

# Generating Handwriting via Decoupled Style Descriptors



Atsunobu Kotani



Stefanie Tellex



James Tompkin



BROWN  
Computer Science









**Target**

gether with the

young offende

earth, enrit

**Ours**

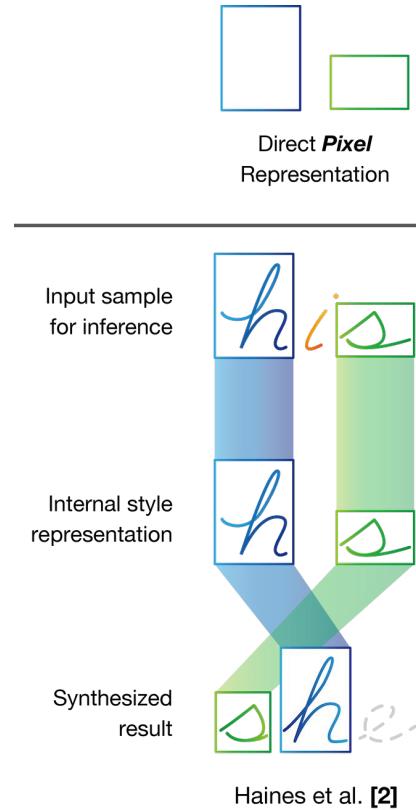
gether with the

young offende

earth, enrit

## Handwriting generation:

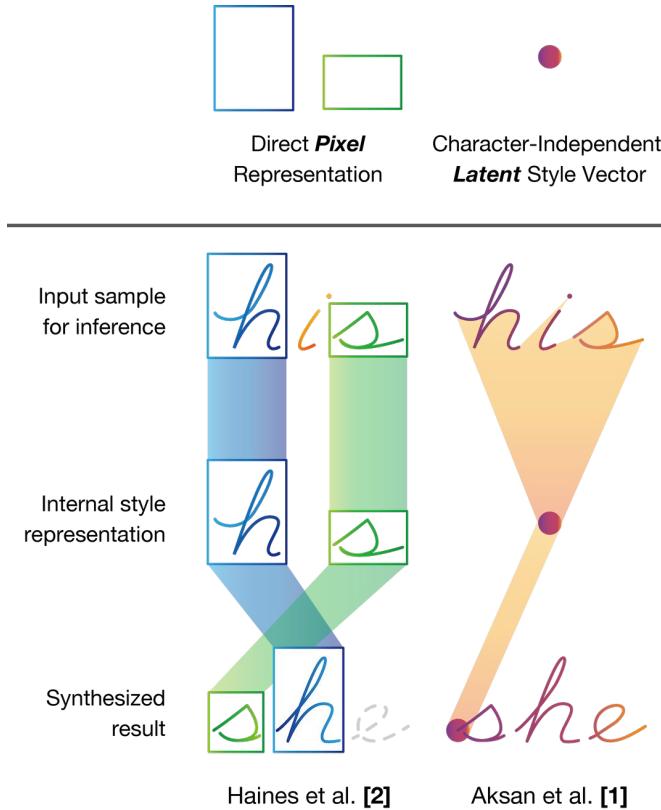
- Pixel representation [2]



- [1] Aksan, E., Pece, F., Hilliges, O.: DeepWriting: Making Digital Ink Editable via Deep Generative Modeling. SIGCHI (2018).  
[2] Haines, T.S.F., Mac Aodha, O., Brostow, G.J.: My text in your handwriting. ACM Trans. Graph. 35(3) (May 2016).

## Handwriting generation:

- Pixel representation [2]
- Learned by neural networks [1]

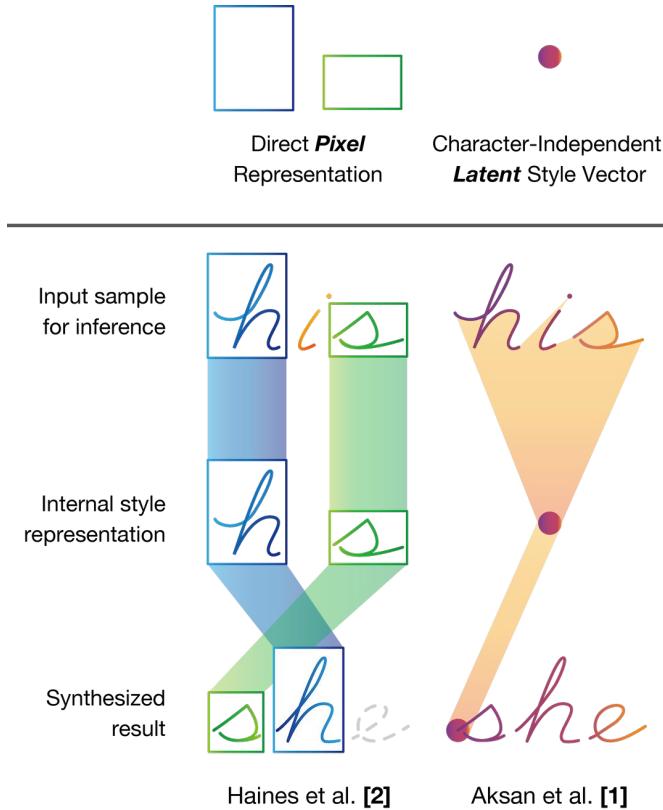


[1] Aksan, E., Pece, F., Hilliges, O.: DeepWriting: Making Digital Ink Editable via Deep Generative Modeling. SIGCHI (2018).

