# Lec8 Image Stylization

人工智能引论实践课 计算机视觉小班

主讲人: 刘家瑛

# 部分讲义来源与致谢


1. Leon A. Gatys, Alexander S. Ecker, Matthias Bethge. Image style transfer using convolutional neural networks. CVPR 2016.
2. Chuan Li, Michael Wand. Combining Markov random fields and convolutional neural networks for image synthesis. CVPR 2016.
3. Alex J. Champandard. Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artworks. arXiv 2016.
4. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. CVPR 2017.
5. Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. ICCV 2017.

# Image Style Transfer

- **Image stylization**
  - Enrich content
  - Intensify representation
  - Convey emotion
- **Application**
  - Social media
  - Graphic design


# Image Style Transfer

- **Text stylization**
  - Enrich content
  - Intensify representation

BBS进站图片

logo设计

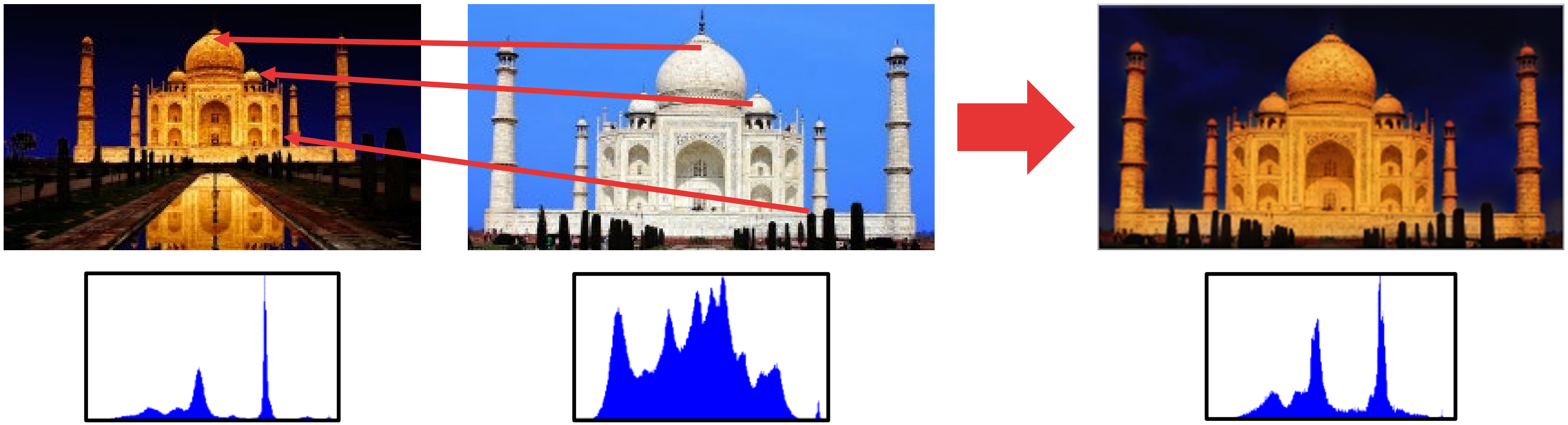
海报设计

游戏主视觉


纪念衫设计

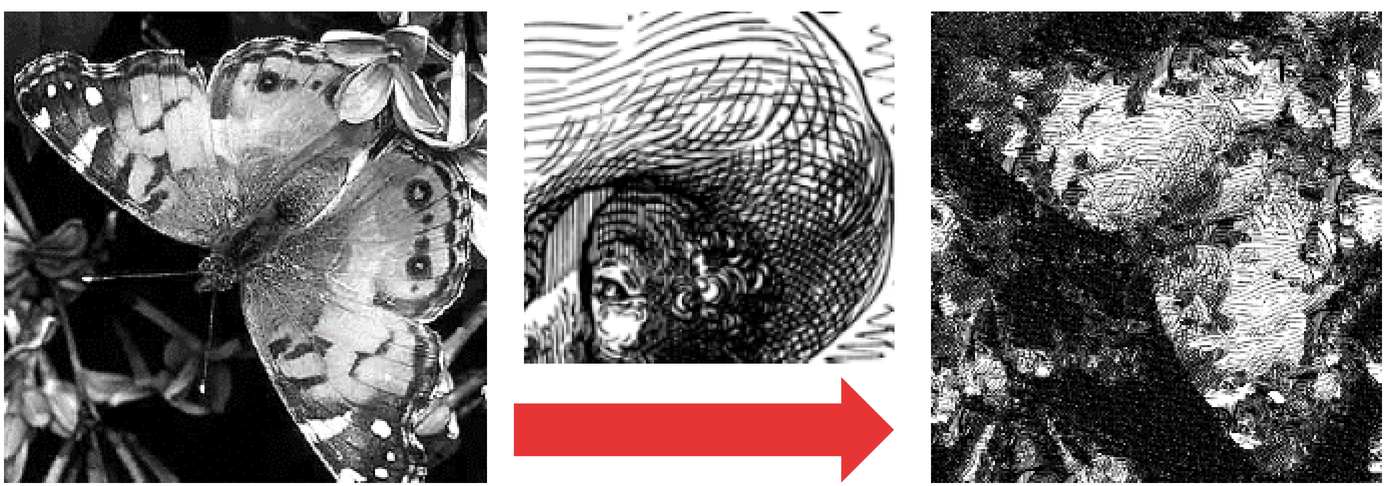
# Image Style Transfer

- Color transfer
- Texture transfer



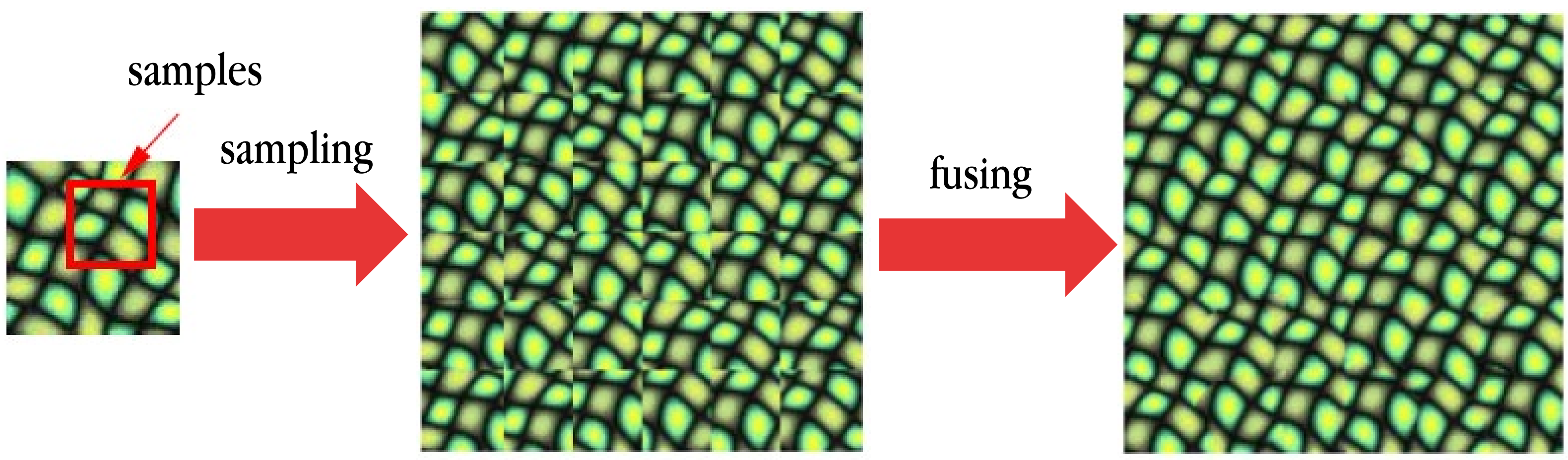
# Image Style Transfer

- Color transfer
- Texture transfer
  - Texture synthesis + structure constraint
- Texture synthesis
  - Non-parametric methods
  - Parametric methods



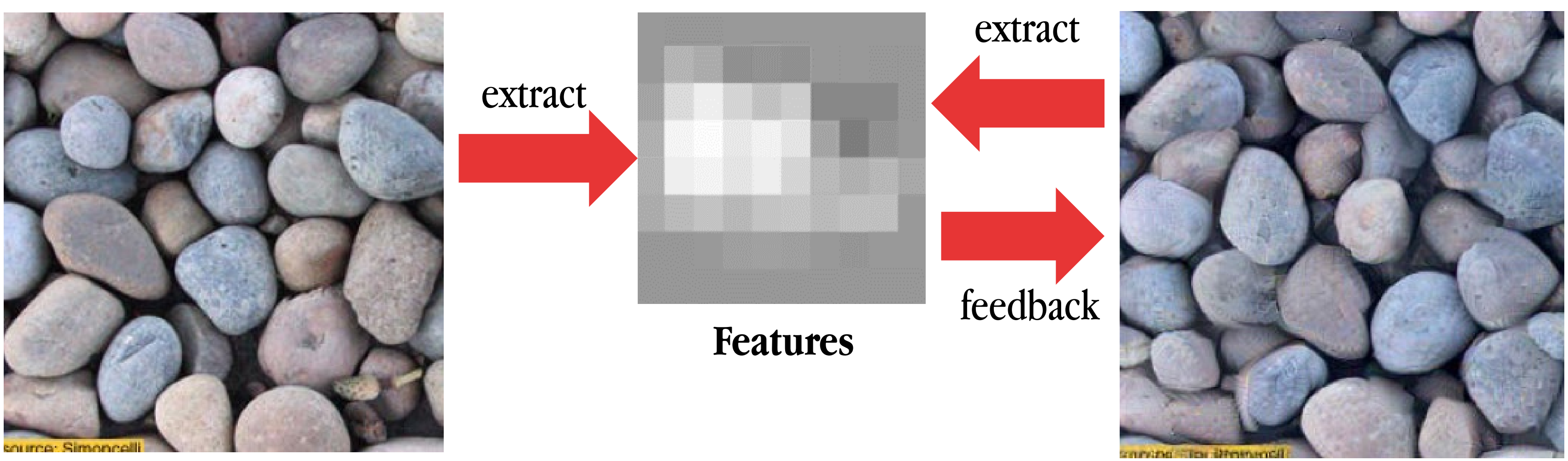
# Image Style Transfer

- Color transfer
- **Texture transfer**
  - Texture synthesis + structure constraint
- **Texture synthesis**
  - Non-parametric methods
  - Parametric methods



# Image Style Transfer

- Color transfer
- **Texture transfer**
  - Texture synthesis + structure constraint
- **Texture synthesis**
  - Non-parametric methods
  - Parametric methods

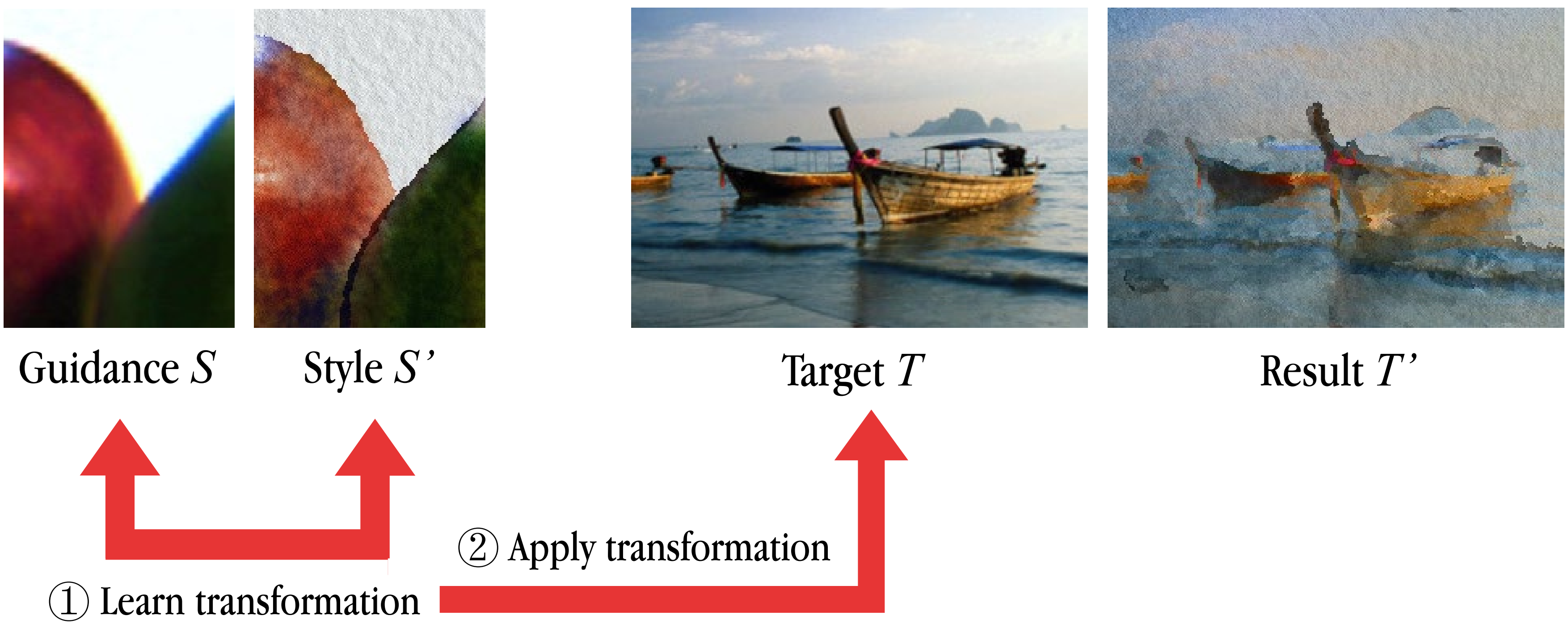


# Image Style Transfer

- Supervised method
- Unsupervised method

## Image Style Transfer

- **Supervised method**
- Unsupervised method



# Image Style Transfer

- Supervised method
- **Unsupervised method**

Style *S'*

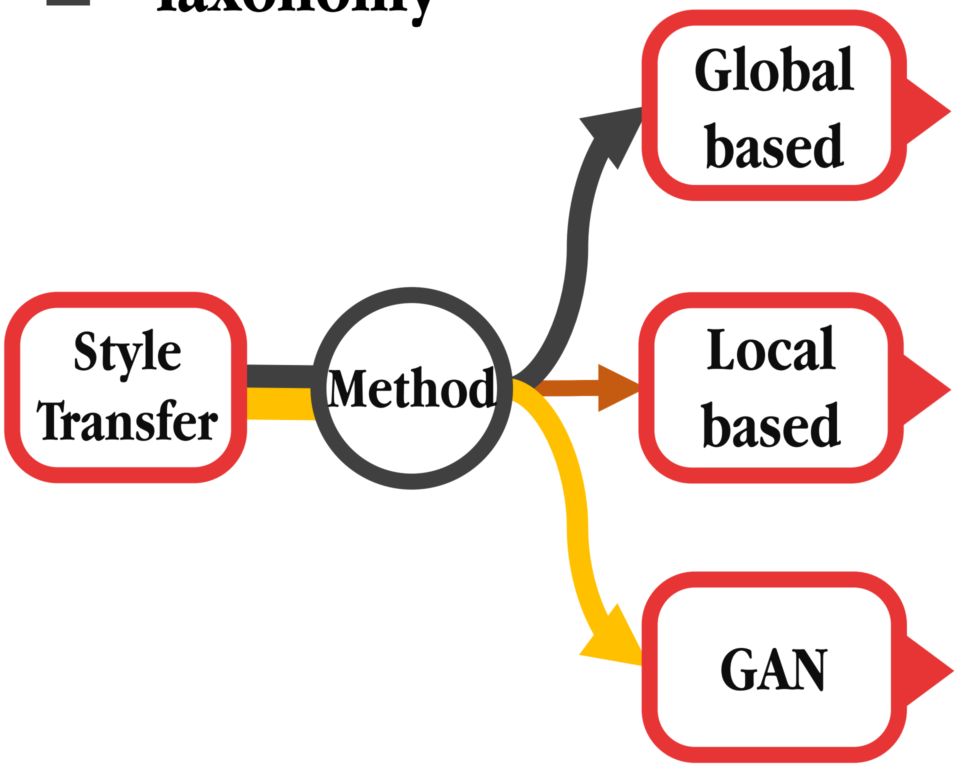
Target *T*

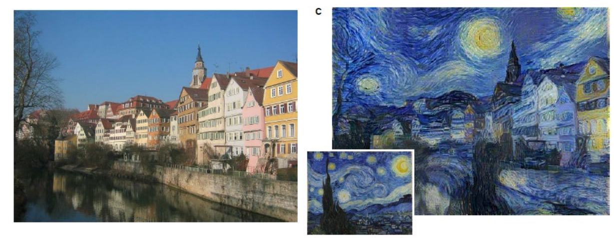
Result *T'*

Extract features to build mappings

# Image Style Transfer

## Taxonomy



Neural Style Transfer[1]

CNNMRF[2], Neural Doodle[3]

Pix2pix-cGAN, cycleGAN

[1]2016 CVPR. Image style transfer using convolutional neural networks

[2]2016 CVPR Combining Markov random fields and convolutional neural networks for image synthesis

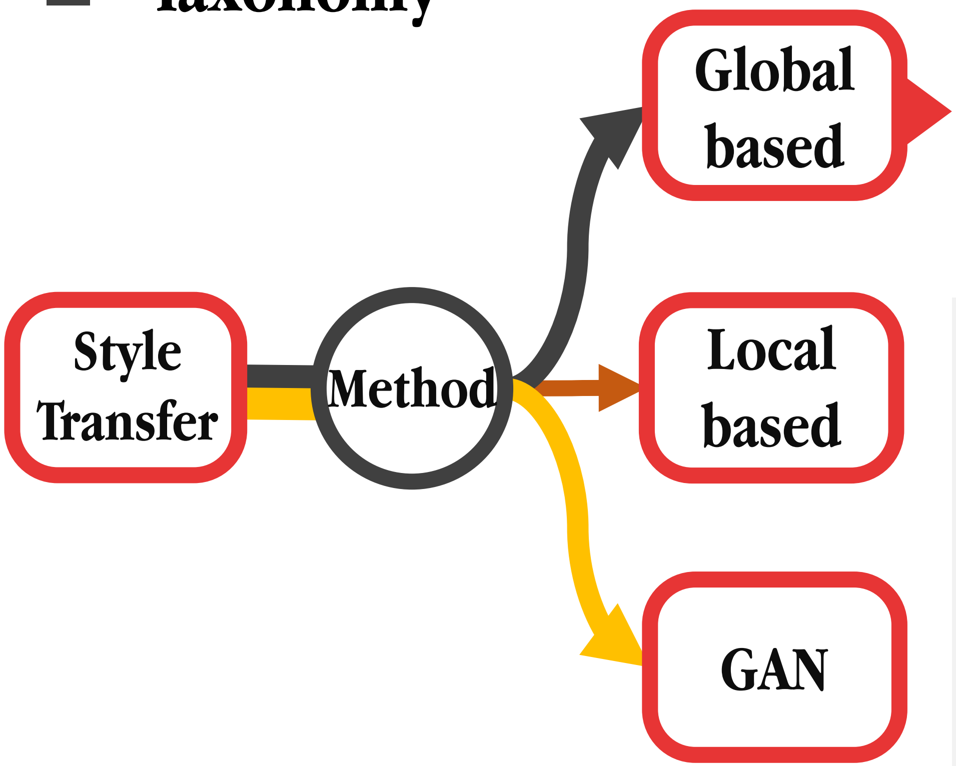
[3]2016 Arxiv Semantic style transfer and turning two-bit doodles into fine artworks

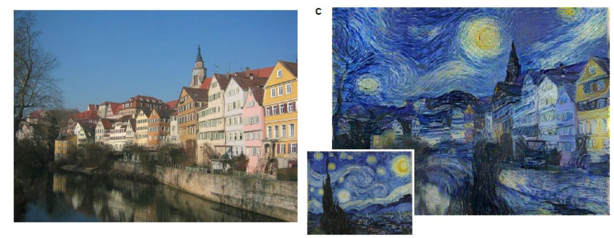
[4]2017 CVPR Image-to-Image Translation with Conditional Adversarial Networks

[5]2017 ICCV Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks

# Image Style Transfer

## Taxonomy



Neural Style Transfer[1]

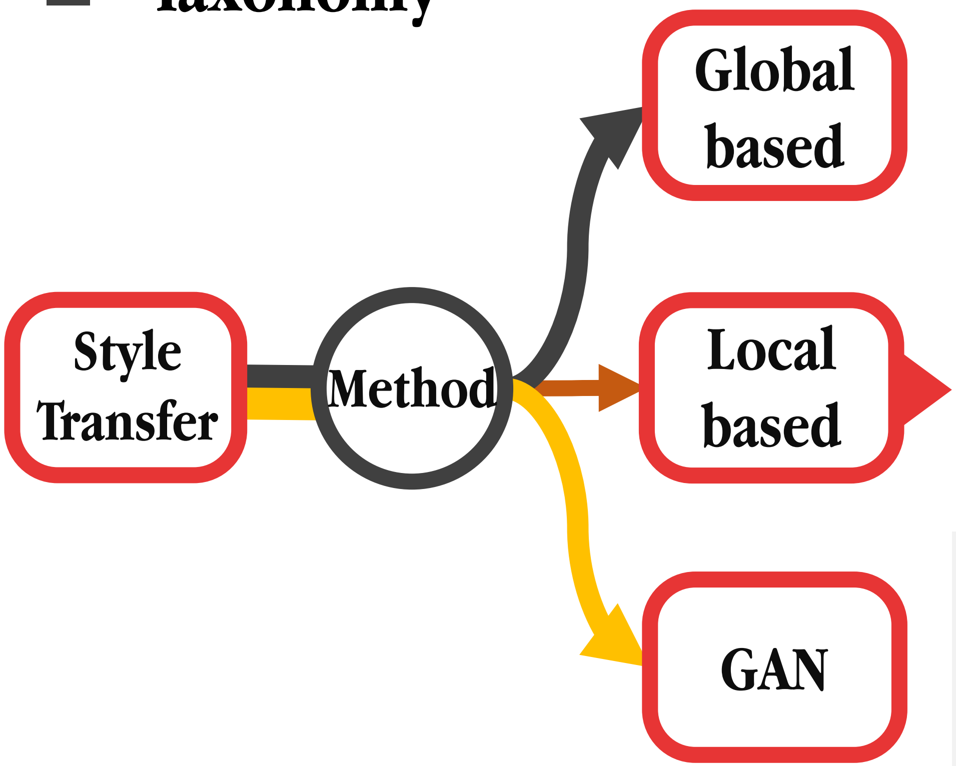
**Main idea:**
Match the global feature distributions between the style image and the target image

- Gram Matrix (Coherence)
- Mean & Variance
- Whitening & Coloring

Style as global distribution

# Image Style Transfer

## Taxonomy




CNNMRF[2], Neural Doodle[3]

