:

Skeleton Optimization

- Initial Skeleton
- Registration
- Initial Segmentation
- Final Segmentation
Our Approach

Outline Optimization

\[ E = E_{\text{corr}} + E_{\text{ft}} \]
Our Approach

Examples:

Template Fitting

Consistency:

Outline Optimization

\[ E = E_{\text{corr}} + E_{\text{ft}} \]

\[ E_{\text{corr}}(G_{c,f}, \Theta) = \sum_{x \in P(G_{c,f})} h_{\sigma_{\text{corr}}} (D(x, O_c(\Theta))) + \sum_{y \in P(O_c(\Theta))} h_{\sigma_{\text{corr}}} (D(y, G_{c,f})) \]
Our Approach

Template Fitting

Examples:

Consistency:

Outline Optimization

\[ E = E_{\text{corr}} + E_{\text{ft}} \]

\[ E_{\text{corr}}(G_{c,f}, \Theta) = \sum_{x \in P(G_{c,f})} h_{\sigma_{\text{corr}}} (D(x, O_{c}(\Theta))) + \sum_{y \in P(O_{c}(\Theta))} h_{\sigma_{\text{corr}}} (D(y, G_{c,f})) \]

\[ x = [q_X, q_Y, w_n n_X(q), w_n n_Y(q)] \]
Our Approach

Template Fitting

Examples:

Outline Optimization

\[
E = E_{\text{corr}} + E_{\text{ft}}
\]

\[
E_{\text{ft}}(G_{c,f}, \Theta) = \sum_{x \in F(O_c(\Theta))} D_{\text{curve}}(x, F_{\tau_{\text{ft}}}(G_{c,f}))
\]
Our Approach

Examples:

Template Fitting

Consistency:

Outline Optimization

\[ E = E_{\text{corr}} + E_{\text{ft}} \]

\[ E_{\text{ft}}(G_{c,f}, \Theta) = \sum_{x \in F(O_c(\Theta))} D_{\text{curve}}(x, F_{\tau_{\text{ft}}}(G_{c,f})) \]

- \( E_{\text{corr}} \): Consistency term
- \( E_{\text{ft}} \): Template fitting term
- \( G_{c,f} \): GT outline
- \( \Theta \): Template parameters
- \( F \): Set of feature points
- \( D_{\text{curve}} \): Arc-length intrinsic distance
- \( \tau \): Subset of junction points near feature point
Our Approach

Template Fitting

Examples:

Consistency:

Outline Optimization

Input

Initial Width

Initial Guess

Only $E_{corr}$

With Normal Reg

With Feature Alignment
Our Approach

Examples: aaaa bbbb cccc

Consistency: aaaa bbbb cccc

Manifold Learning:
Our Approach

Examples: $\begin{array}{c}
aaaa \\
bbbb \\
cccc \\
\end{array}$

Consistency: $\begin{array}{c}
aaaa \\
bbbb \\
cccc \\
\end{array}$

Manifold Learning: $\begin{array}{c}
\text{image}
\end{array}$

$$f = \begin{bmatrix} f_{a_1} & f_{a_2} & \cdots \end{bmatrix}$$
Our Approach

EM-PCA
[Row98]

Input       Latent
Vectors     coordinates

\[ Y = CX + V \]

Transformation       Noise
Matrix
Our Approach

Examples:

Consistency:

Manifold Learning:

**EM-PCA**

[Row98]

Input Vectors

Latent coordinates

Transformation Matrix

Noise

\[ Y = CX + V \]

Learning Steps:

E-step: \[ X = \left( C^T C \right)^{-1} C^T Y \]

M-step: \[ C_{new} = YX^T (XX^T)^{-1} \]
Our Approach

Examples:

Consistency:

Manifold Learning:

Iterative Improvement:

Initial Fit  Manifold Guess  Improved Fit
Fitting Results
Fitting Results

[Graph showing error distribution with percentile on the y-axis and error on the x-axis. Different lines represent different data sets labeled a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z.]