**Main idea:**

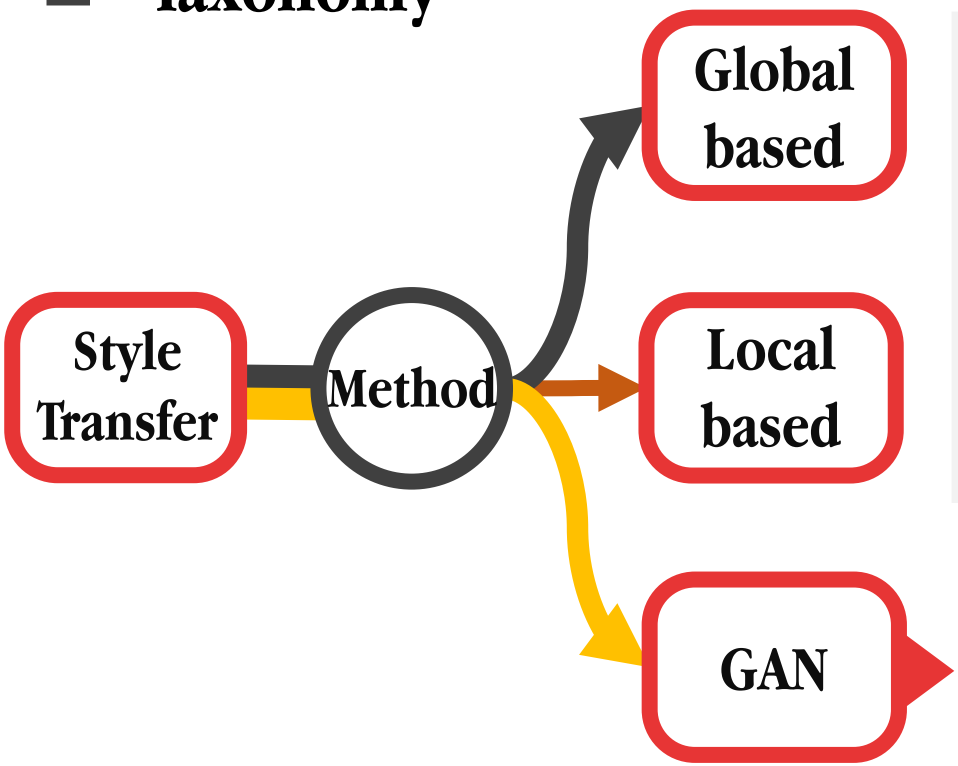
match local patches / pixels

- Patch/pixel in image domain
  - Image Analogy (SIGGRAPH01)
- Neural patches
  - CNNMRF
- Hybrid: Deep image analogy

Style as local patches

# Image Style Transfer

## Taxonomy




**Main idea:**

Learn mapping between two domains
Instead of defining what style is,
Learn to classify the style

- Paired
  - Pix2pix-cGAN
- Unpaired
  - cycleGAN

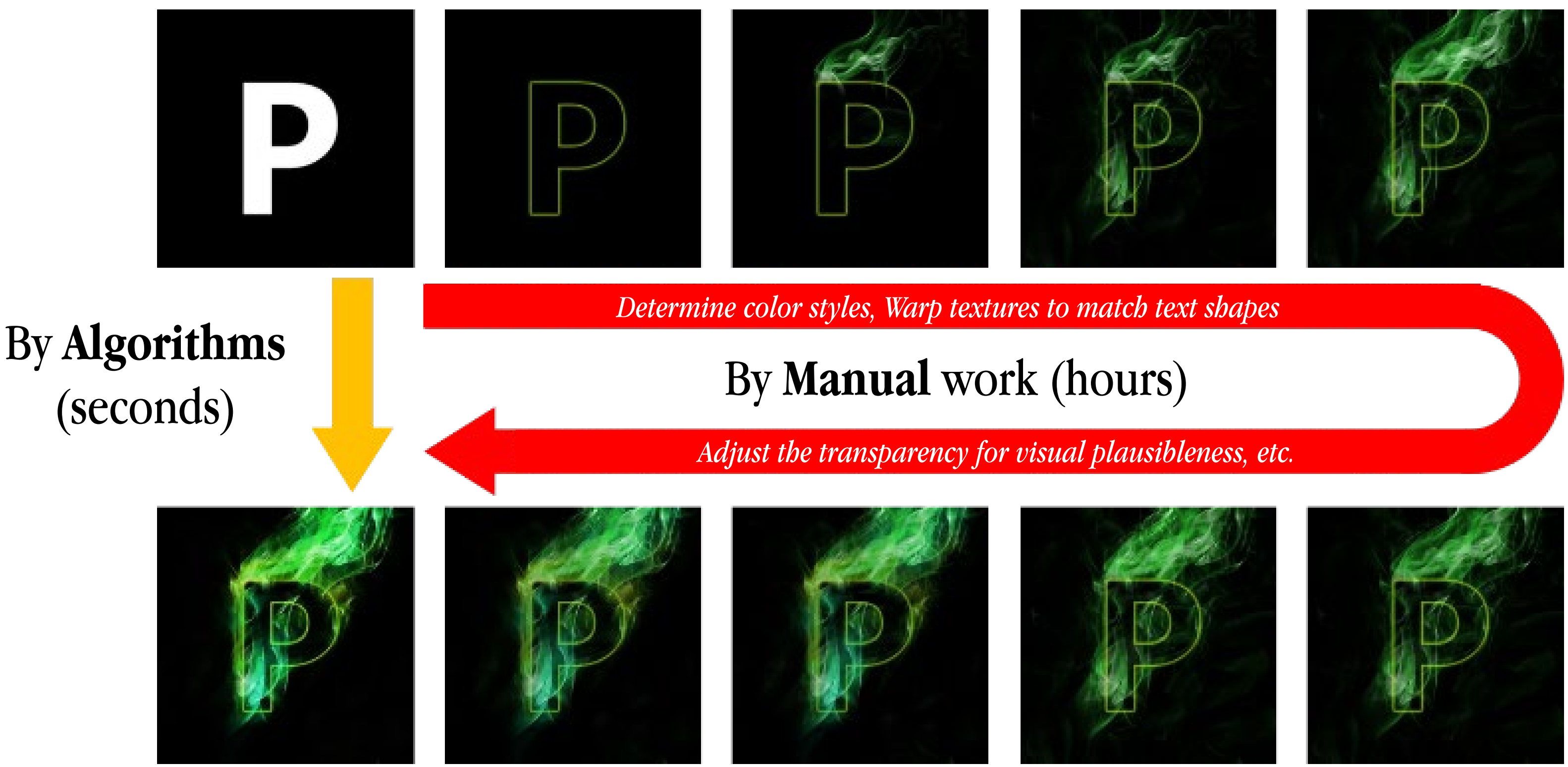
Pix2pix-cGAN, cycleGAN

Style is learned by data

## Problem Analysis

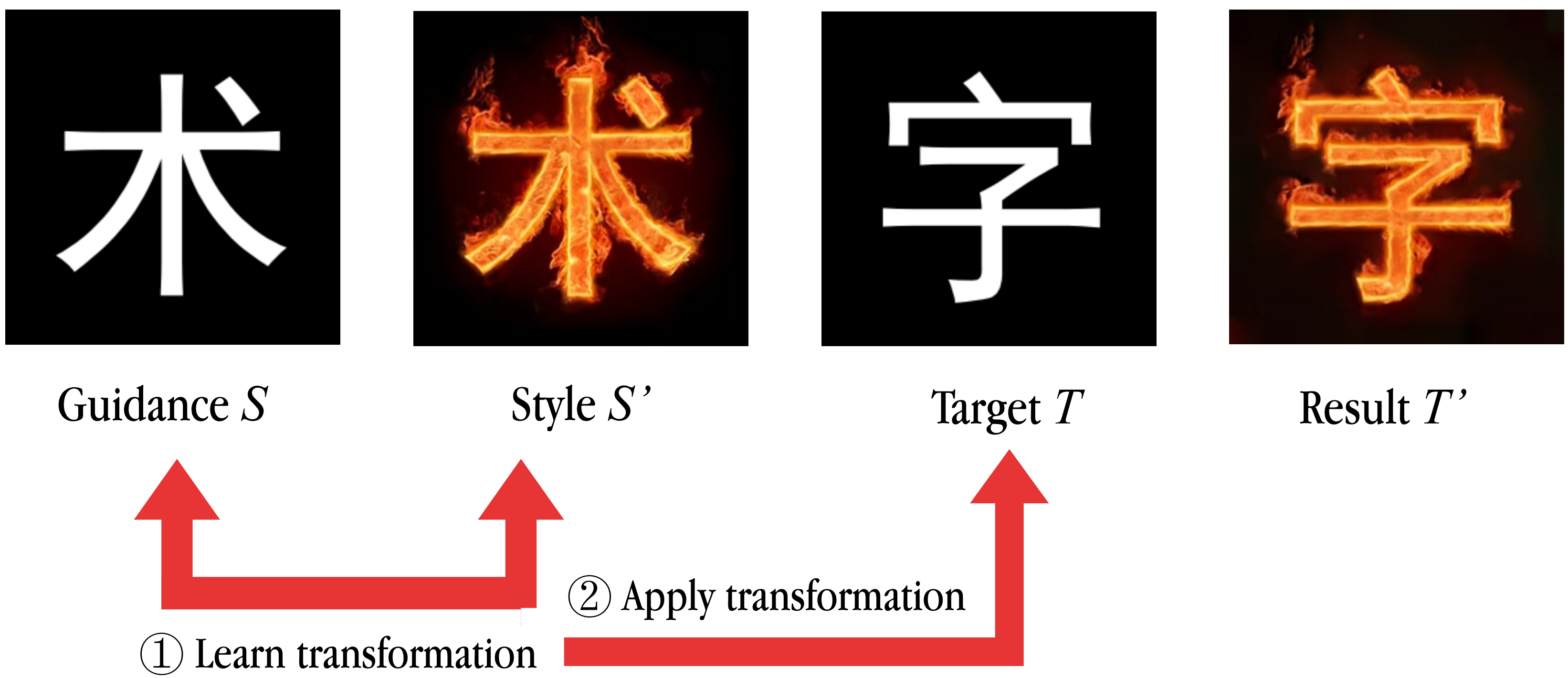
### Motivation

- Manually design: Require skills & Take too much time
- Automatic text effects transfer



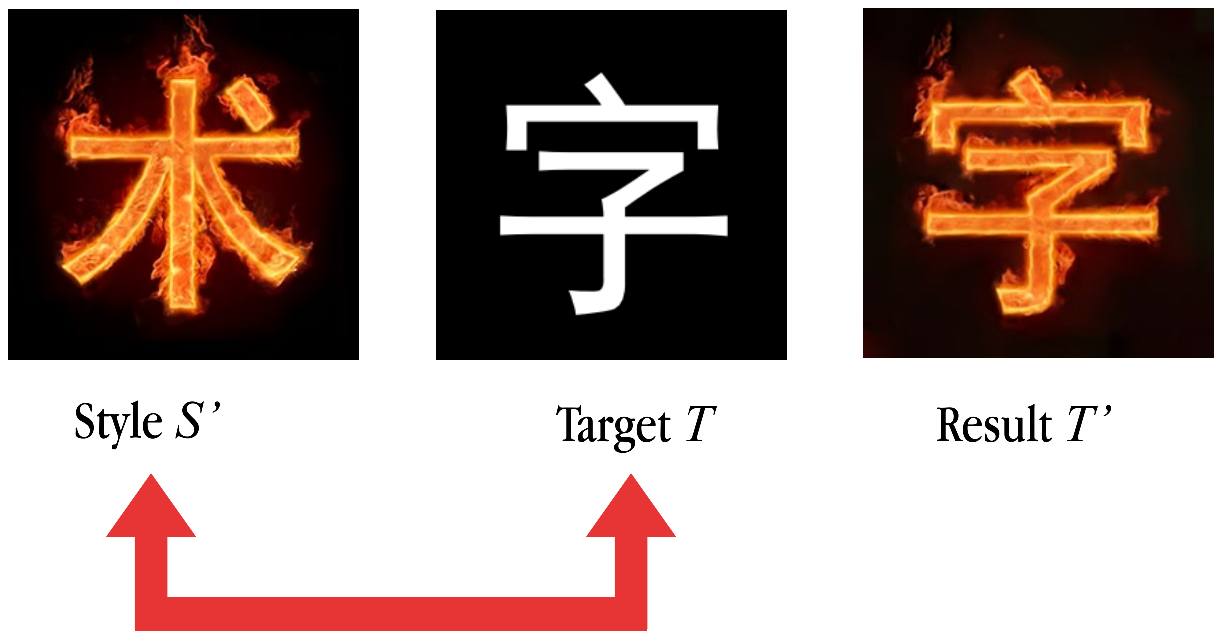
## Image Style Transfer

- **Supervised method**
- Unsupervised method



# Image Style Transfer

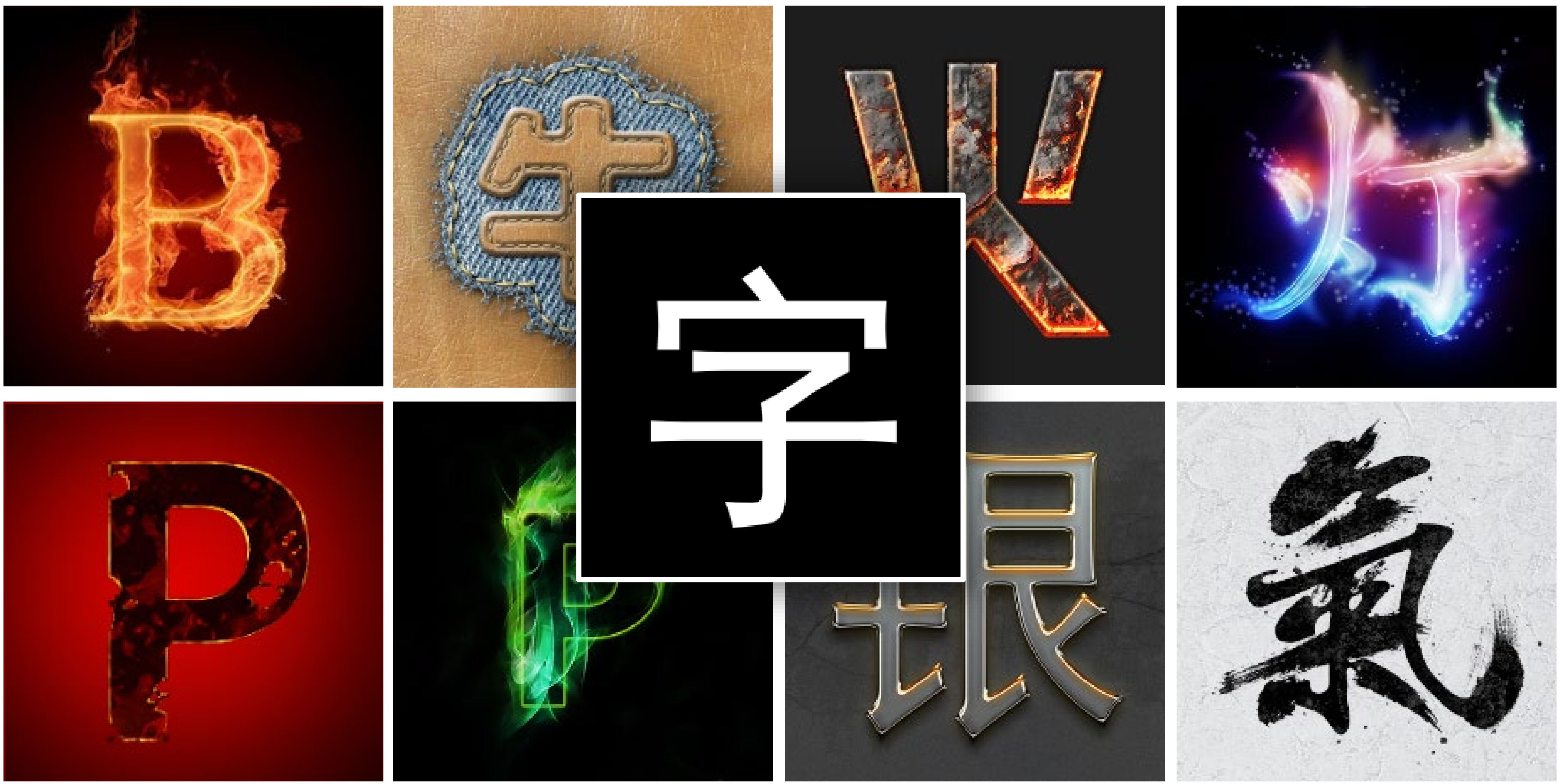
- Supervised method
- **Unsupervised method**



## Supervised Text Stylization

# Awesome Typography: Statistics-Based Text Effects Transfer

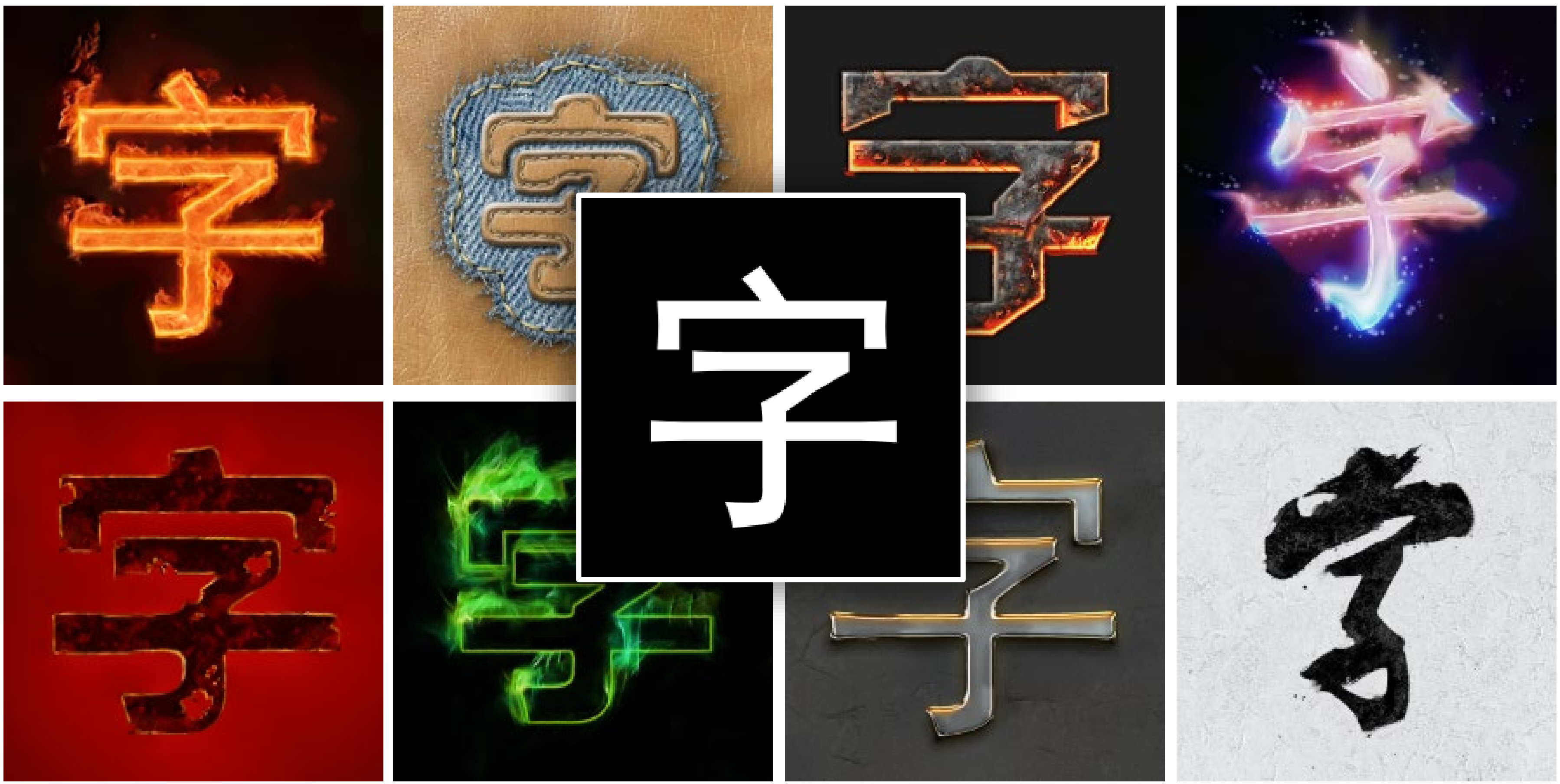
*Shuai Yang, Jiaying Liu, Zhouhui Lian, Zongming Guo*, CVPR 2017



## Supervised Text Stylization

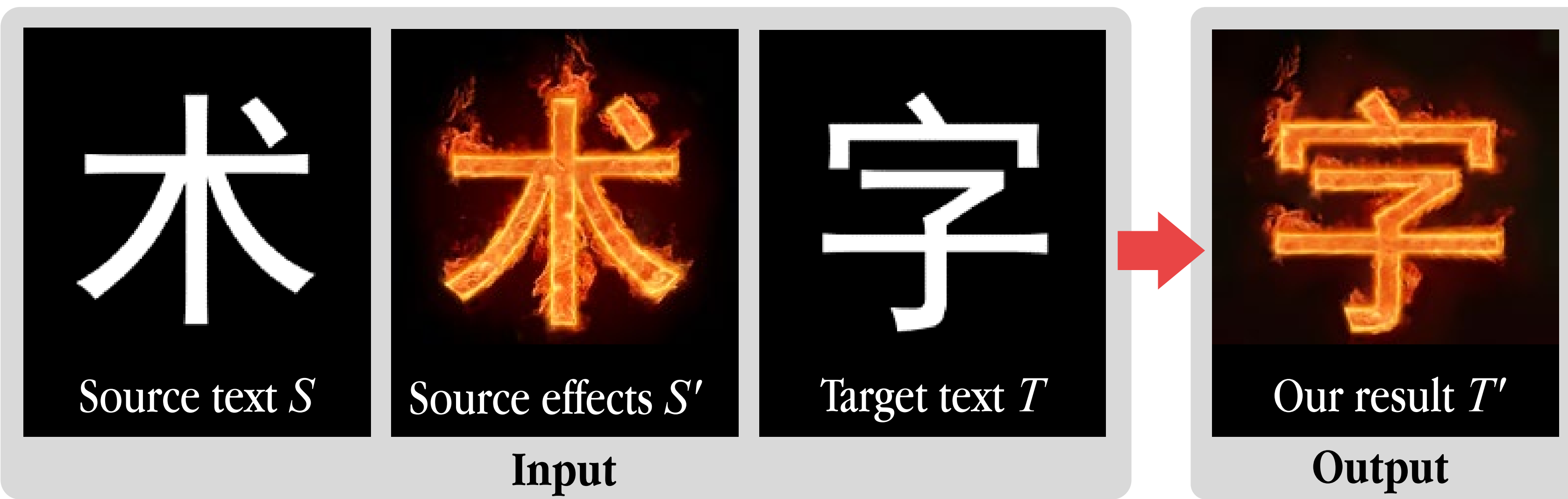
# Awesome Typography: Statistics-Based Text Effects Transfer

*Shuai Yang, Jiaying Liu, Zhouhui Lian, Zongming Guo*, **CVPR 2017**



# Problem: Text Effects Transfer

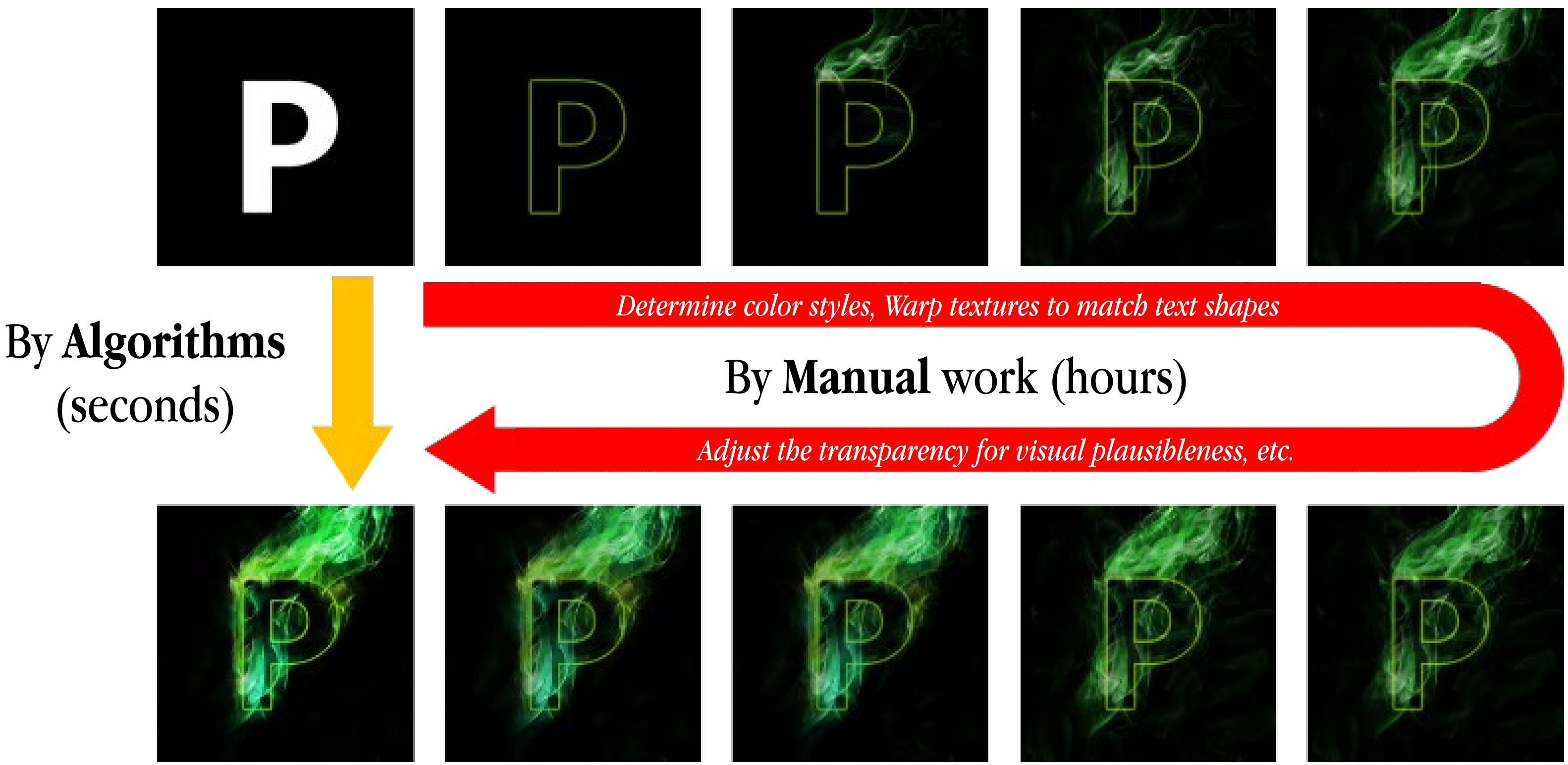
- Input: $S$, $S'$, $T$
- Output: $T'$



# Problem Analysis

## Motivation

- Manually design: Require skills & Take too much time
- Automatic text effects transfer



# Problem Analysis

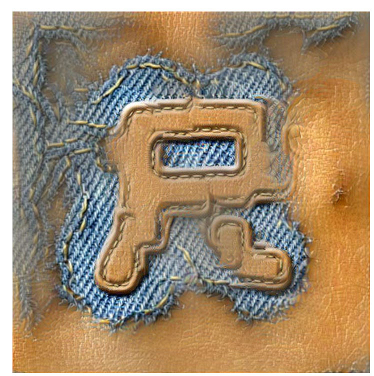
## Challenges

- Extreme diversity of the text effects and characters
- Complicated composition of style elements
  - → Texture-by-number methods ✗
- Simpleness of guidance images
  - → Deep-based methods ✗

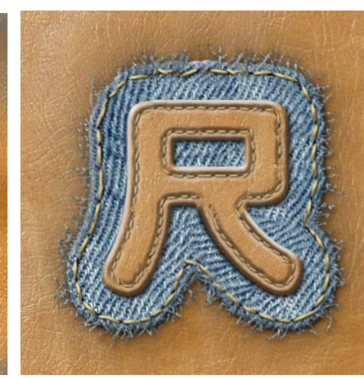
Complicated composition

Plain guidance images

texture-by-number
Image Analogies (2001)



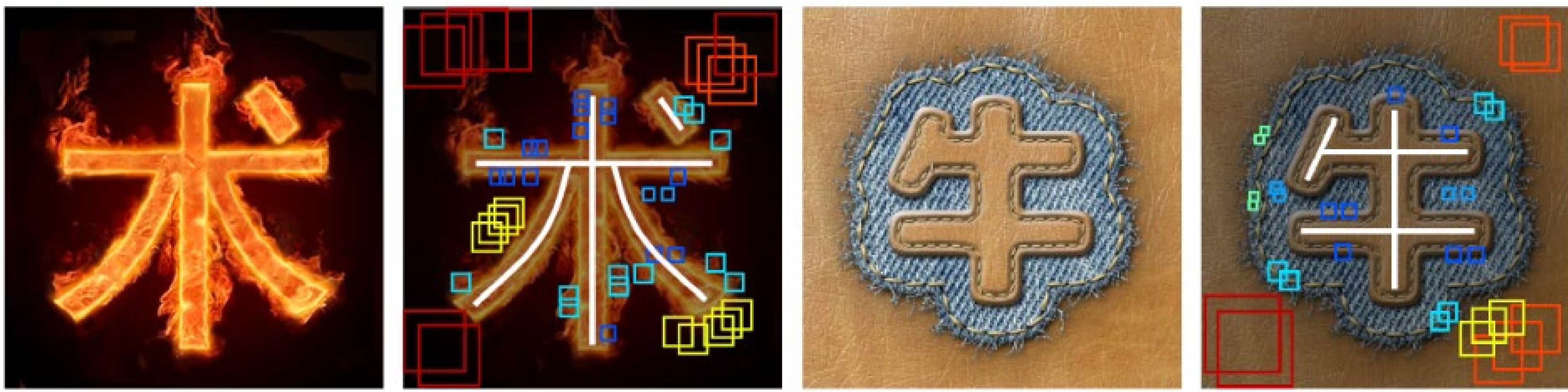
deep-based
Neural Doodles (2016)



Our result

## Problem Analysis

- **Key Observation:** Distance-based characteristics
  - Textures with similar **distances** share similar **patterns**
  - **Distance**: to text skeleton
  - **Pattern**: ***Color*** and ***Scale***



Different patch colors represent different distances to the text skeleton (white)

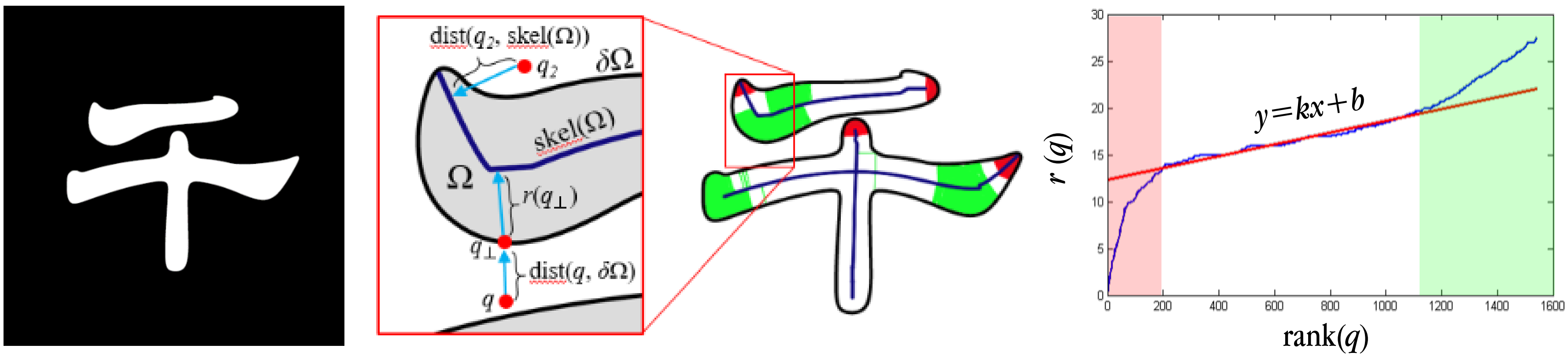
# Statistics-Based Texture Effects Transfer

- **Texture Effects Statistics Calculation**
  - Distance
  - Optimal patch scale

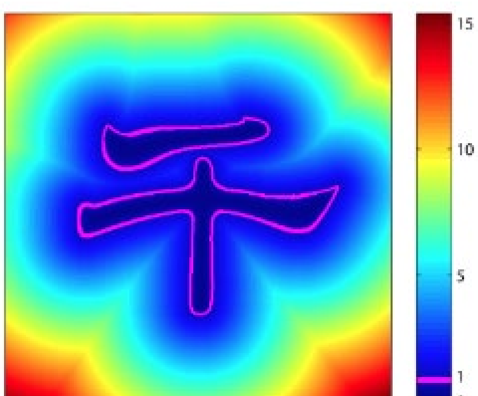
  *Relationship*
- **Text Effects Generation**
  - Multiscale
  - Texture distribution
  - Naturalness

# Text Effects Statistics Estimation

## Distance Estimation




(a) Text image (b) Radius and distance (c) Text skeleton (d) Radius and its rank



(e) Normalized distance

### Normalized distance calculation

$$\widetilde{\text{dist}}(q, \text{skel}(\Omega)) = \begin{cases} 1 + \text{dist}(q, \delta\Omega)/\overline{r}, & \text{if } q \notin \Omega \\ 1 - \text{dist}(q, \delta\Omega)/\widetilde{r}(q_\perp), & \text{other} \end{cases}$$

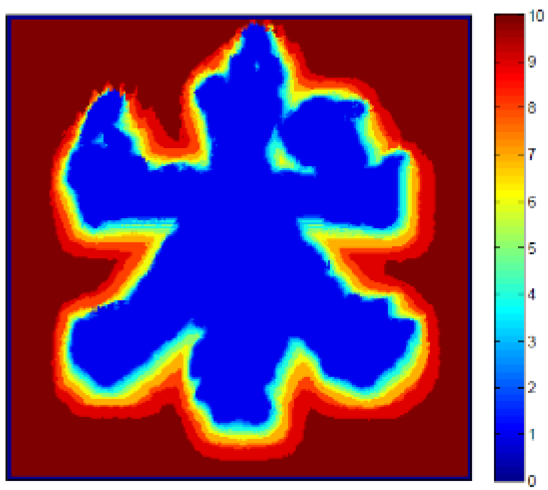
# Text Effects Statistics Estimation

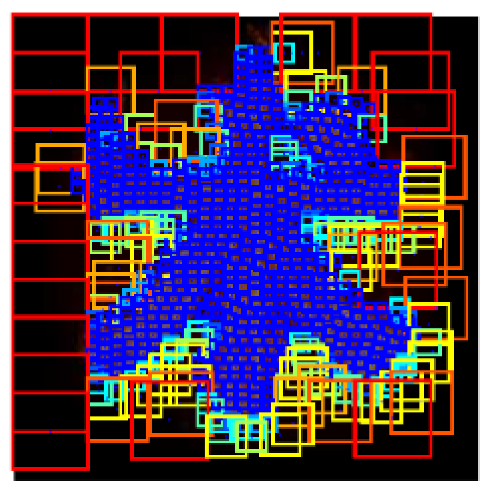
## Optimal Patch Scale Detection

- Fix patch size and resize image to realize multiple scales
- Determine from large scale to small scale
- Determination criterion:

$$d_\ell(q,\hat{q}) = \underbrace{\|Q_\ell(q) - Q_\ell(\hat{q})\|^2}_{\text{Text shape}} + \underbrace{\|Q'_\ell(q) - Q'_\ell(\hat{q})\|^2}_{\text{Texture}}$$

$$\zeta_\ell(q,\hat{q}) = \left(\sigma_\ell + \sqrt{d_\ell(q,\hat{q})} > \omega\right)$$



# Text Effects Statistics Estimation

## Relation Between Optimal Patch Scale and Distance

- **Histogram**

$$hist(\ell, x) = \sum \psi(\mathrm{scal}(q) = \ell \wedge \mathrm{bin}(q) = x)$$

- **Joint probability**

$$\mathcal{P}(\ell, x) = hist(\ell, x) / \sum_{\ell, x} hist(\ell, x)$$

- **Posterior probability**

$$p \sim \mathcal{P}(\ell | \mathrm{bin}(q)) = \mathcal{P}(\ell, \mathrm{bin}(q)) / \sum_{\ell} \mathcal{P}(\ell, \mathrm{bin}(q))$$

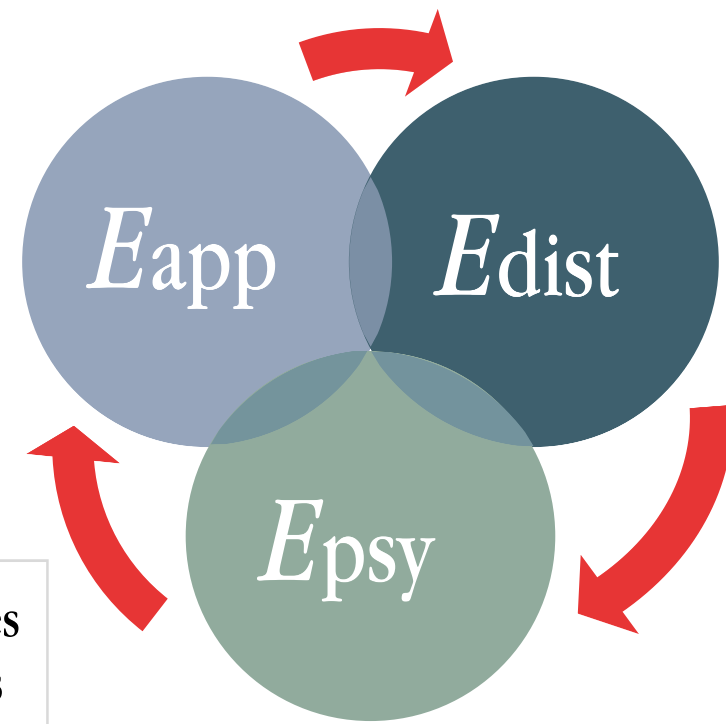
(for $l$ being the appropriate scale to depict the patches with distances corresponding to bin($q$))

## Text Effect Transfer

### Objective Function

$$\min_{q} \sum_{p} E_{\mathrm{app}}(p, q) + \lambda_1 E_{\mathrm{dist}}(p, q) + \lambda_2 E_{\mathrm{psy}}(p, q)$$

**Appearance Term**
Texture Style Transfer
**Low-level**
**objective** evaluation




**Distribution Term**
Spatial Style Transfer
**Mid-level**
**objective** evaluation

**Psycho-Visual Term**
Naturalness Preservation
**subjective** evaluation

$p$: pixel in source images
$q$: pixel in target images

# Text Effect Transfer

## Appearance Term

$$E_{\text{app}}(p,q) = \lambda_3 \sum_{\ell} \mathcal{P}(\ell|\text{bin}(p)) \underbrace{\boxed{\|P_\ell(p) - Q_\ell(q)\|^2}}_{\substack{\textbf{Text shape term} \\ \text{Constrain text shape}}} + \sum_{\ell} \mathcal{P}(\ell|\text{bin}(p)) \underbrace{\boxed{\|P'_\ell(p) - Q'_\ell(q)\|^2}}_{\substack{\textbf{Text effects term} \\ \text{Ensure texture transfer}}}$$

$P/P'$ : patch in $S/S'$ centered at $p$

$Q/Q'$ : patch in $T/T'$ centered at $q$

# Text Effect Transfer

## Appearance Term

$$E_{\text{app}}(p,q) = \lambda_3 \sum_{\ell} \boxed{\mathcal{P}(\ell|\text{bin}(p))} \|P_\ell(p) - Q_\ell(q)\|^2 + \sum_{\ell} \boxed{\mathcal{P}(\ell|\text{bin}(p))} \|P'_\ell(p) - Q'_\ell(q)\|^2$$

Weighted by posterior probability

Patches in different regions use different scales

**Small patch:**

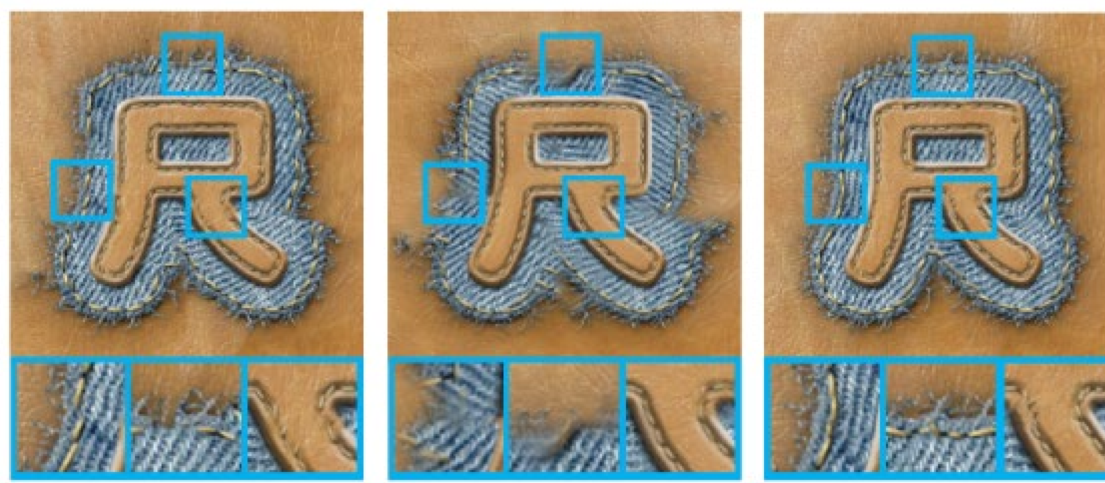
- Lose structures

**Large patch:**

- Lose textures

**Multi-scale:**

- Preserve texture structures
- Preserve texture details



single-scale 5*5 single-scale 15*15 multi-scale 5*5

# Text Effect Transfer

## Distribution Term

$$E_{\text{dist}}(p,q) = \left(\text{dist}(p) - \text{dist}(q)\right)^2 / \max(1, \text{dist}^2(p))$$

- Encourage the text effects of $T'$ to share similar distribution of $S'$
- Used for initialization

no $E_{\text{dist}}$ | with $E_{\text{dist}}$ | with $E_{\text{dist}}$ + initialization

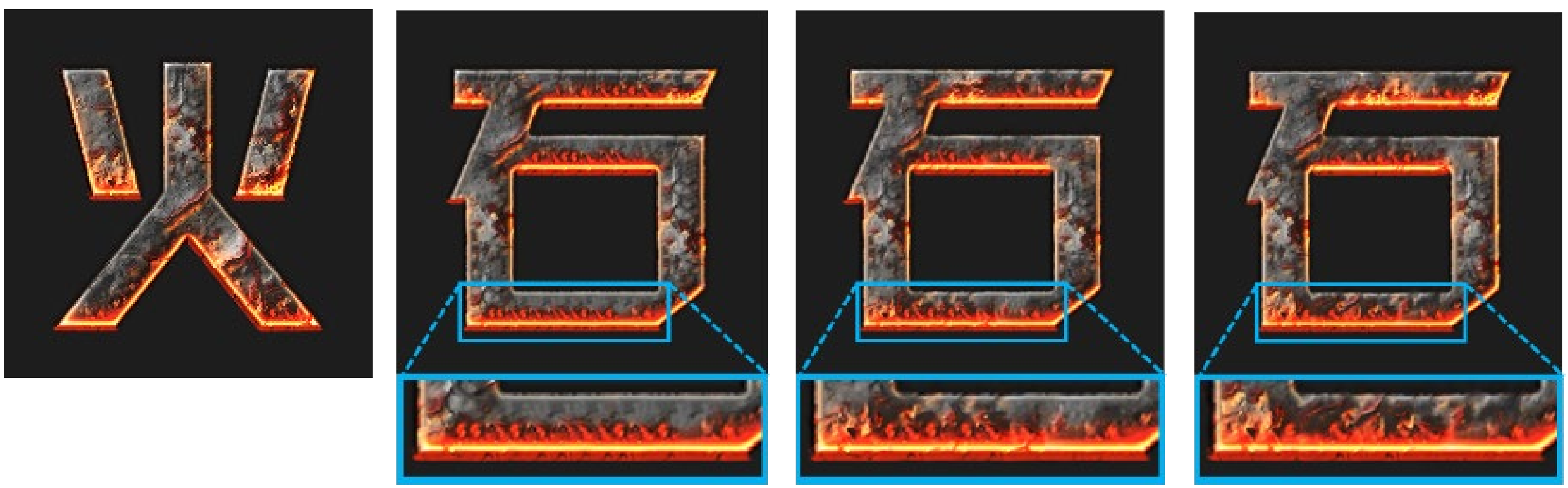
# Text Effect Transfer

## Psycho-Visual Term

$$E_{\text{psy}}(p, q) = |\Phi(q)|$$

$|\Phi(q)|$: Number of patches that find $Q(q)$ as their matched patch

- Penalize texture over-repetitiveness
- Encourage new texture creation



Weights for $E_{\text{psy}}$ (a) $\lambda_2 = 0.0$ (b) $\lambda_2 = 0.005$ (c) $\lambda_2 = 0.01$