0.3 px
Fitting Results

![Graph showing fitting results with percentiles and error (px)]

- Error: 0.52
- Error: 0.99
- Error: 1.96
- Error: 3.49
- Error: 8.54
Fitting Results

![Graph showing fitting results with percentile on the y-axis and error (px) on the x-axis. Each line represents different data sets with associated errors: a = 0.52, b = 0.99, c = 1.96, d = 3.49, e = 8.54.](image)
Iterative Improvement

Evaluation
Iterative Improvement
Evaluation

Initial Fit → Manifold Guess → Improved Fit
Iterative Improvement Evaluation

% of Glyphs Below Learning Threshold vs. Iteration

Start 1 2 3 4 5

0.58 0.6 0.62 0.64 0.66 0.68 0.7 0.72
Iterative Improvement Evaluation

![Graph showing mean error decreasing with iterations](graph.png)
Comparison to Non-Part-Aware Approach
Comparison to Non-Part-Aware Approach

Part-Based Representation
Comparison to Non-Part-Aware Approach

Part-Based Representation

Single Contour Representation [CK14]
Comparison to Non-Part-Aware Approach

Part-Based Representation

Raster-Based

Part-Based Representation

Raster-Based
Comparison to Non-Part-Aware Approach

Part-Based Representation

Single Contour Representation [CK14]

Raster-Based
Generative Model Comparison
Generative Model Comparison

EM-PCA

Denoising VAE

Vanilla VAE
Generative Model Comparison

EM-PCA

Denoising VAE
Applications
Style Completion
Style Completion: Light Fonts

MerriweatherSans-Light

Exo-Light
Style Completion: Regular/Bold Fonts
Style Completion: Italic Fonts

GT

Eval.

Input

Prediction

Nobile-BoldItalic

MinionPro-It

h a m b u r g e f o n

h a m b u r g e f o n

h a m b u r g e f o n

h a m b u r g e f o n
Style Completion:
Serif Fonts

Input

Prediction

Amethysta-Regular

Oldenburg-Regular
Style Completion: Serif Fonts
Style Completion
Style Completion

hand$z0vebcf1knparsstuwxz
Style Completion

hanszovebcfijklmpqrstuvwxyz
hanszovebcfijklmpqrstuvwxyz
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion

handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
handszovebcsftikmparsstuwxyz
Style Completion
Style Completion
Style Completion
# Style Completion

| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
| h | a | n | d | g | z | o | v | e | b | c | f | t | j | k | m | p | a | r | s | t | u | w | x | y | z |
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Style Completion
Font Retrieval
Font Retrieval

Query Font: Ubuntu-Light

Nearest Neighbor: Ubuntu

Farthest Font: Dosis-ExtraLight
Topology-Aware Retrieval

Query Font: Ubuntu-Light

Nearest Neighbor: Ubuntu

Farthest Font: Dosis-ExtraLight

Nearest Serifed Font: AnticSlab-Regular
Topology-Aware Retrieval

Query Font: Ubuntu-Light

Nearest Neighbor: Ubuntu

Farthest Font: Dosis-ExtraLight

Nearest Serifed Font: AnticSlab-Regular

Nearest Font With Topology 1 of a: Inder-Regular

Nearest Font With Topology 2 of a: Ubuntu
Limitations
Limitations
Limitations

[https://design.google]
Limitations

[https://inventingsituations.net]
Future Work
Future Work

[https://www.theatlantic.com]
Future Work

[https://www.123rf.com/]
Future Work

[Nejati '16]
Acknowledgments

• Princeton Graphics and Vision groups, Ariel Shamir, anonymous reviewers

• Princeton CS Staff (Asya Dvorkin)

• Support: National Science Foundation Graduate Fellowship (NSF GRFP), NSF CRI 1729971 and Adobe
Thank You!