# Text Effect Transfer

- Transfer for words

## Stroke Term $E_{str}$

Control the texture similarity of similar strokes

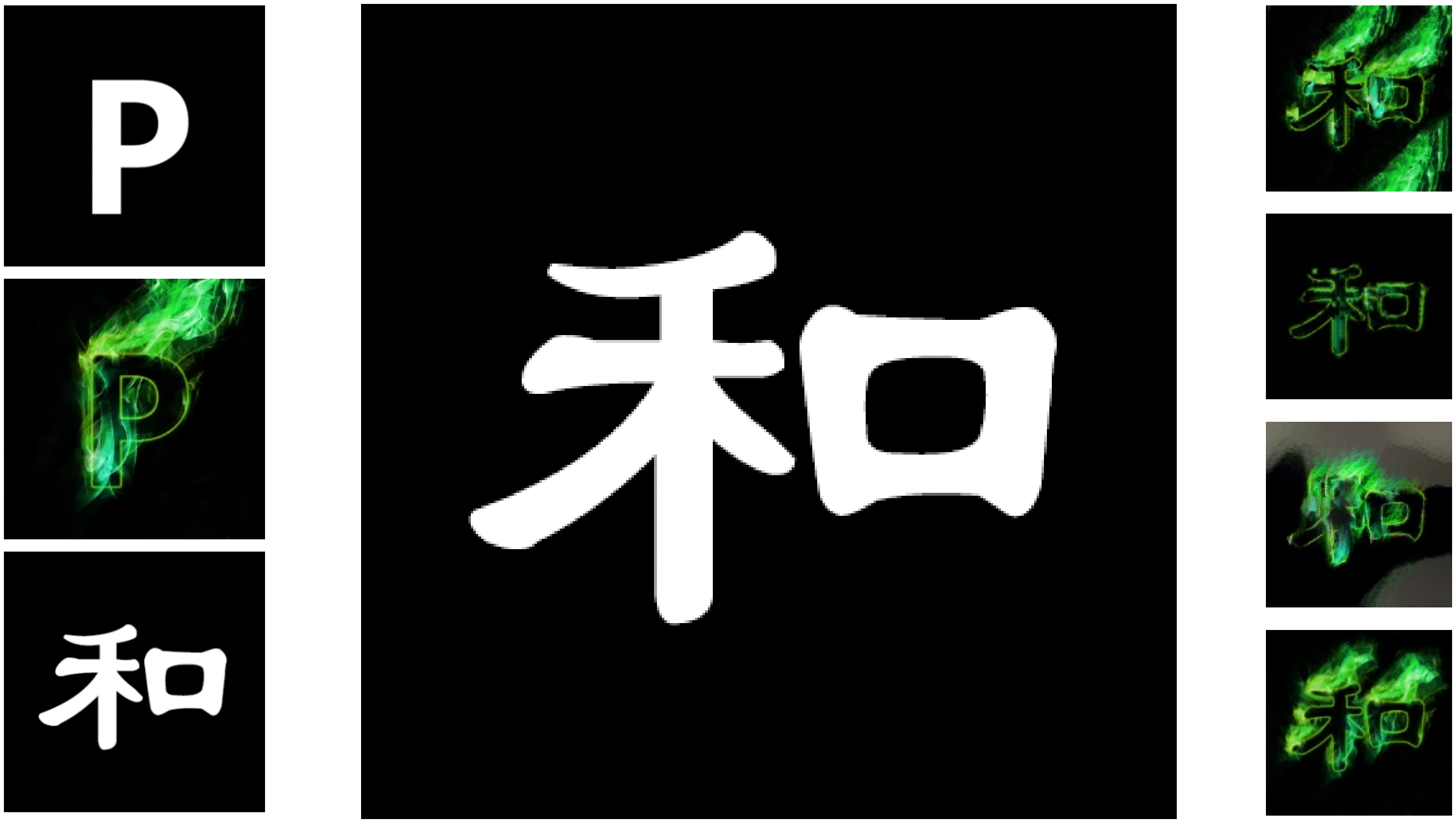
Positive weight:

- stylize in a more consistent way

Negative weight:

- transfer more diverse textures


## Comparison with Other Methods




Target Text

# Comparison with Other Methods


Image Analogy

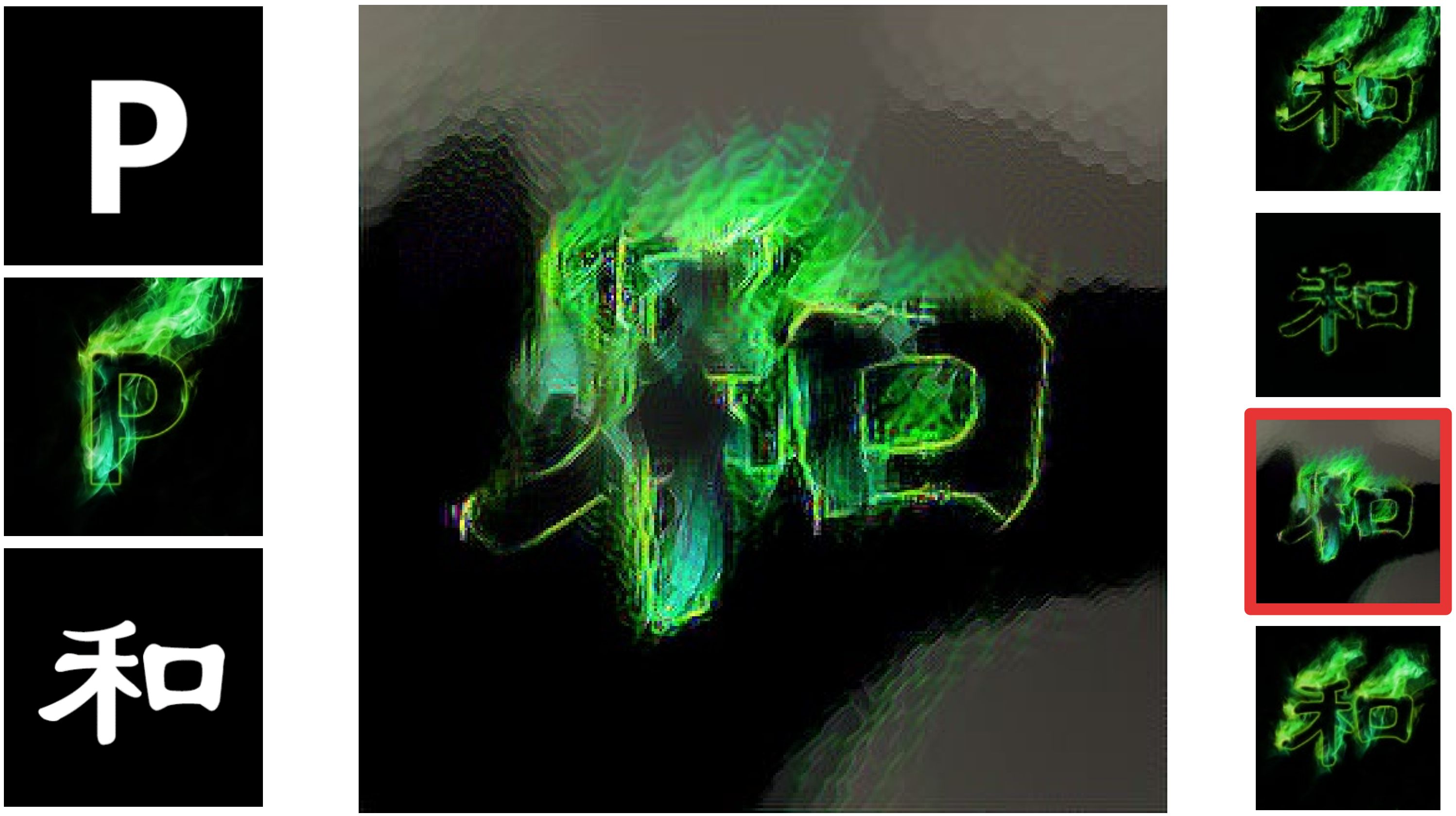
A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin.. “Image analogies”. SIGGRAPH. 2001.

# Comparison with Other Methods


Split and Match

O. Frigo, N. Sabater, J. Delon, and P. Hellier. Split and match: example-based adaptive patch sampling for unsupervised style transfer". CVPR. 2016.

# Comparison with Other Methods



Neural Doodles

A. J. Champandard. “Semantic style transfer and turning two-bit doodles into fine artworks”. Arxiv. 2016.

# Comparison with Other Methods


Our method

# Comparison with Other Methods


(a) $(S,S')$ (b) $T$ (c) Image Analogies[1] (d) Split & Match[2] (e) Neural Doodle[3] (f) Baseline (g) Proposed

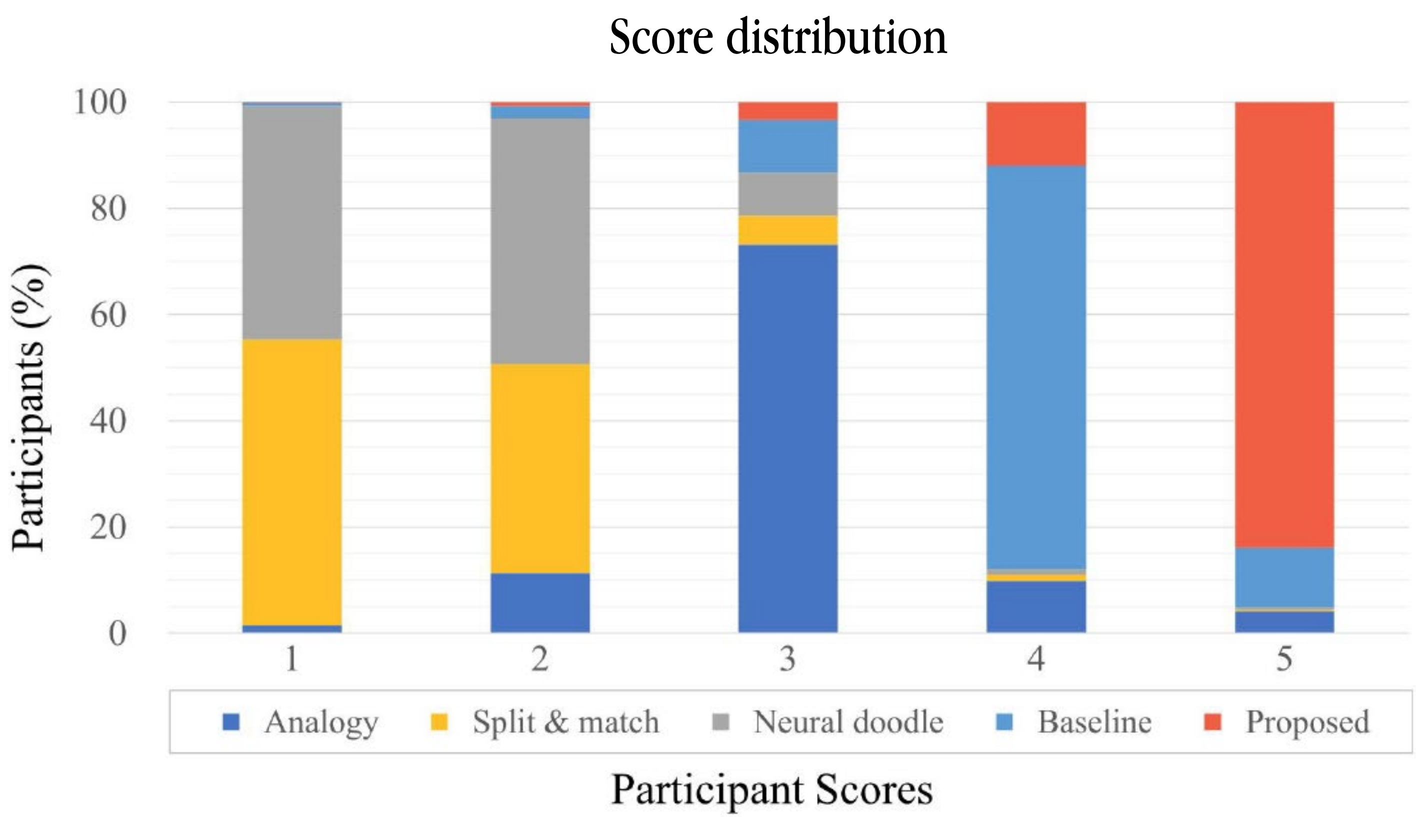
[1]A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. "Image analogies". SIGGRAPH. 2001.

[2]O. Frigo, N. Sabater, J. Delon, and P. Hellier. Split and match: example-based adaptive patch sampling for unsupervised style transfer". CVPR. 2016.

[3]A. J. Champandard. "Semantic style transfer and turning two-bit doodles into fine artworks". Arxiv. 2016.

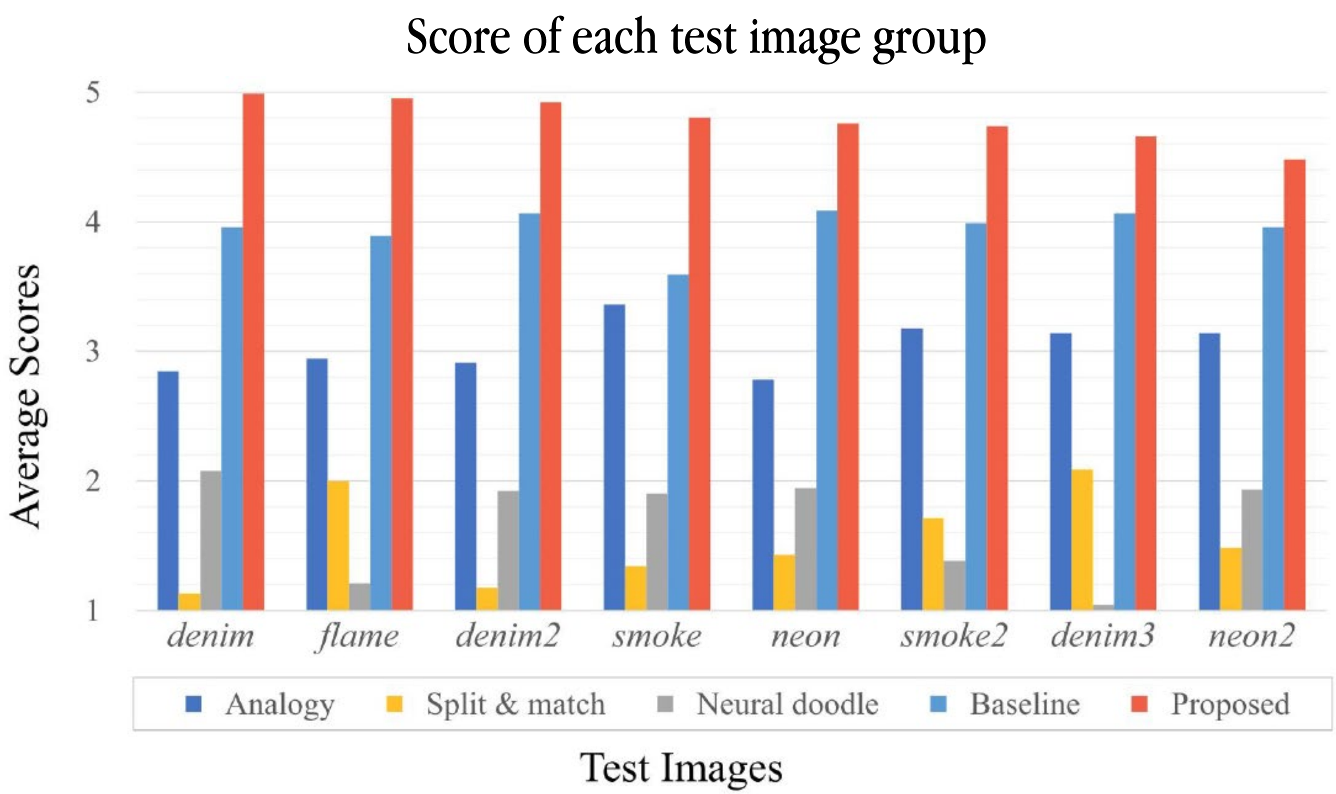
# User Study

- 91 participants, give 1-5 scores (5 means best)
  - Averaged scores: 3.04, 1.54, 1.68, 3.95, 4.79



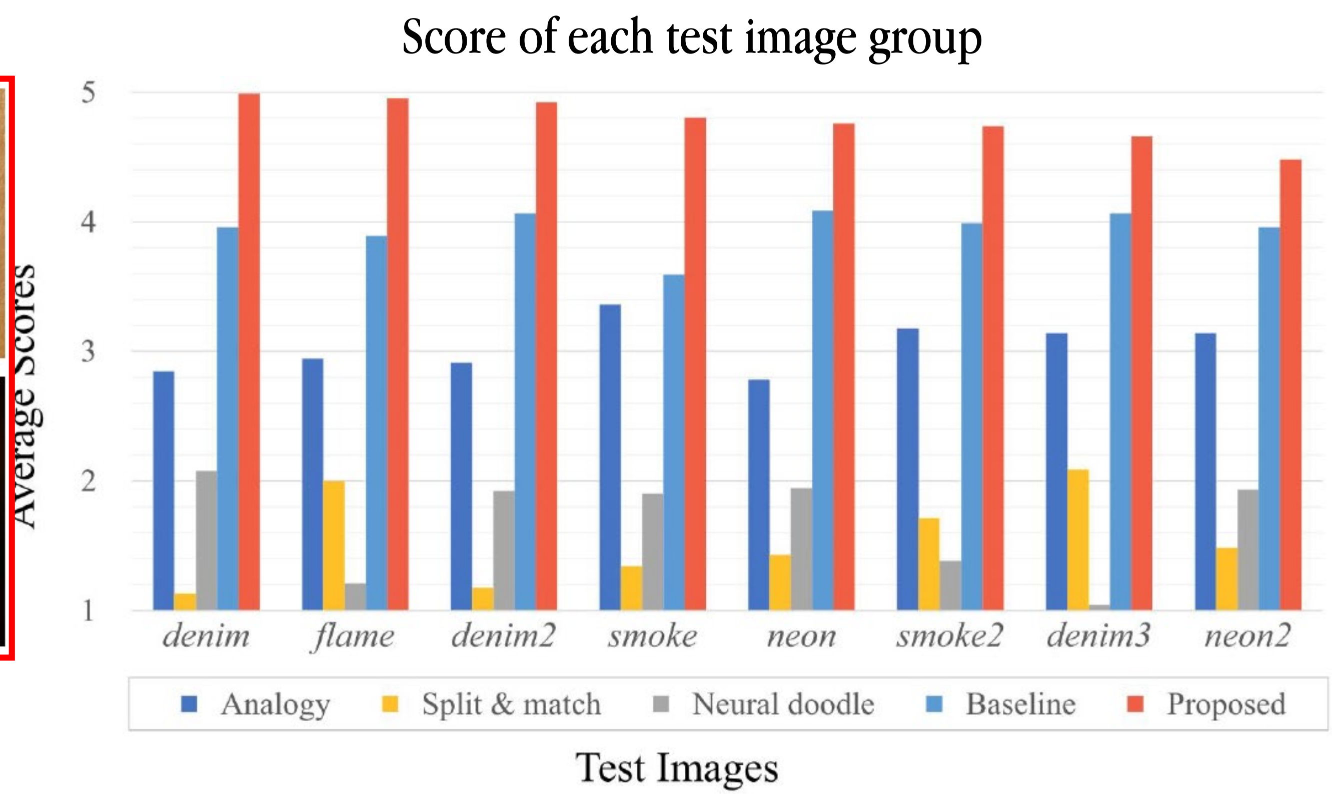
# User Study

- 91 participants, give 1-5 scores (5 means best)
  - Highest score over all 8 test images



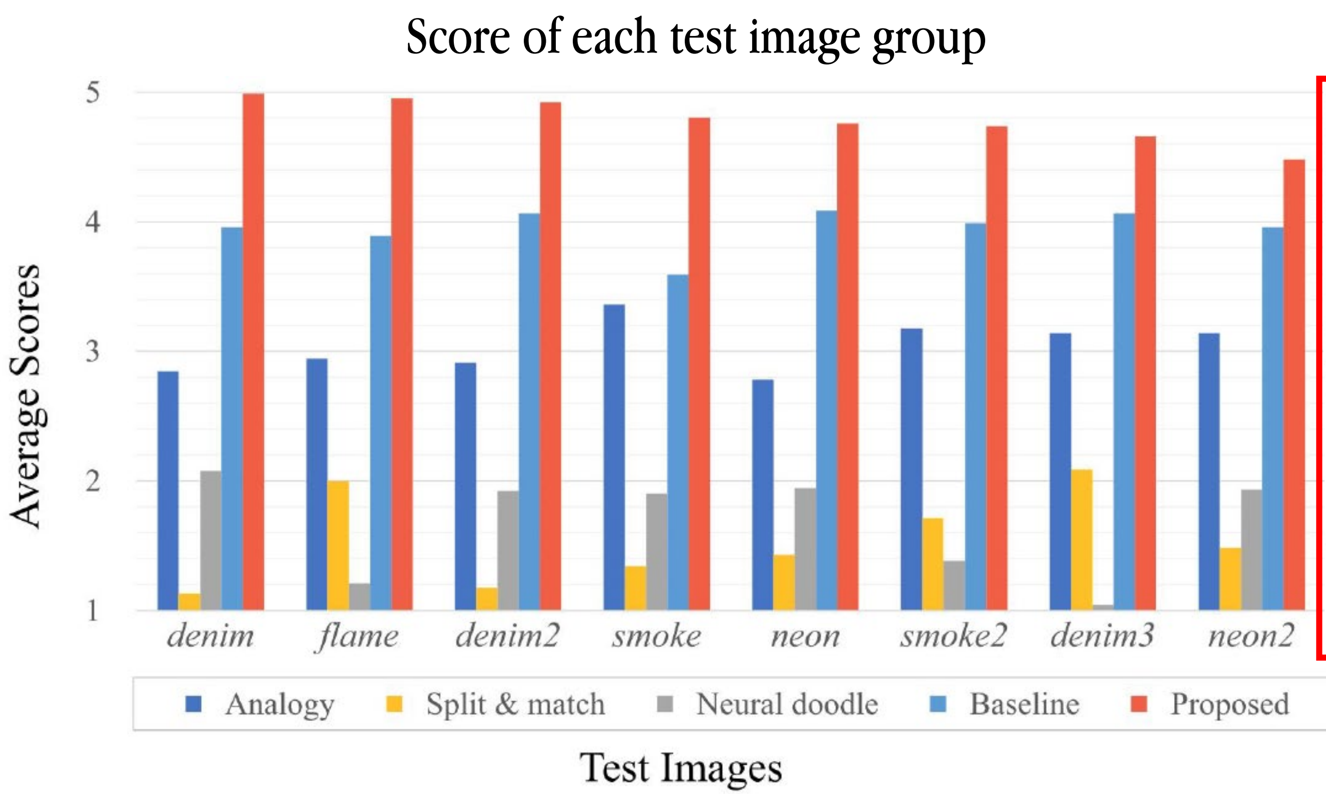
# User Study

- 91 participants, give 1-5 scores (5 means best)
  - Clear advantages: highly regular texture effects



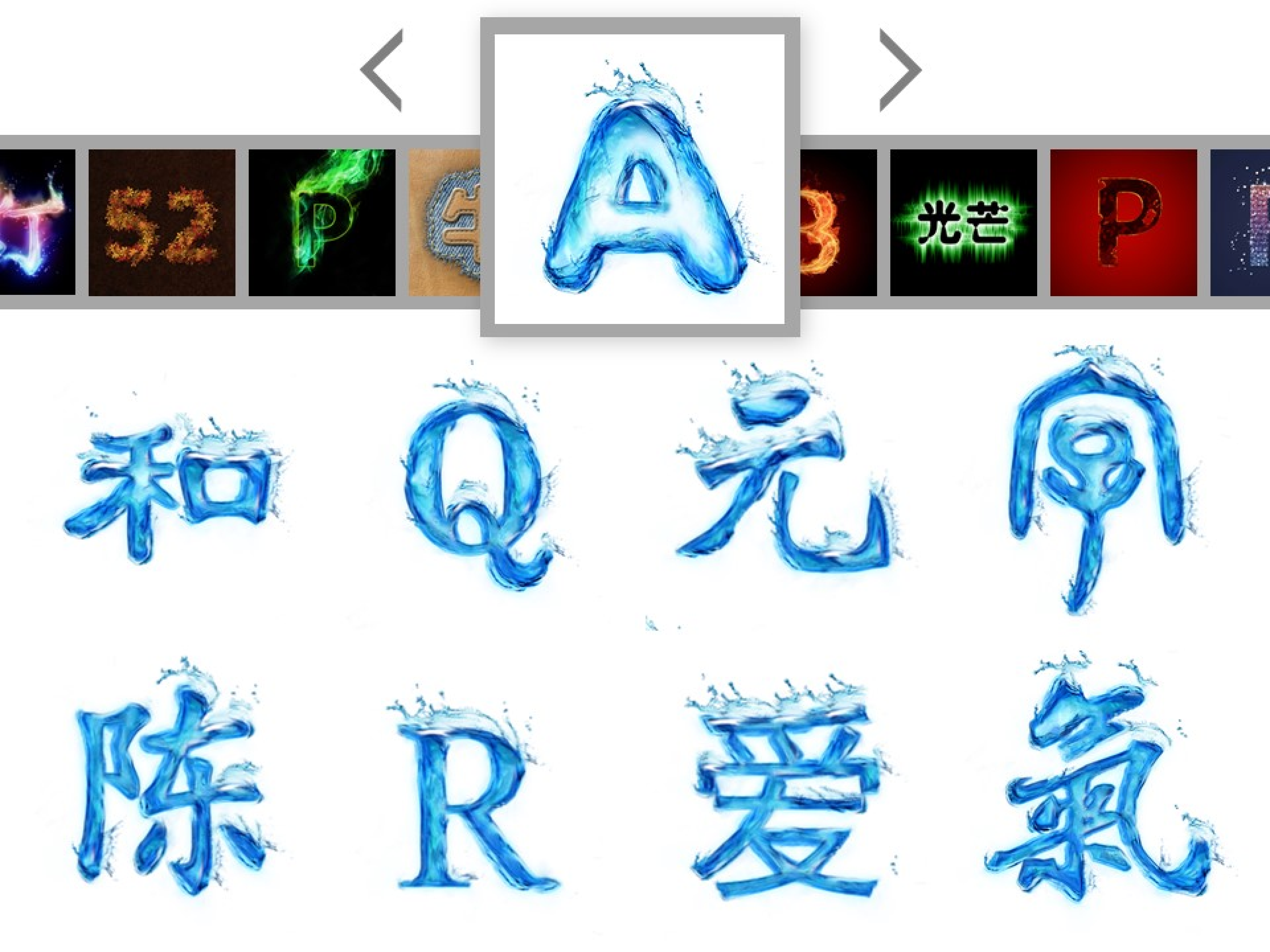
# User Study

- 91 participants, give 1-5 scores (5 means best)
  - Less obvious advantages: simple effects, fail to cover all colors




# Diversified Transfer

- Apply different text effects to representative characters (Chinese, alphabetic, handwriting)



52
P
A
光芒
P
和
Q
陈
R
爱
氣

2
P
牛
B
芒
P
R
和
Q
元
陈
R
爱
氣

光芒
牛
A
R
型
和
Q
元
陈
R
爱
氣

A
B
光
P
型
水
和
Q
元
陈
R
爱
氣


B
光芒
R
水
银
和
Q
元
陈
R
爱
氣

## Typography Library Generation

爱 氨 岸 盎 凹 敖 奥 澳 芭

八 把 爸 佰 班 办 帮 包 保

报 豹 爆 奔 本 鼻 比 鄙 碧 弊 必

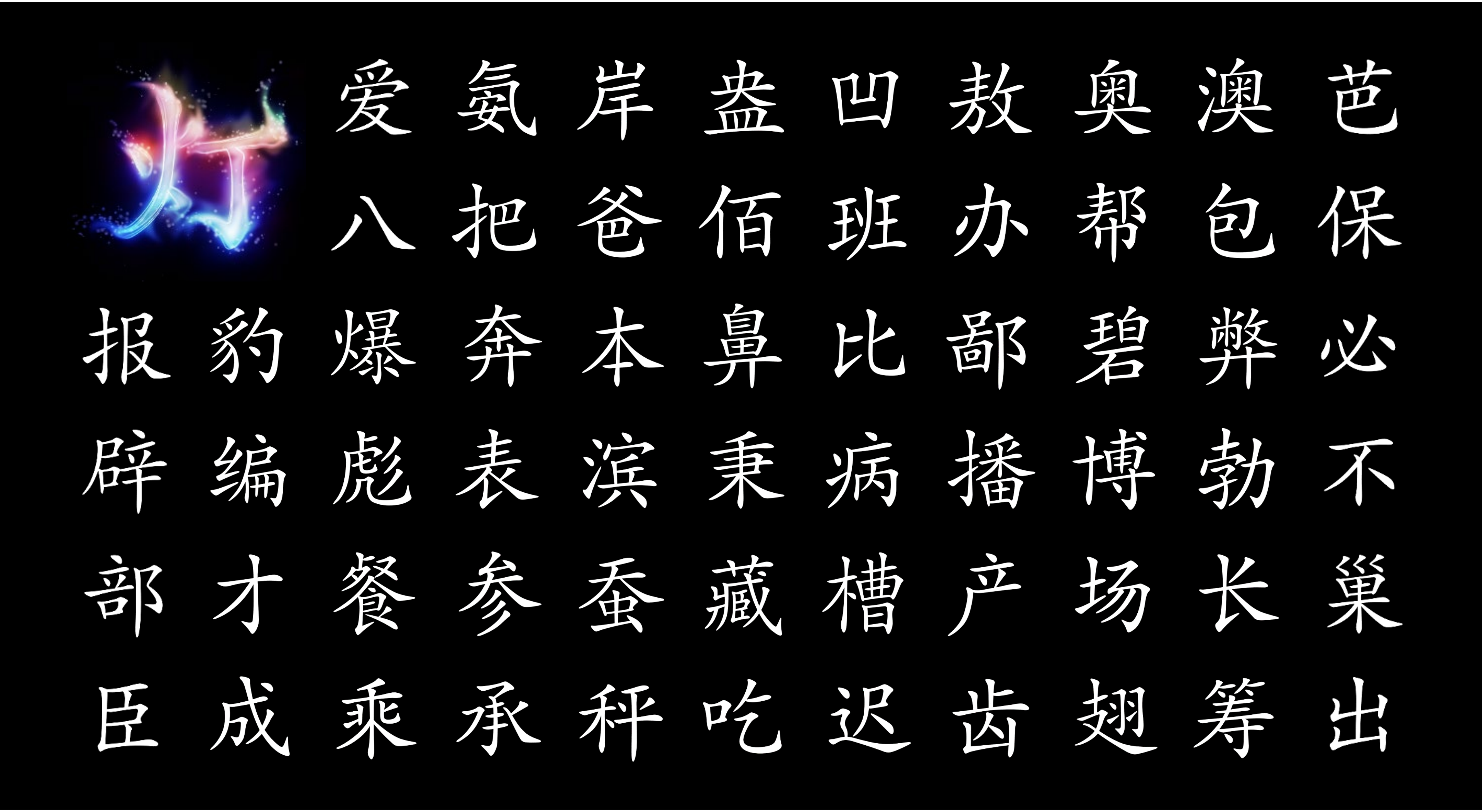
辟 编 彪 表 滨 秉 病 播 博 勃 不

部 才 餐 参 蚕 藏 槽 产 场 长 巢

臣 成 乘 承 秤 吃 迟 齿 翅 筹 出

## Typography Library Generation

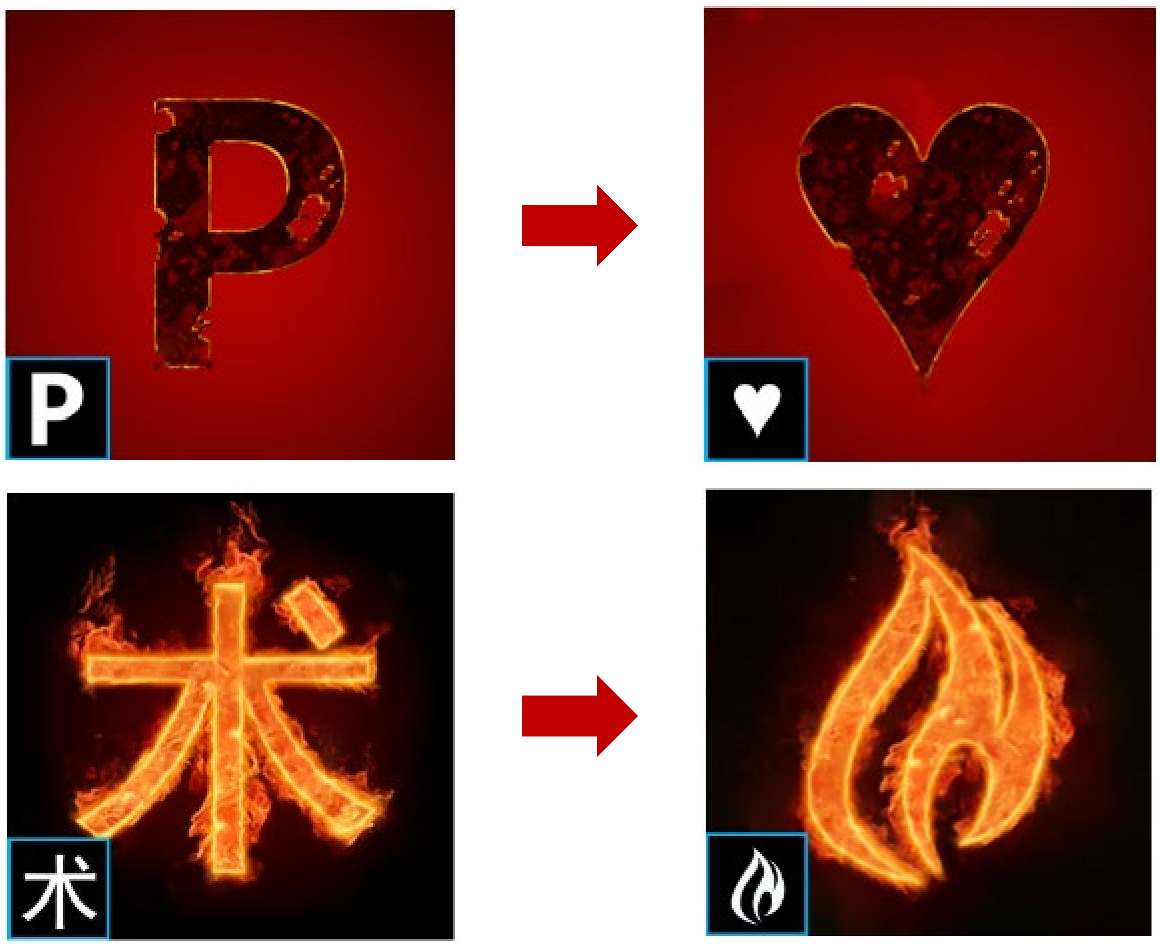
## Typography Library Generation



# Typography Library Generation

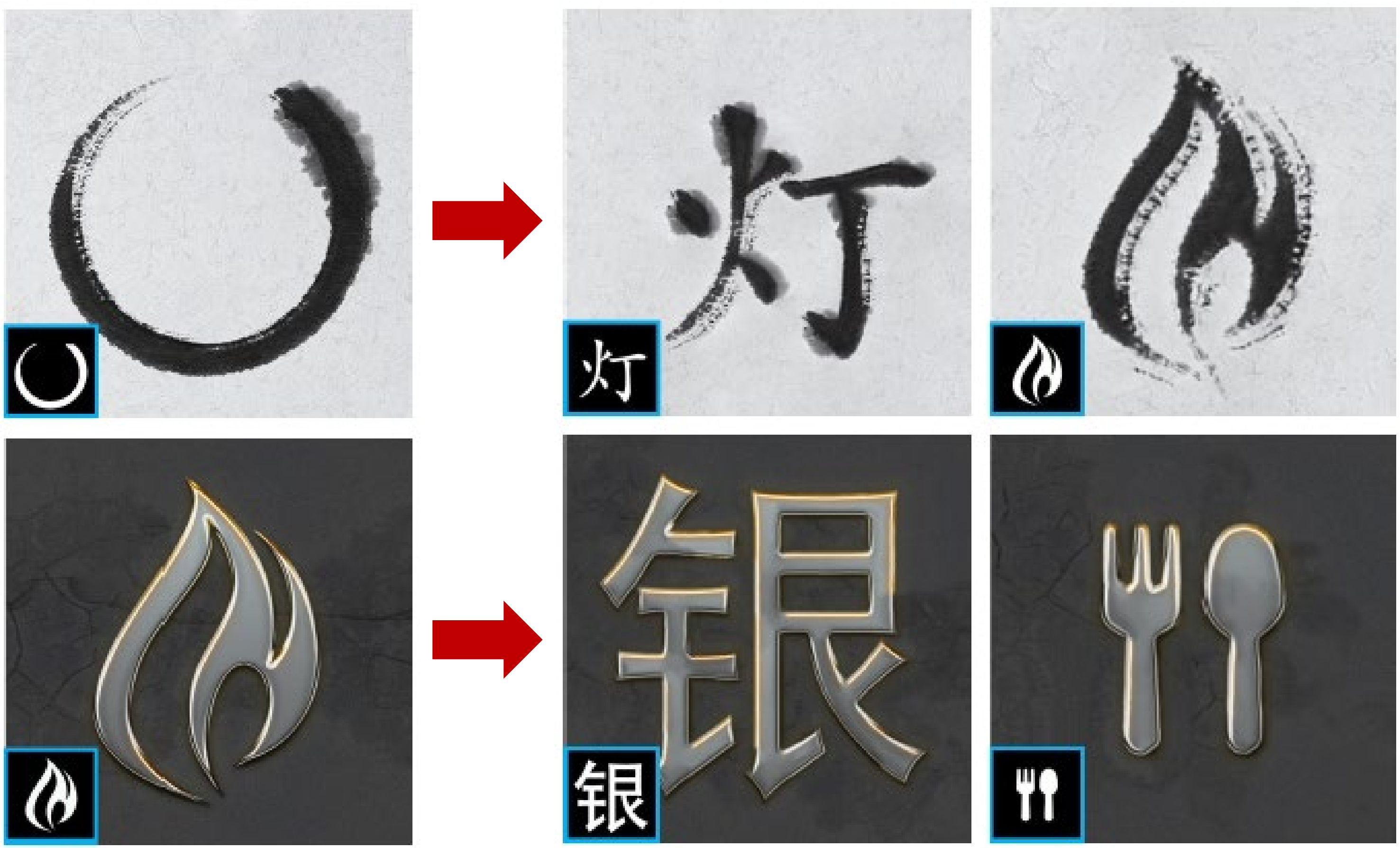
# Stroke-Based Graphics Rendering

- Text → Icons



# Stroke-Based Graphics Rendering

- Icons → Icons
- Icons → Text



## Unsupervised Text Stylization

# Context-Aware Unsupervised Text Stylization

Shuai Yang, Jiaying Liu, Wenhan Yang, Zongming Guo, **ACM MM 2018 / TIP 2019**


# Unsupervised Text Stylization

## Unsupervised Text Stylization

- Texture image $S'$ without guidance $S$
- $S'$ arbitrary texture images
- Broaden application scope

## Unsupervised Text Stylization

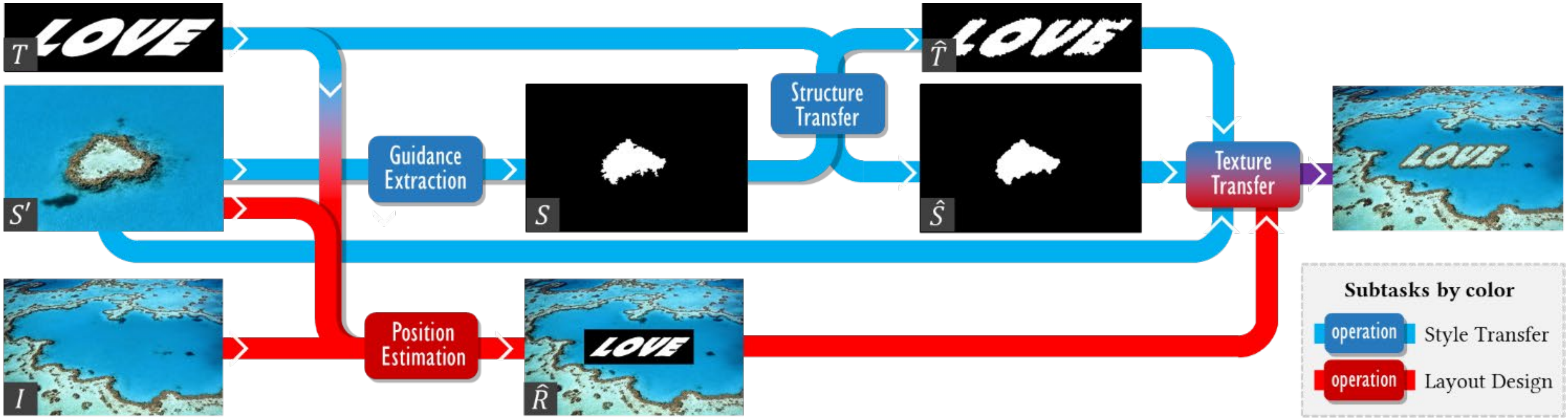
- **Context-Aware Layout Design**
  - Photo + text: poster, magazine cover, billboard
  - Optimal text layout selection
  - Seamlessly embedding


Background photo | $(T, S')$ | photo + text | Application: poster

# Context-Aware Unsupervised Text Stylization

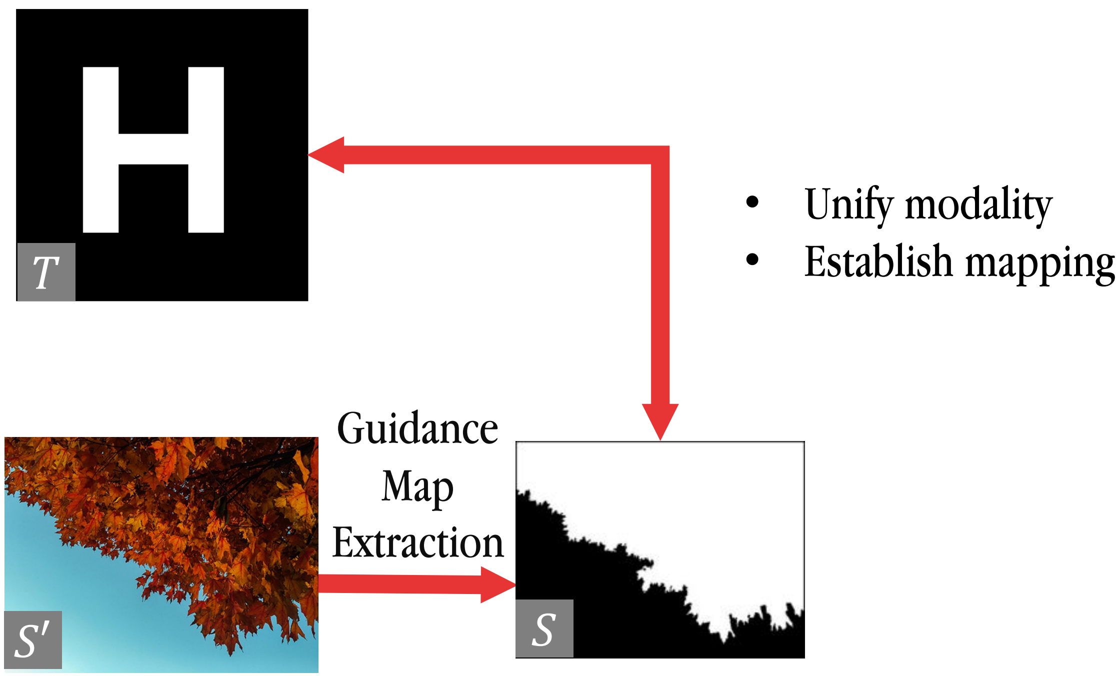
- **Unsupervised Text Stylization**
  - Structure transfer
  - Texture transfer

  } narrow the visual discrepancy
- **Context-Aware Layout Design**
  - Position estimation
  - Text embedding



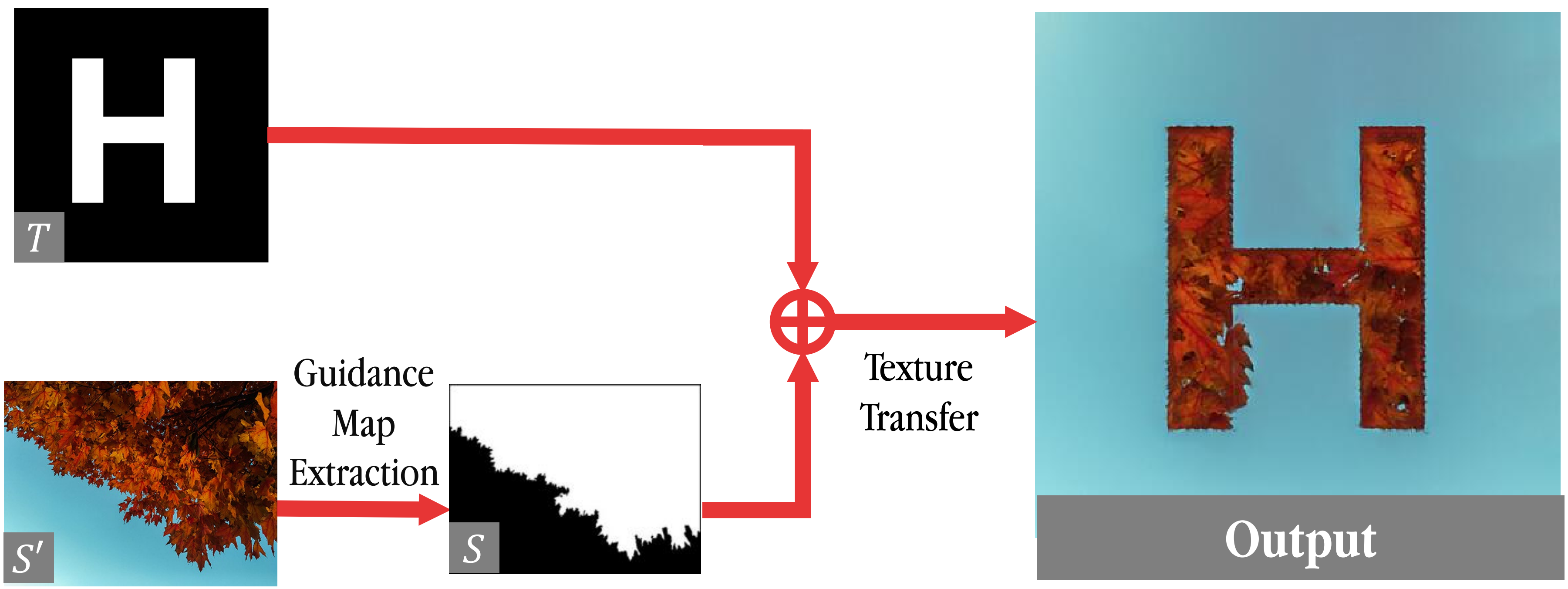
# Unsupervised Text Stylization

- **Guidance Map Extraction: Clustering**
- Structure Transfer
- Texture Transfer



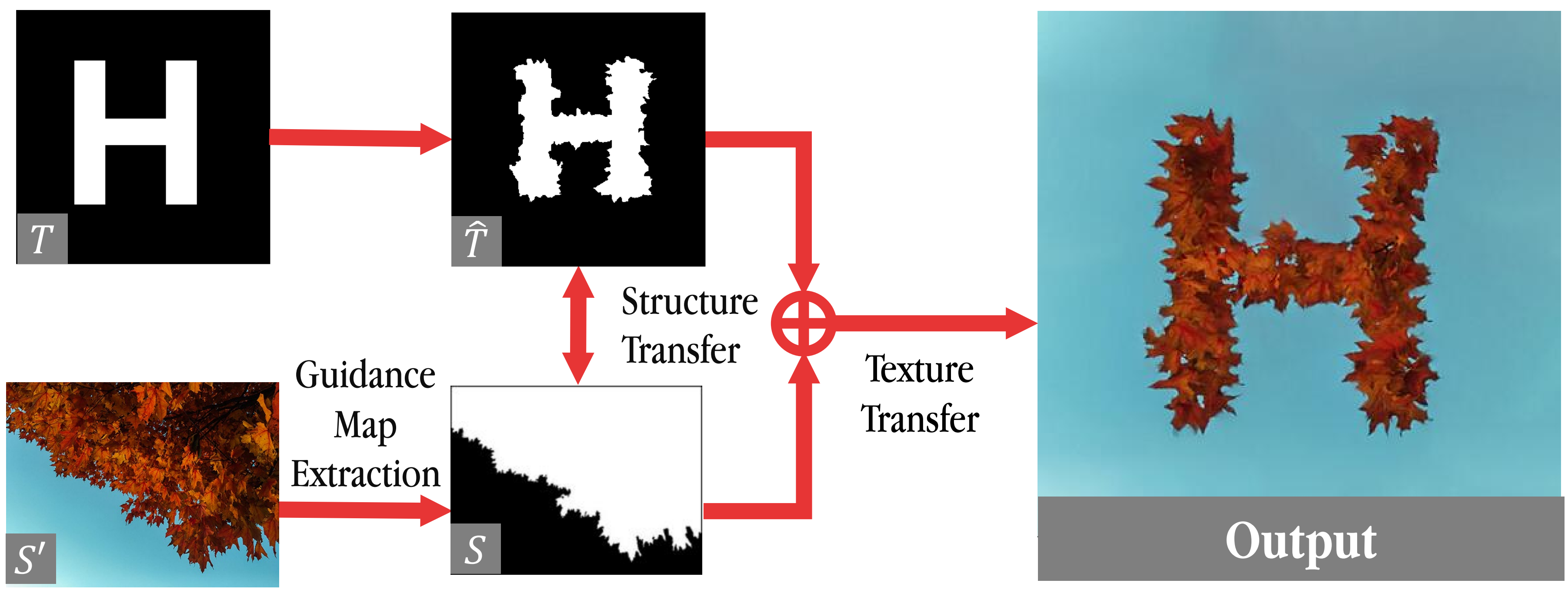
# Unsupervised Text Stylization

- Guidance Map Extraction
- Structure Transfer
- **Texture Transfer**



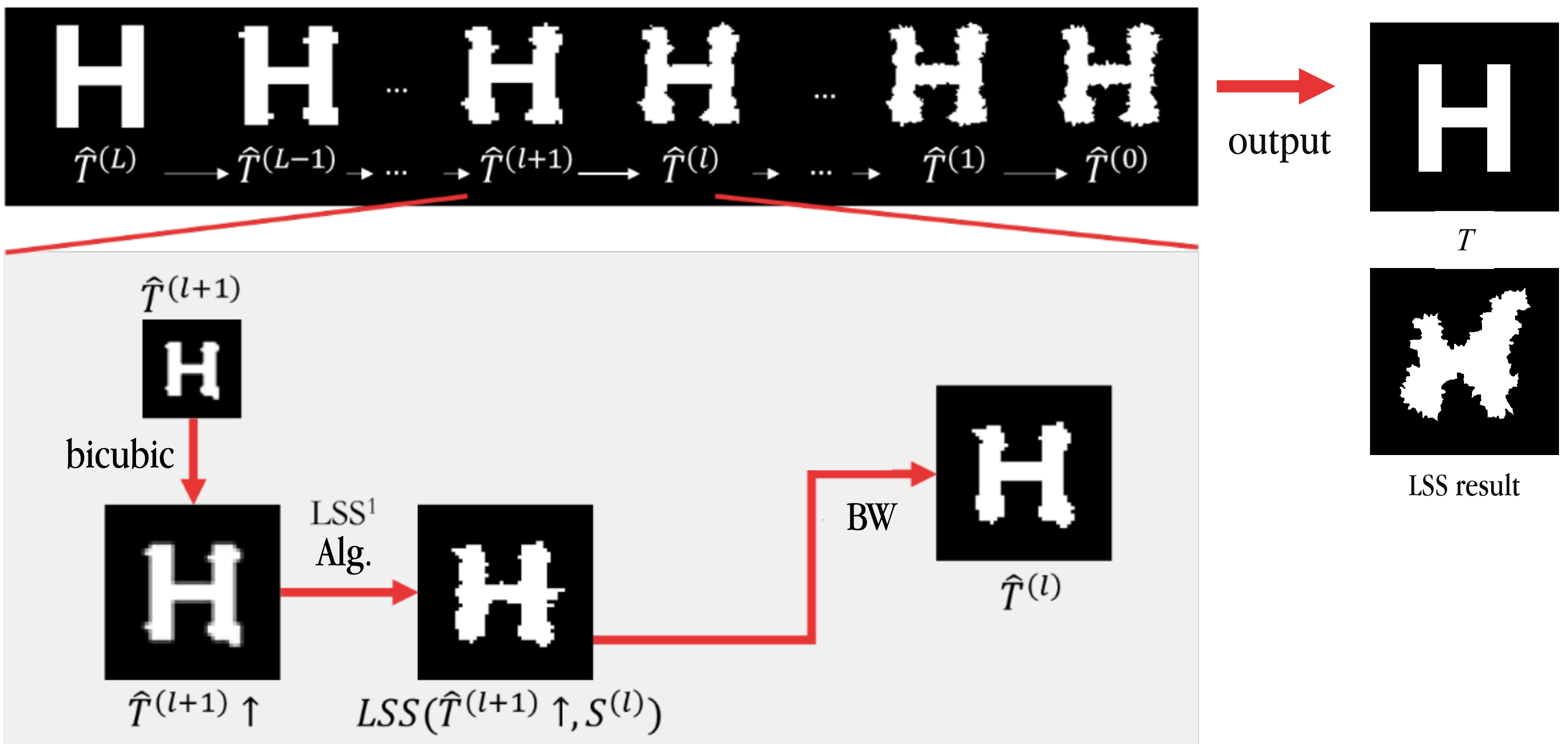
# Unsupervised Text Stylization

- Guidance Map Extraction
- **Structure Transfer: Minimize Structural Inconsistency**
- Texture Transfer



# Unsupervised Text Stylization

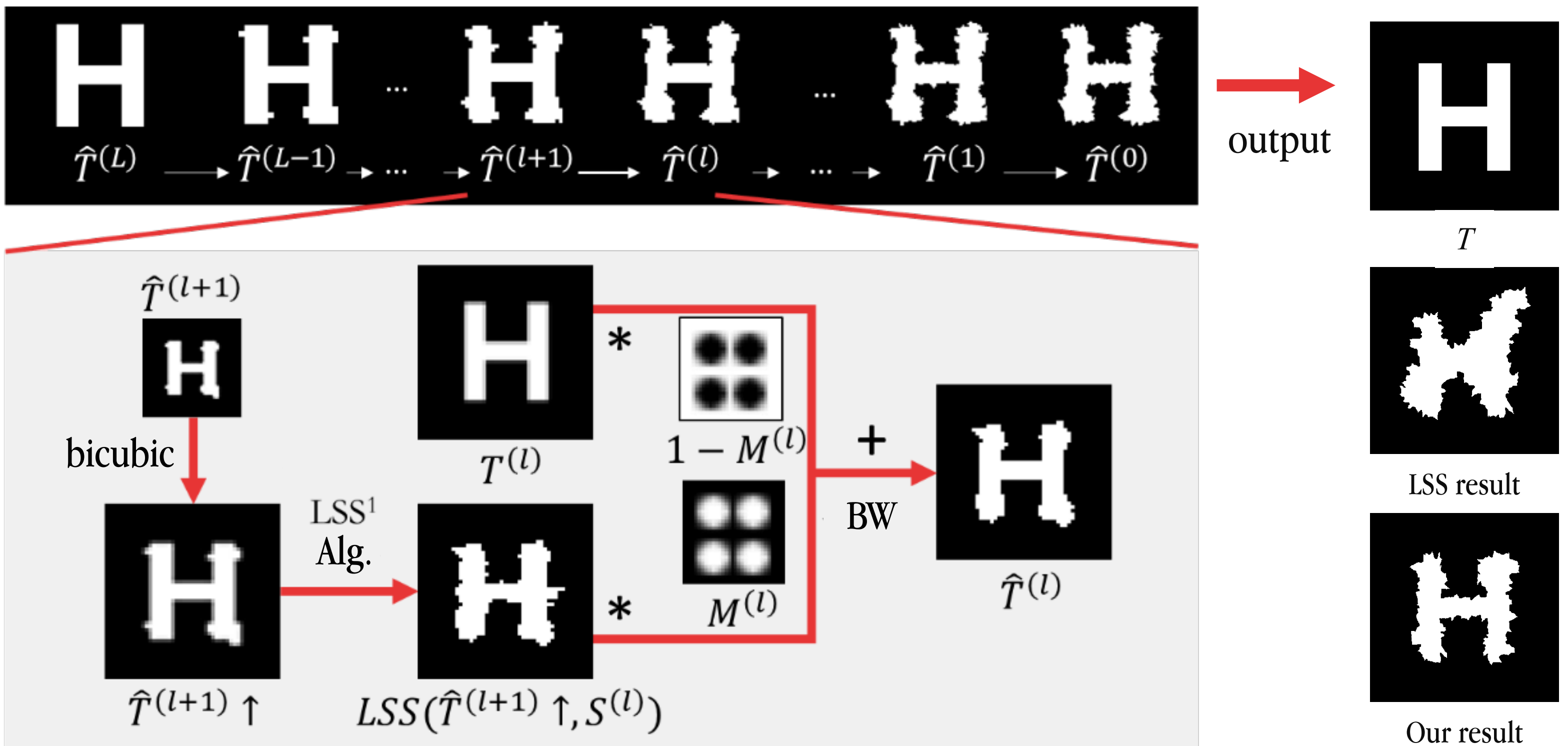
- Guidance Map Extraction
- **Structure Transfer: Legibility-Preserving Structure Transfer**
- Texture Transfer




[1]A. Rosenberger, D. Cohen-Or, and D. Lischinski. Layered shape synthesis: automatic generation of control maps for non-stationary textures. TOG. 2009.

# Unsupervised Text Stylization

- Guidance Map Extraction
- **Structure Transfer: Legibility-Preserving Structure Transfer**
- Texture Transfer



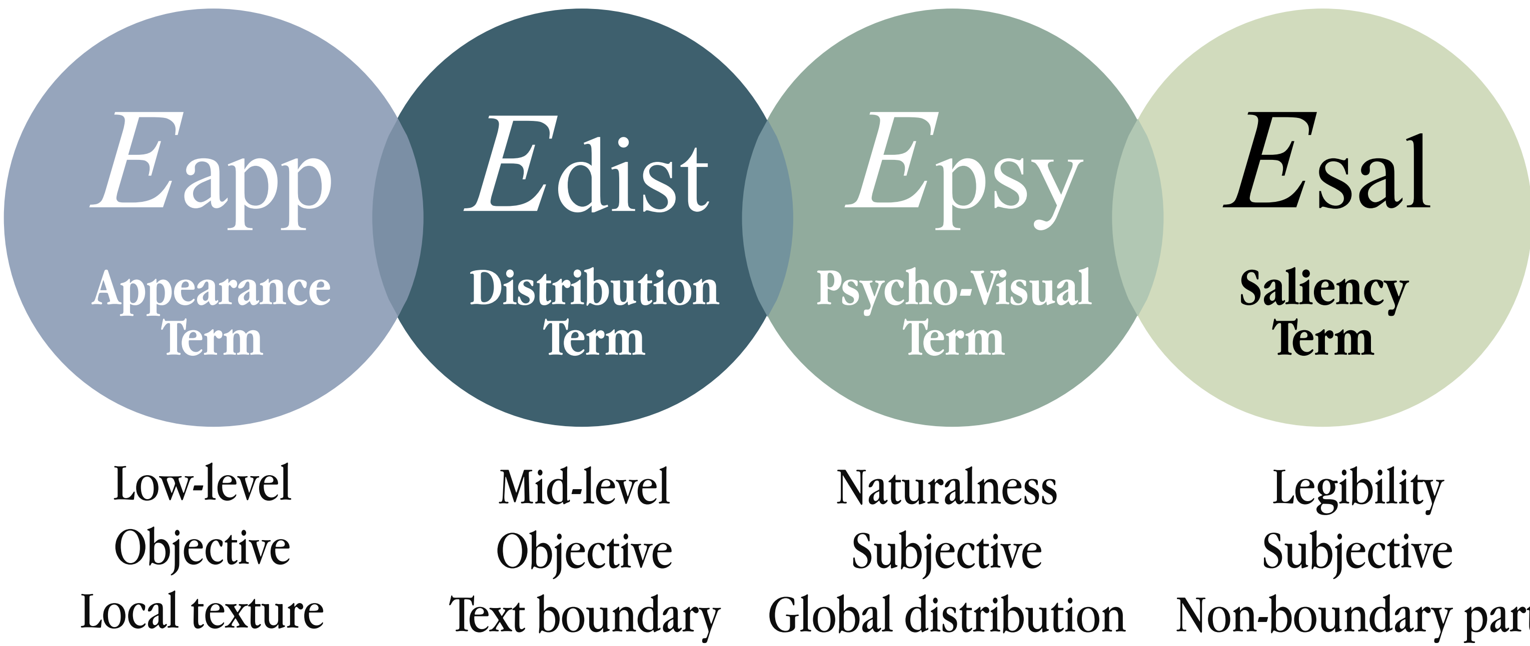

[1] A. Rosenberger, D. Cohen-Or, and D. Lischinski. Layered shape synthesis: automatic generation of control maps for non-stationary textures. TOG. 2009.

# Unsupervised Text Stylization

## Texture Transfer

- Objective Function

$$\min_q \sum_p E_{app}(p,q) + \lambda_1 E_{dist}(p,q) + \lambda_2 E_{psy}(p,q) + \lambda_3 E_{sal}(p,q)$$



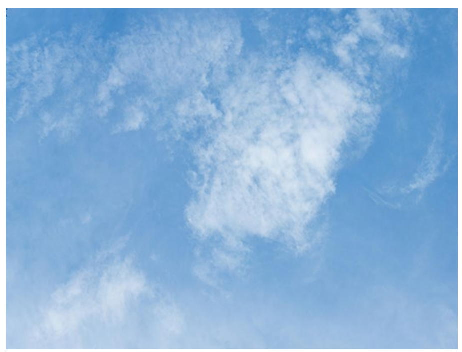
# Texture Transfer

## Saliency Term

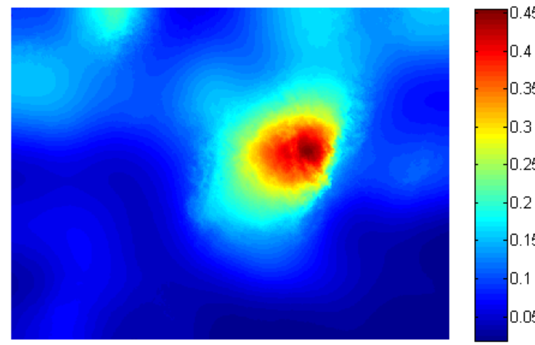
Internal area: salient textures

External area: indistinctive flat textures

**Increase text legibility**



$S'$




Saliency map of $S'$


$\lambda_3$=0.00

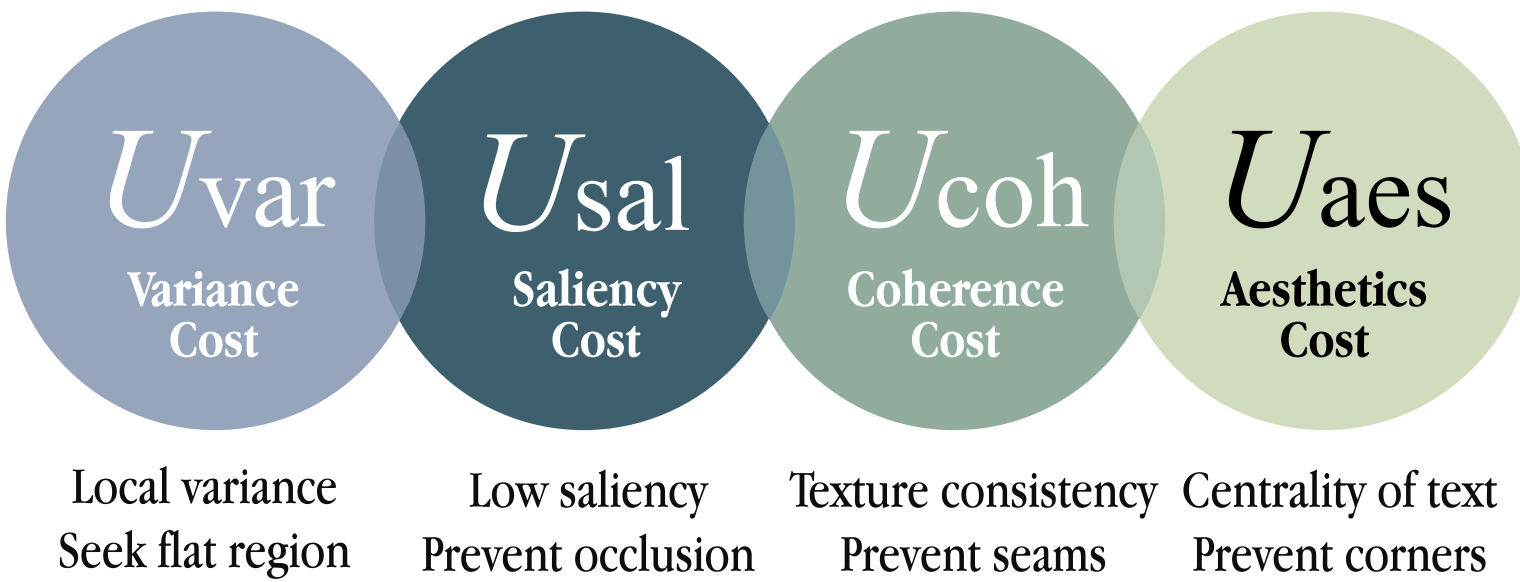

$\lambda_3$=0.01


$\lambda_3$=0.05

## Context-Aware Layout Design

- **Position Estimation**
  - Cost function

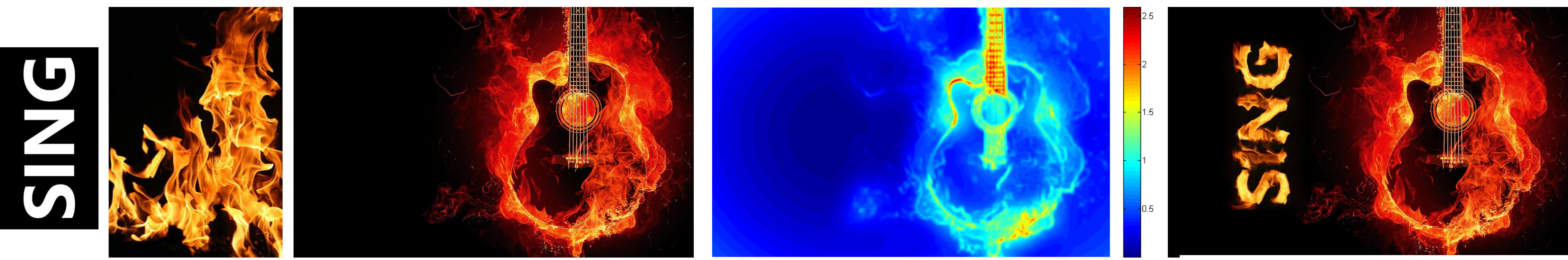
$$\hat{R} = \arg\min_{R} \sum_{p \in R} U_{var}(p) + U_{sal}(p) + U_{coh}(p) + U_{aes}(p)$$



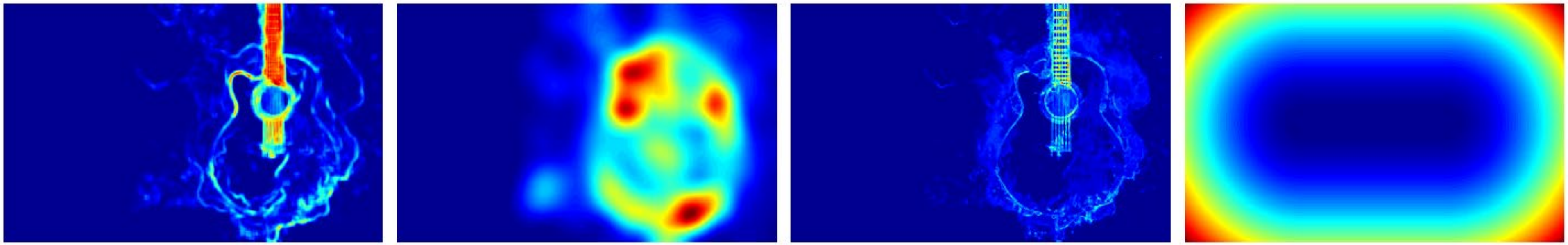
# Context-Aware Layout Design

## Position Estimation

- Cost function




Input $T$ , $S'$ and $I$ Total placement cost Position estimation result



| **Variance Cost** | **Saliency Cost** | **Coherence Cost** | **Aesthetics Cost** |
|---|---|---|---|
| Local variance | Low saliency | Texture consistency | Centrality of text |
| Seek flat region | Prevent occlusion | Prevent seams | Prevent corners |

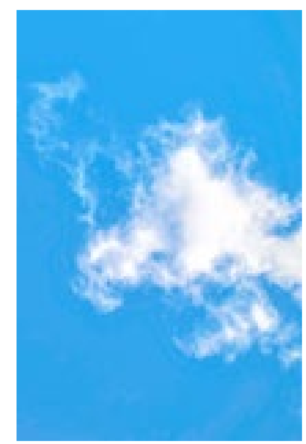
# Context-Aware Layout Design

## Text Embedding

- Image inpainting: fill Inpainting region with samples from $S'$
- Structure guidance of $T$ + Contextual boundary constraint


Contextual region

Inpainting region



Sampling region $S'$

# Comparison with Other Methods


*T* and *S′*

Image Analogy[1]

Neural Doodle[2]

Text Effects Transfer[3]

Image Quilting[4]

Neural Style[5]

CNNMRF[6]

Our method

[1] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. SIGGRAPH. 2001.

[2] A. J. Champandard. Semantic style transfer and turning two-bit doodles into fine artworks. Arxiv. 2016.

[3] S. Yang, J. Liu, Z. Lian, and Z. Guo. Awesome typography: statistics-based text effects transfer. CVPR. 2017.

[4] A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. TOG. 2001.

[5] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. CVPR. 2016

[6] C. Li and M. Wand. Combining Markov random fields and convolutional neural networks for image synthesis. CVPR. 2016

# Comparison with Unsupervised Methods

## Structural Consistency


$S'$ | $T$ | Structure transfer result


Image Quilting[4] | Neural Style[5] | Our method

[4] A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. TOG. 2001.

[5] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. CVPR. 2016

# Comparison with Supervised Methods

## Text Legibility


*T* and *S′*

Image Analogy[1]

Text Effects Transfer[3]


Our method

[1]A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. SIGGRAPH. 2001.

[3]S. Yang, J. Liu, Z. Lian, and Z. Guo. Awesome typography: statistics-based text effects transfer. CVPR. 2017.

## Different Fonts and Languages


Visual effects for text stylization on different languages.

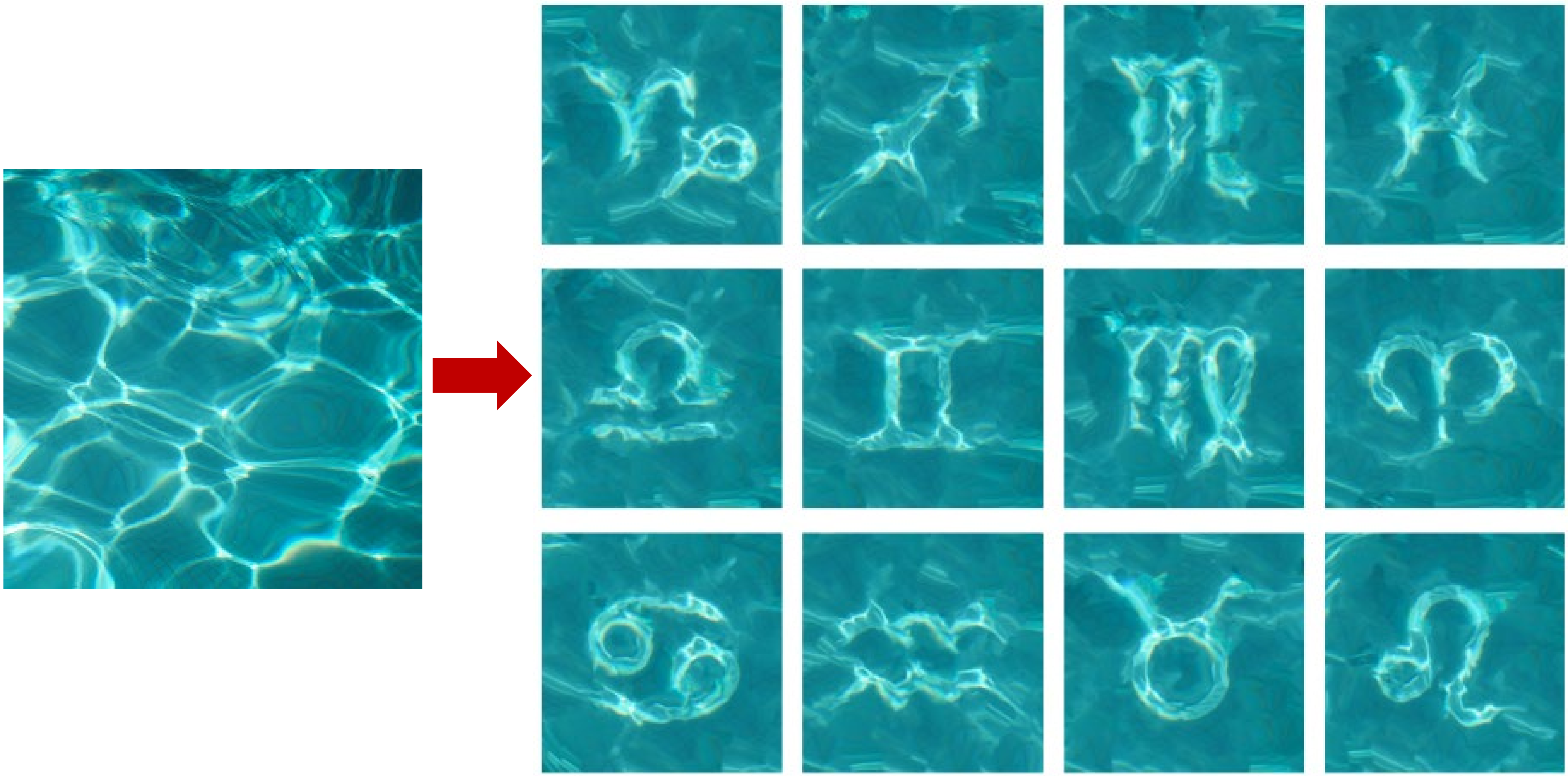

Visual effects for text stylization on different fonts.

# Unsupervised Text Effects Synthesis

- Text effects synthesis results

# Unsupervised Text Effects Synthesis

- Application: texture rendering for symbols



# Stroke-Based Graphics Rendering

## Icon Rendering

Rendering emoji icons with the painting style of Van Gogh using “The Starry Night”

# Graphical Arts: Text + Images

- Synthesize stylish text onto photos
  - Follow image completion process
    - Background images provide boundary information
    - Texture synthesis under boundary constrains

Background image


($T$, $S'$)


Text + Image


Application:
Poster design

# Graphical Arts: Text + Images

- Graphical arts design results (1/2)
  - $S'$ is sampled from background images

# Graphical Arts: Text + Images

- Graphical arts design results (2/2)
  - $S'$ is not from background images: color transfer is used

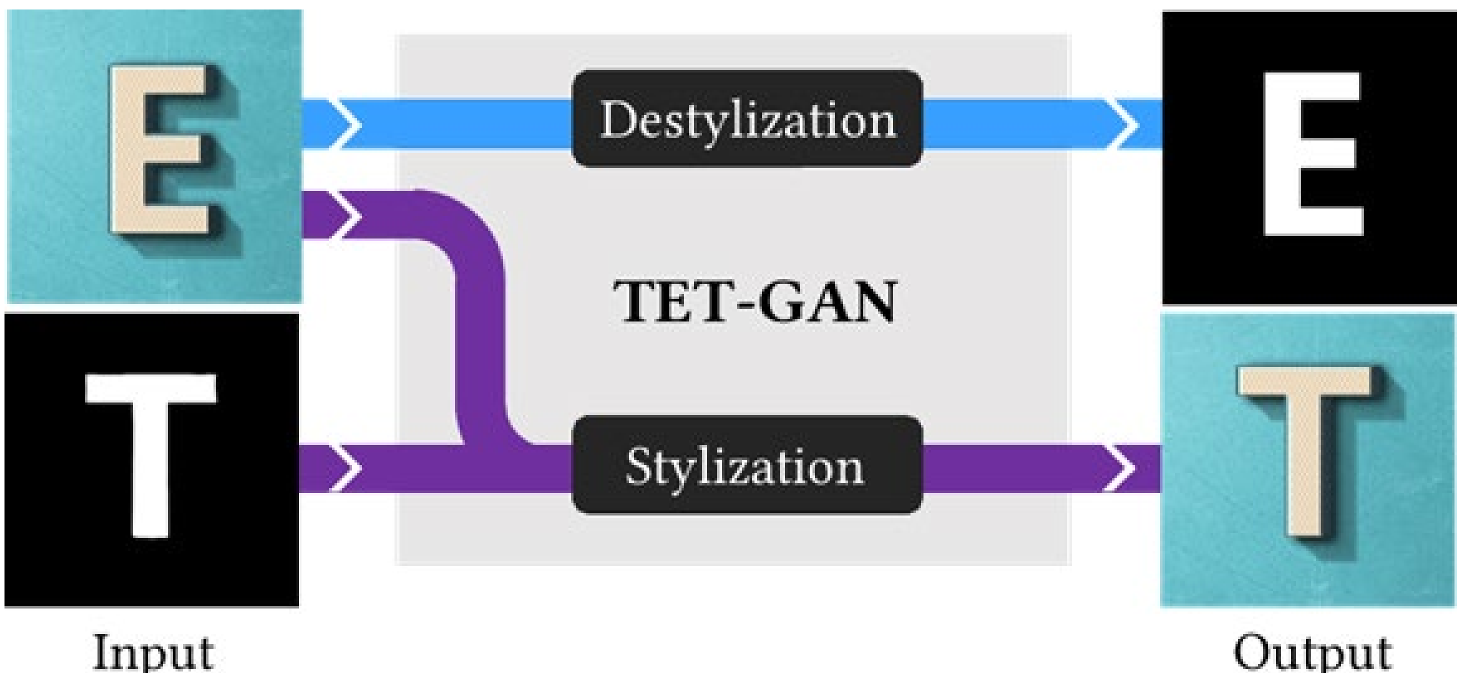
## Deep Learning Based Text Stylization

# Text Effects Transfer via Stylization and Destylization

Shuai Yang, Jiaying Liu, Wenjin Wang, Zongming Guo, **AAAI 2019**

## Problem: Text Stylization and Destylization

- Task1: **Stylization** for transferring the visual effects from highly stylized text onto other glyphs
- Task2: **Destylization** for removing style features from text



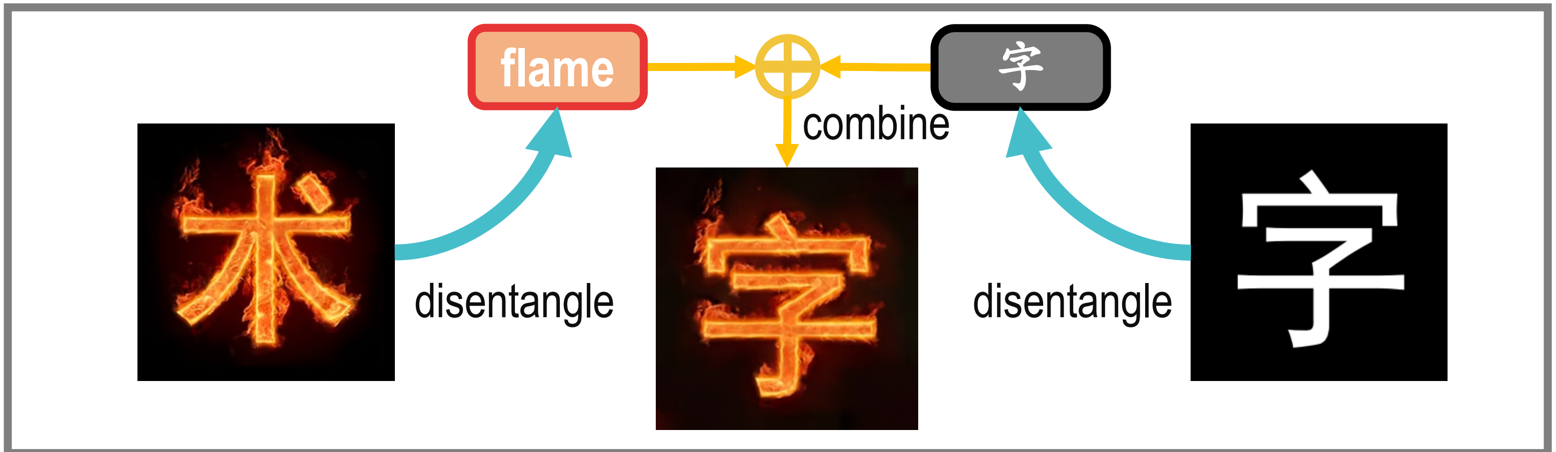
# Problem Analysis

## Methods

- Text effects are highly structured along the glyph

| | Global-Based | Local-Based | GAN |
|---|---|---|---|
| pros | Need no dataset | Preserve texture details<br>Need no dataset | Reconstruct vivid styles |
| cons | Text effects cannot be simply characterized as the mean, variance, covariance and other global statistics | Fail to characterize global distribution | Need dataset for training<br>Deals only two domains or<br>Hard to extend to new style |
| e.g. |   |   |   |

## Problem Analysis




**DATA**

- 64 different style
- 775 Chinese characters
- 52 English letters
- 10 Arabic numerals

**METHOD**

- Feature disentanglement
- Feature recombination
- One-shot learning

**APPLICATION**

- Stylization
- Destylization
- Style creation

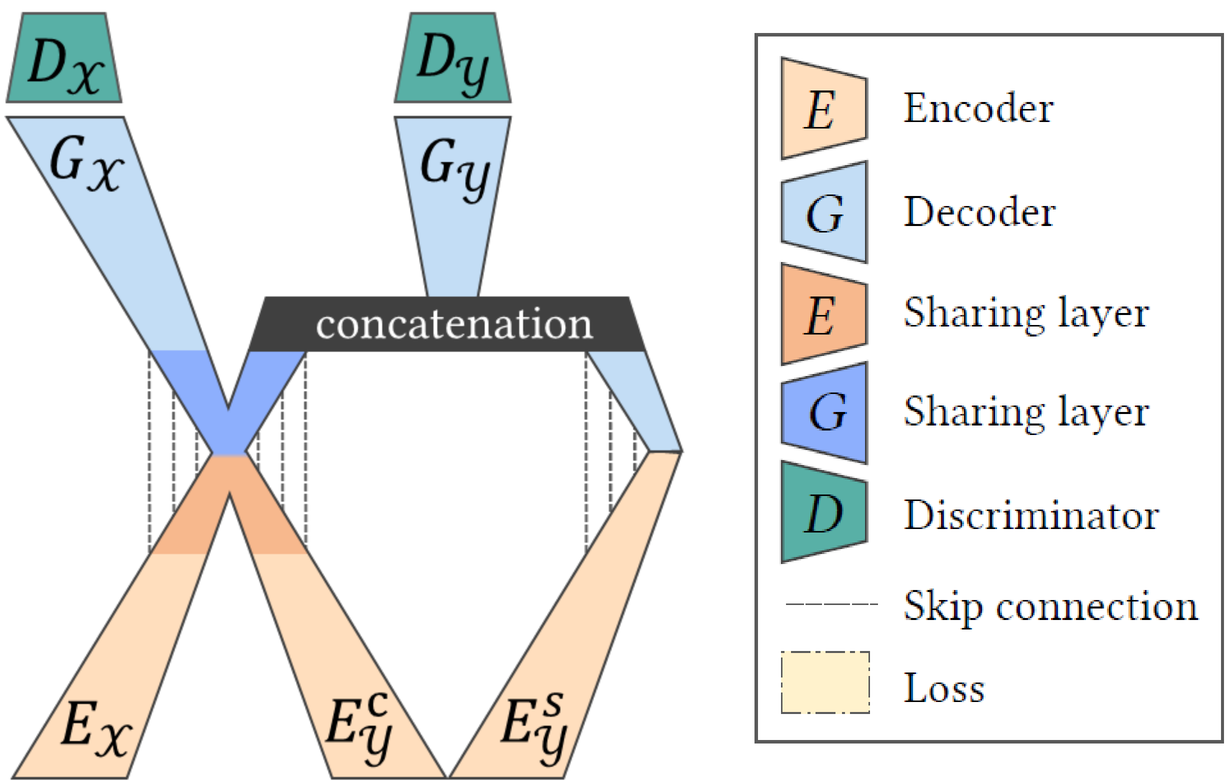
# Data Overview

## Dataset

- 64 different text effects
- 775 Chinese characters, 52 English letters, 10 Arabic numerals
- 64 * 837 $\approx$ 54K pairs
- Train : Test = 708 : 129

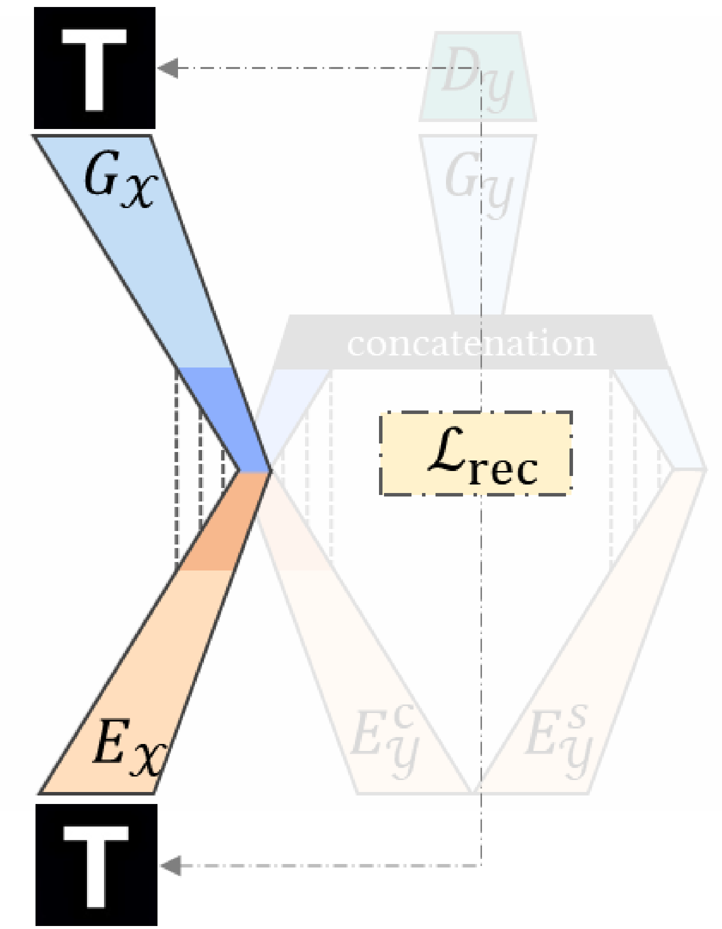
# Network Architecture

- Learn a two-way mapping between two visual domains X (text) and Y (text effects)




- **Encoder:**
  Text content encoder
  Style content encoder
  Style encoder
- **Decoder:**
  Content decoder
  Style decoder
- **Discriminator:**
  Content discriminator
  Style discriminator

## Network Architecture

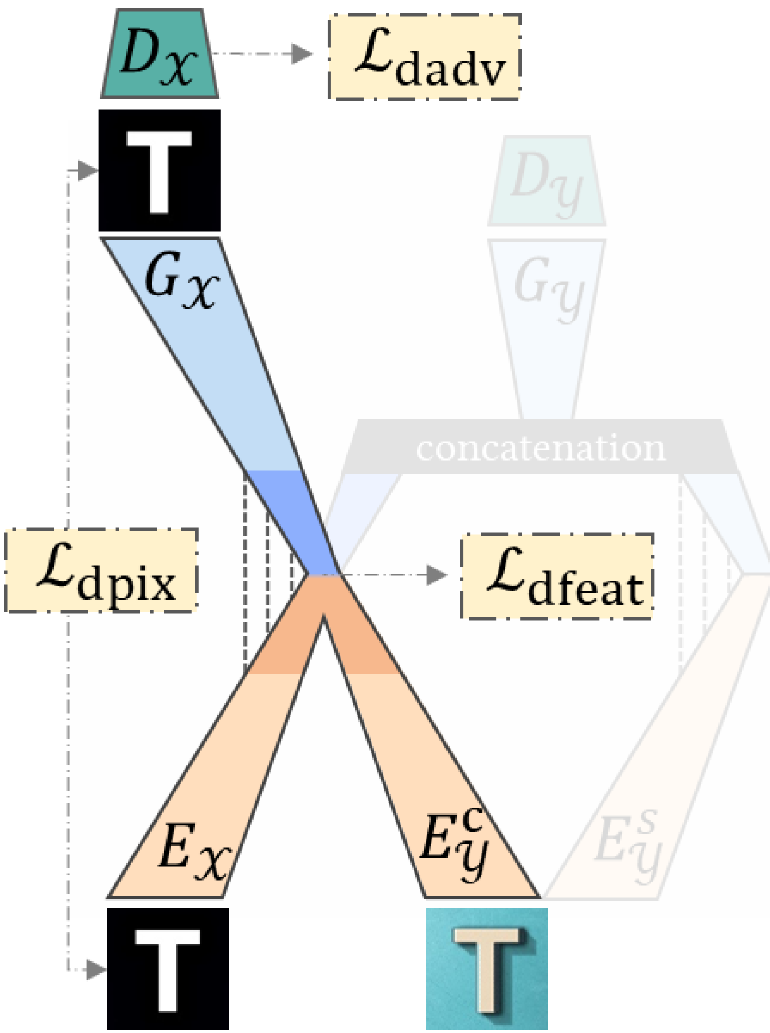



- **Autoencoder**
  - Reconstruction loss

$$\mathcal{L}_{\text{rec}} = \lambda_{\text{rec}} \mathbb{E}_x [\| G_{\mathcal{X}}(E_{\mathcal{X}}(x)) - x \|_1]$$

  - Preserve the core information of the glyph

# Network Architecture




## Destylization

- Feature loss

  $$\mathcal{L}_{\text{dfeat}} = \mathbb{E}_{x,y}[\|S_{\mathcal{X}}(E^c_{\mathcal{Y}}(y)) - z\|_1]$$

  - Approach ground truth content feature extracted by Ex

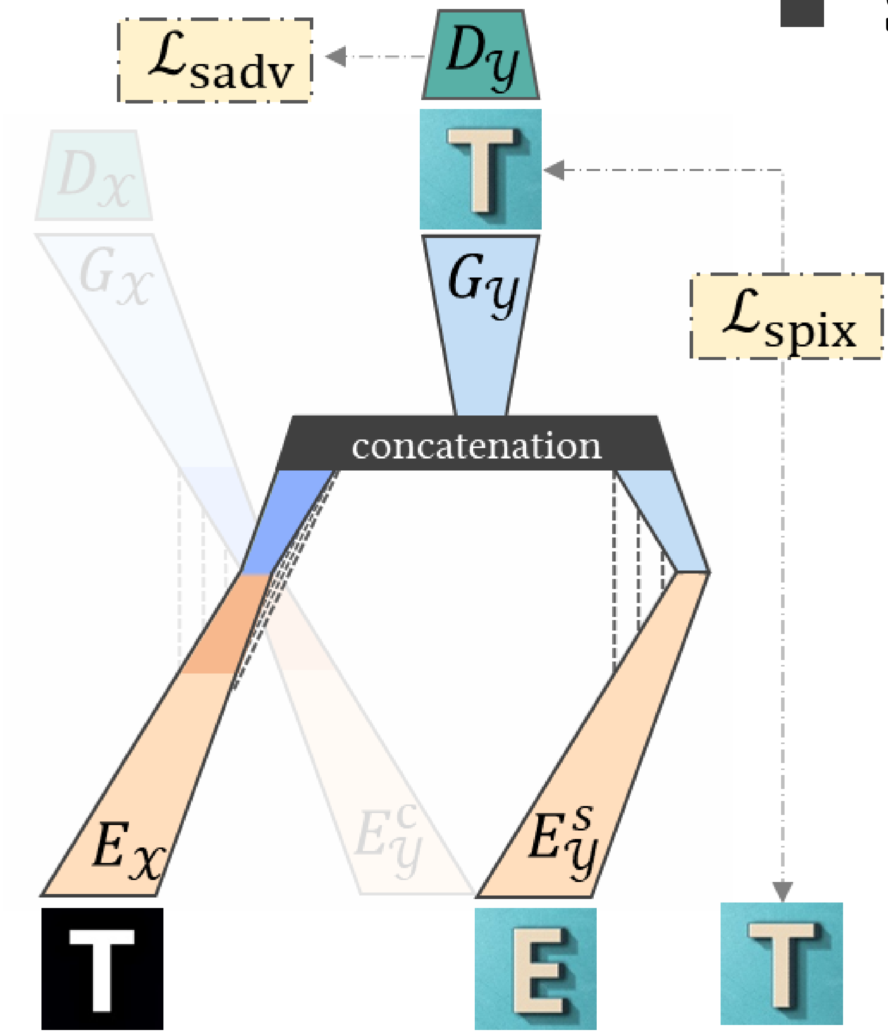
- Pixel loss

  $$\mathcal{L}_{\text{dpix}} = \mathbb{E}_{x,y}[\|G_{\mathcal{X}}(E^c_{\mathcal{Y}}(y)) - x\|_1]$$

  - Approach ground truth output

- Adversarial loss
  - WGAN-GP
  - Improve the quality

# Network Architecture




## Stylization

- Pixel loss

$$\mathcal{L}_{\text{spix}} = \mathbb{E}_{x,y,y'}[\|G_{\mathcal{Y}}(E_{\mathcal{X}}(x), E_{\mathcal{Y}}^{s}(y')) - y\|_1]$$

  - Approach ground truth output
- Adversarial loss
  - WGAN-GP
  - Improve the quality

# One-shot fine-tuning (supervised)

- **Extend to new style with only one example pair**
  - Random crop to generate multiple training pairs
  - More precise texture details

(a) (b) (c)

(d) self-stylization training scheme

(a) User-specified new text effects.
(b) Result on an unseen style
(c) Result after one-shot fine-tuning

## One-shot fine-tuning (unsupervised)

- **Extend to new style with only one style image**
  - Use destylization network to generate auxiliary raw text
  - Style autoencoder reconstruction loss

# Comparison


Figure 6: Comparison with state-of-the-art methods on various text effects. (a) Input example text effects with the target text in the lower-left corner. (b) Our destylization results. (c) Our stylization results. (d) pix2pix-cGAN (Isola et al. 2017). (e) StarGAN (Choi et al. 2018). (f) T-Effect (Yang et al. 2017). (g) Neural Doodles (Champandard 2016). (h) Neural Style Transfer (Gatys, Ecker, and Bethge 2016).

# Comparison

- One-shot supervised style transfer
  - Pretraining on our dataset successfully teaches our network the domain knowledge of text effects synthesis

(a) input (b) TET-GAN (c) TET-GAN2 (d) T-Effect (e) Doodle

Fine-tune Train from scratch

# Comparison

- One-shot unsupervised style transfer


(a) input (b) TET-GAN (c) Quilting (d) NST (e) CNNMRF

## Other results

- Style interpolation

# ● 形变程度可控的文字风格化技术

## Controllable Artistic Text Style Transfer via Shape-Matching GAN

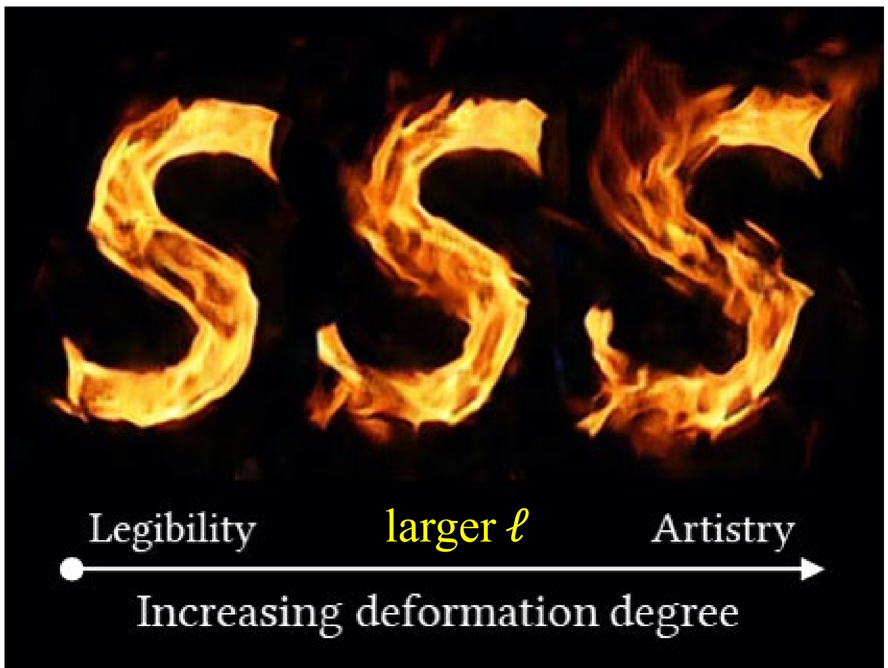
*Shuai Yang, Zhangyang Wang, Zhaowen Wang, Ning Xu, Jiaying Liu and Zongming Guo*, ICCV 2019

## Problem: Controllable Text Style Transfer

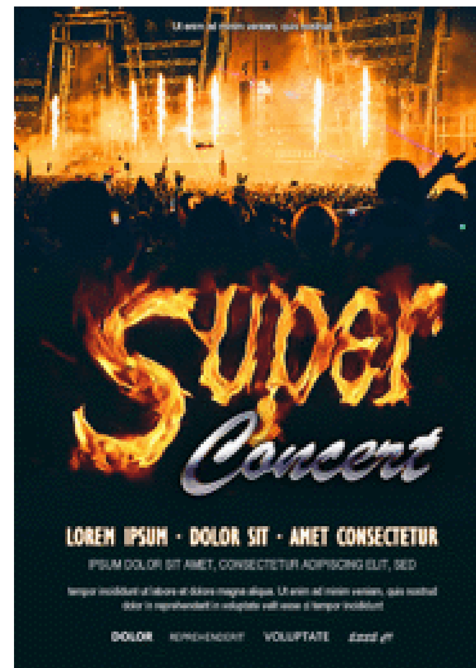

- Input: $Y$, $T$, $\ell$; Output: $T_\ell^Y$
- $\ell$: Deformation degree $\ell$
- Large $\ell$ → more artistry; less legibility: balance?



Input



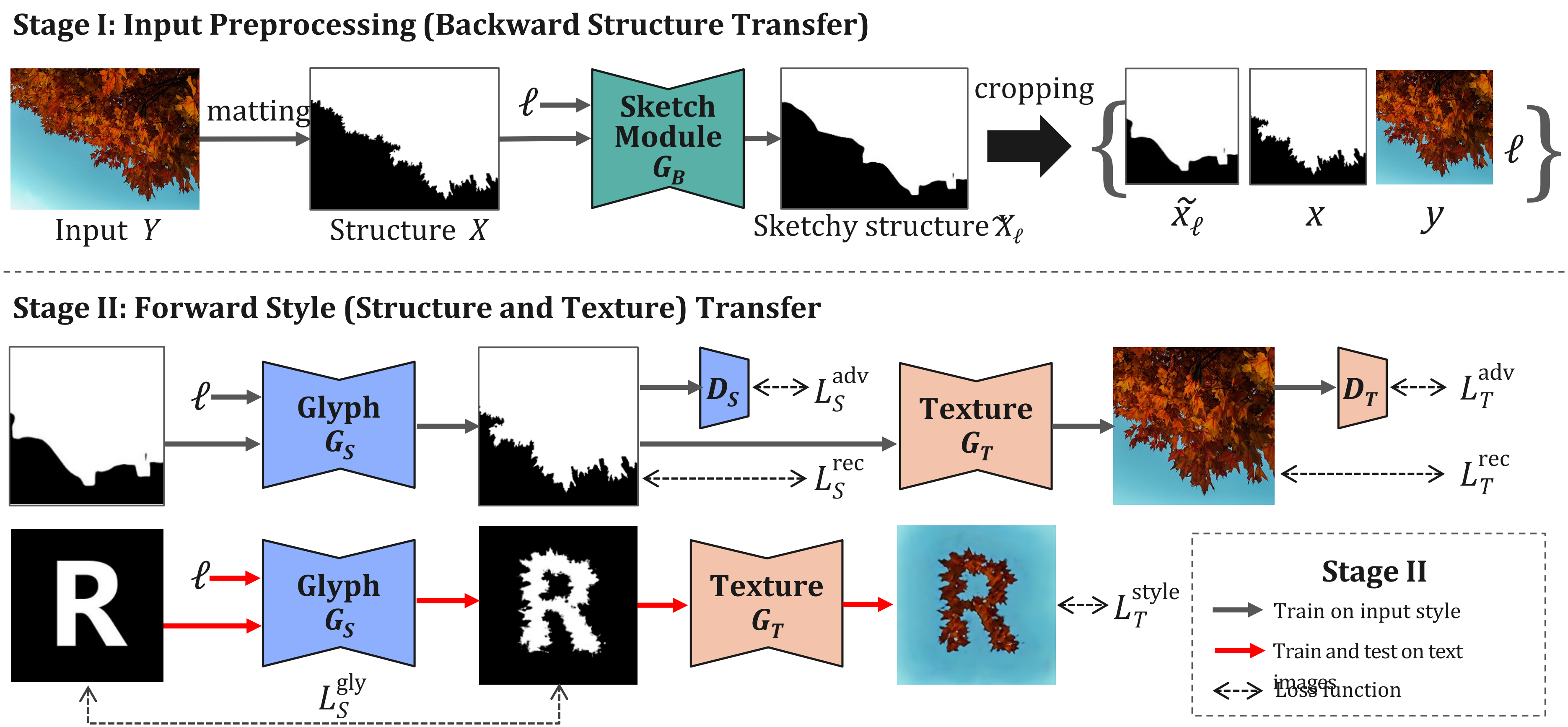

Adjust the stylistic degree of glyph




Application

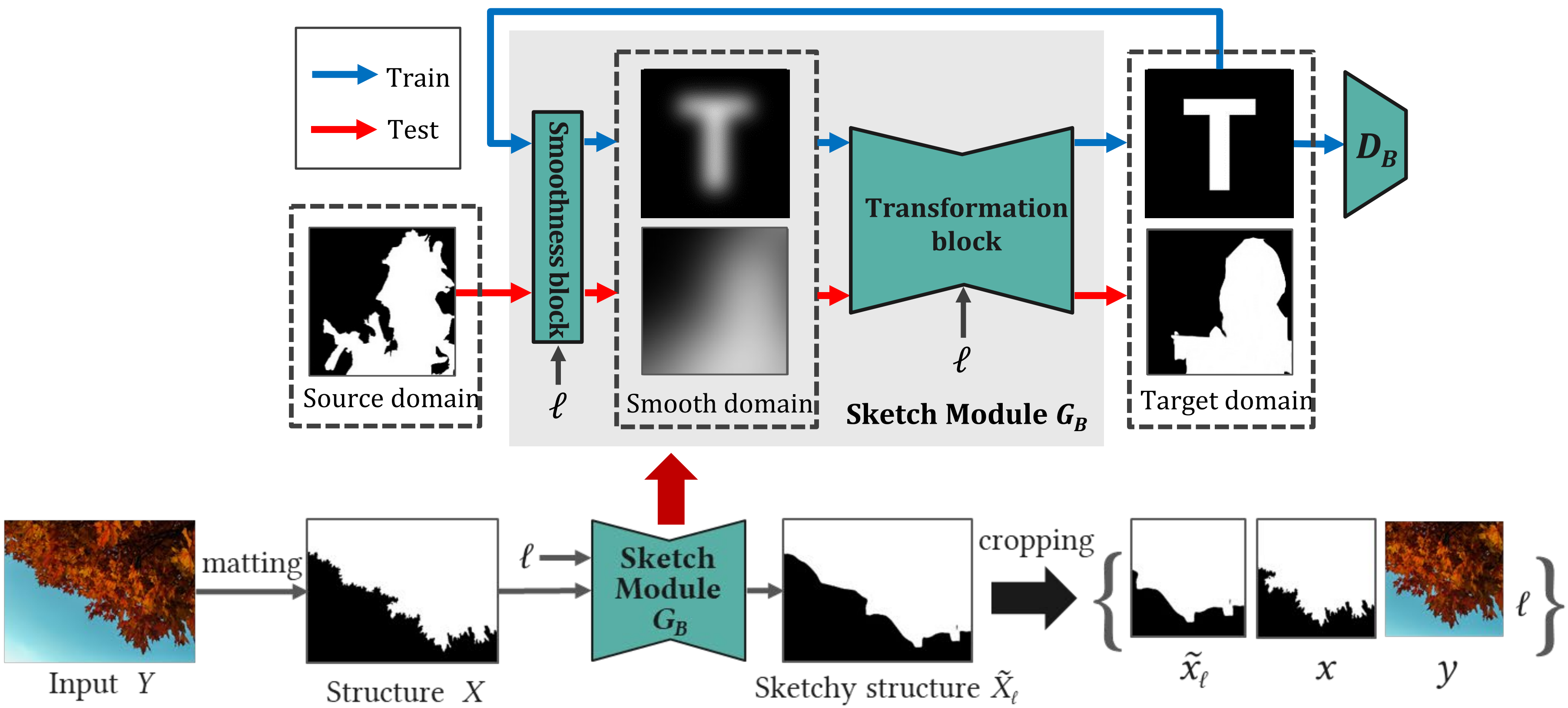
# Framework

- Stage I: prepare training data
- Stage II: style transfer

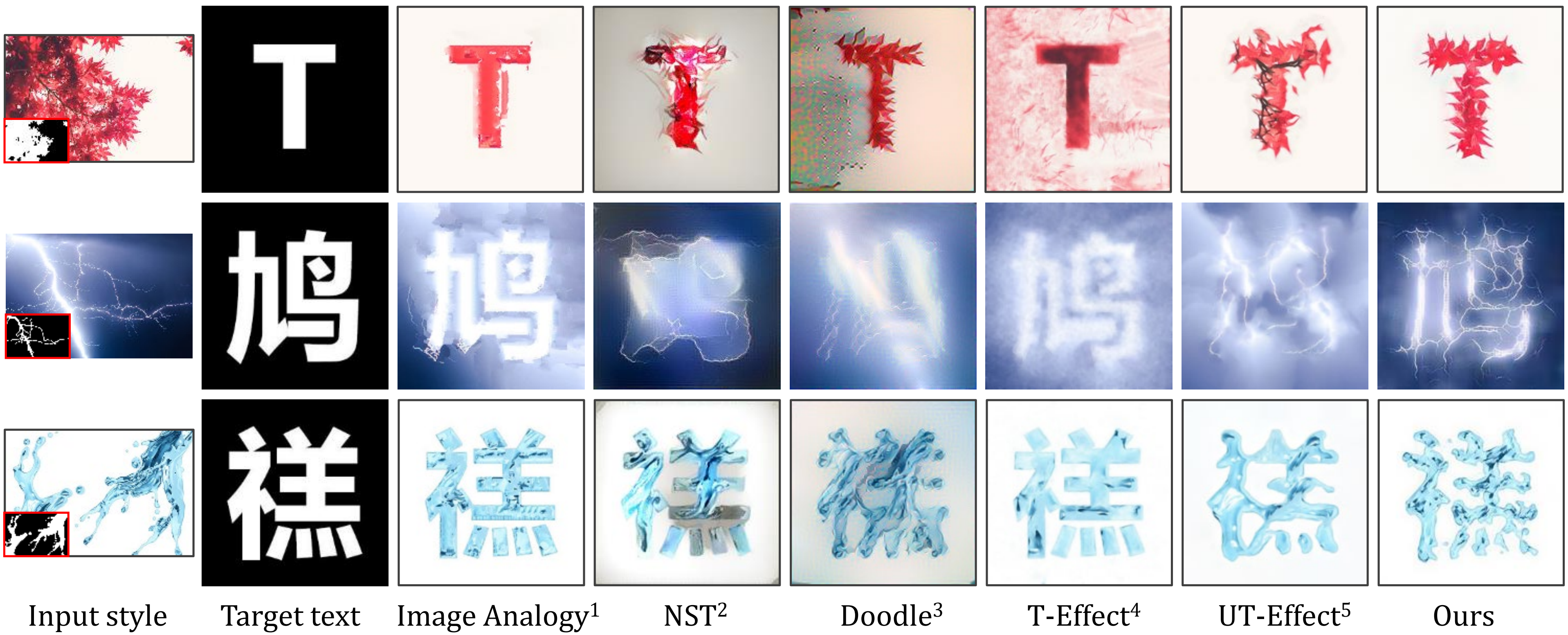


# Backward Structure Transfer ($G_B$)

- Gaussian blur to maps $T$ and $X$ into a smooth domain
- Train CNN to map the smoothed image back to the text domain
- The standard deviation of Gaussian kernel is controlled by $\ell$



# Comparison with Other Methods



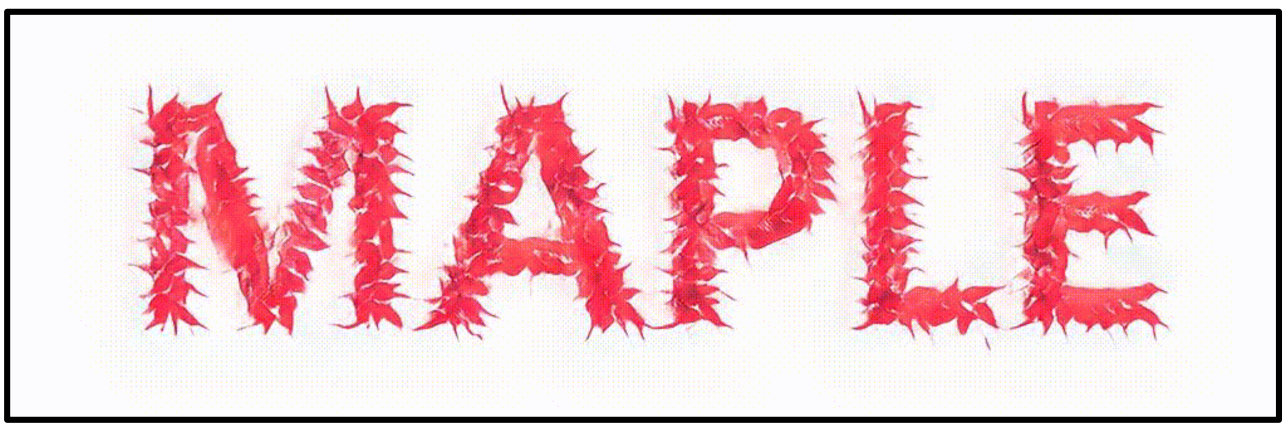

[1] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. SIGGRAPH. 2001
[2] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using Convolutional neural networks. CVPR. 2016
[3] A. J. Champandard. Semantic style transfer and turning two-bit doodles into fine artworks. Arxiv. 2016
[4] S. Yang, J. Liu, Z. Lian, and Z. Guo. Awesome typography: statistics-based text effects transfer. CVPR. 2017
[5] S. Yang, J. Liu, W. Yang, and Z. Guo. Context-aware text-based binary image stylization and synthesis. TIP. 2019

## Scale-Controllable Style Transfer

Reference style

MAPLE

Target text




legible — stylish

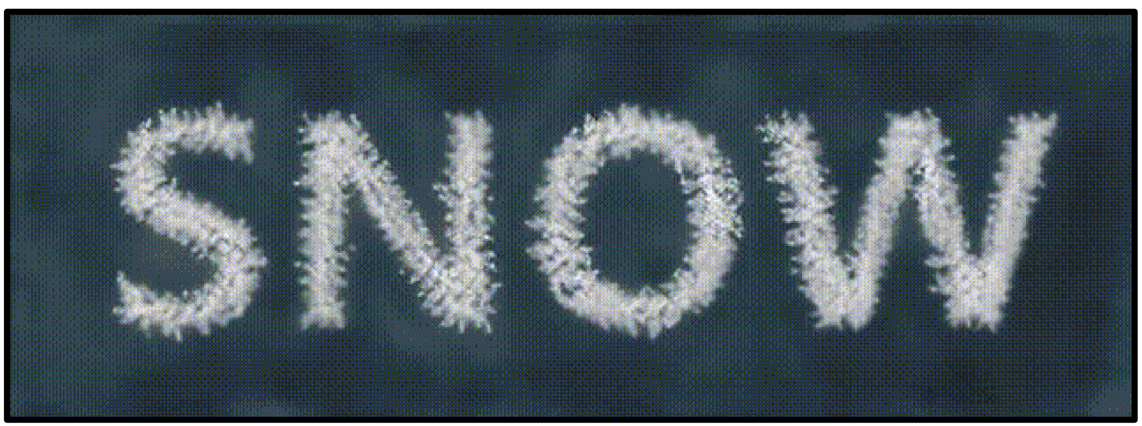
Adjusting glyph deformation degree

## Scale-Controllable Style Transfer

Reference style

SNOW

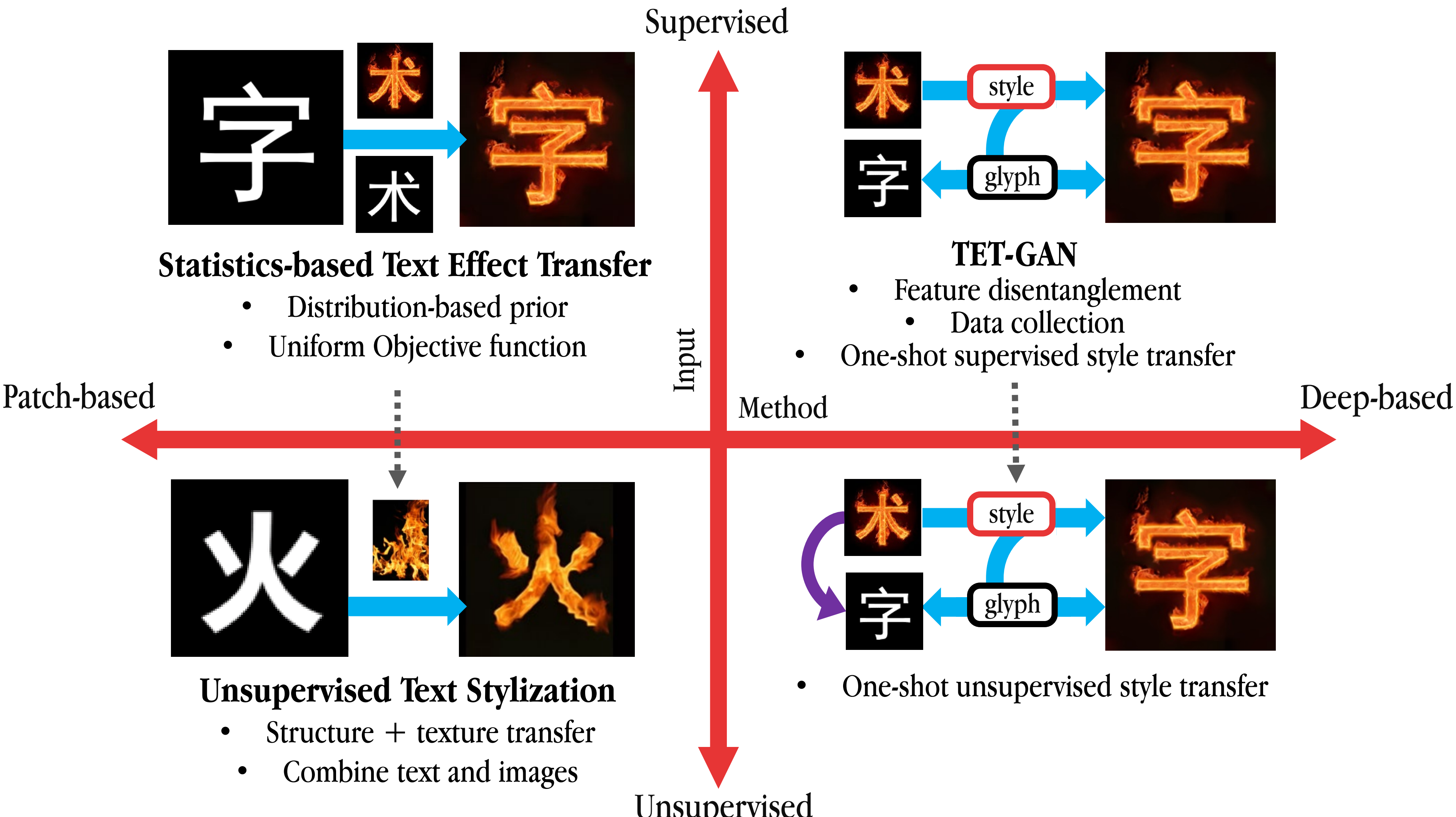
Target text




legible stylish

Adjusting glyph deformation degree

# Conclusion


Supervised
Unsupervised
Patch-based
Deep-based
Input
Method
字
术
Statistics-based Text Effect Transfer
• Distribution-based prior
• Uniform Objective function
style
glyph
TET-GAN
• Feature disentanglement
• Data collection
• One-shot supervised style transfer
火
Unsupervised Text Stylization
• Structure + texture transfer
• Combine text and images
• One-shot unsupervised style transfer