[2] Haines, T.S.F., Mac Aodha, O., Brostow, G.J.: My text in your handwriting. ACM Trans. Graph. 35(3) (May 2016).

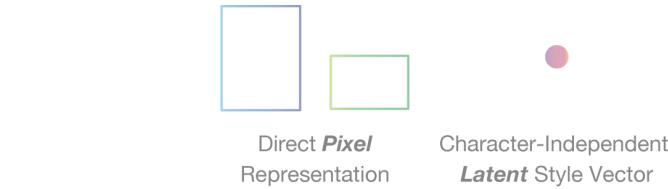
## Handwriting generation:

- Pixel representation [2]
- Learned by neural networks [1]

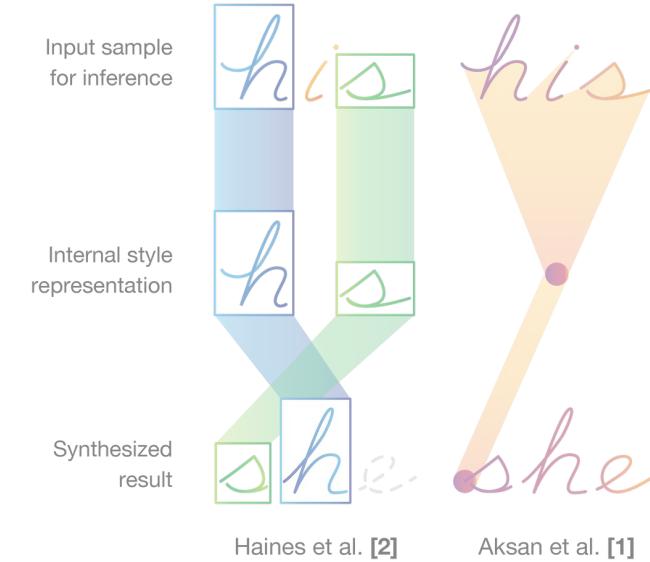


[1] Aksan, E., Pece, F., Hilliges, O.: DeepWriting: Making Digital Ink Editable via Deep Generative Modeling. SIGCHI (2018).

[2] Haines, T.S.F., Mac Aodha, O., Brostow, G.J.: My text in your handwriting. ACM Trans. Graph. 35(3) (May 2016).



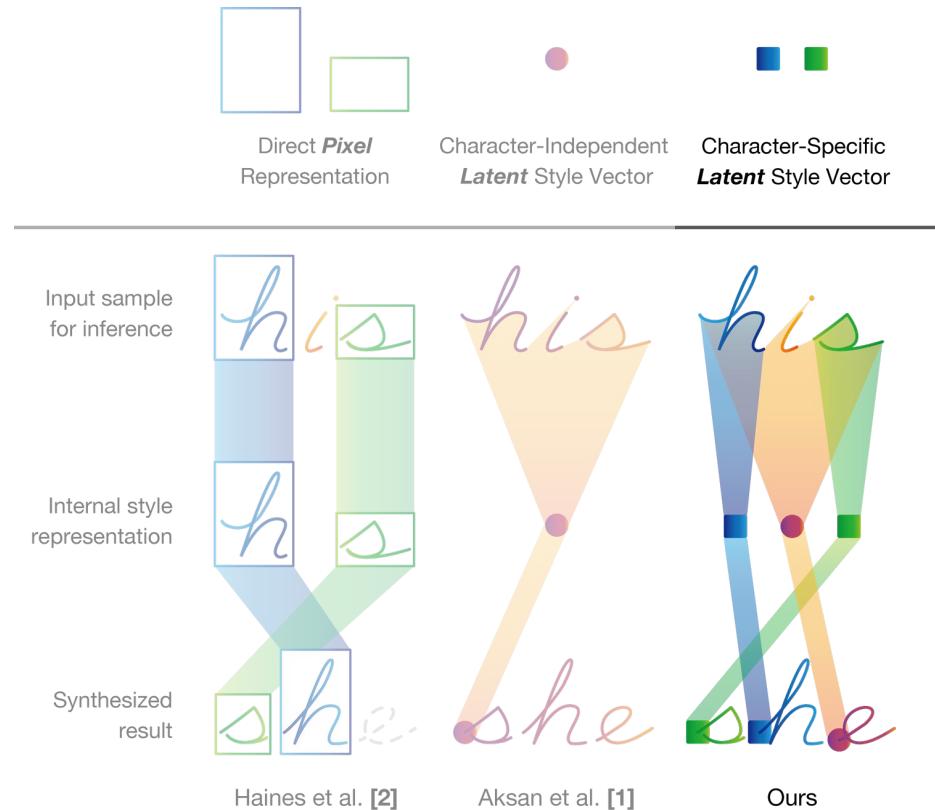
**Underlying problem:**  
No explicit separation of writer style  
from character style.



[1] Aksan, E., Pece, F., Hilliges, O.: DeepWriting: Making Digital Ink Editable via Deep Generative Modeling. SIGCHI (2018).

[2] Haines, T.S.F., Mac Aodha, O., Brostow, G.J.: My text in your handwriting. ACM Trans. Graph. 35(3) (May 2016).

We learn to decouple writer and character style into specific vectors.



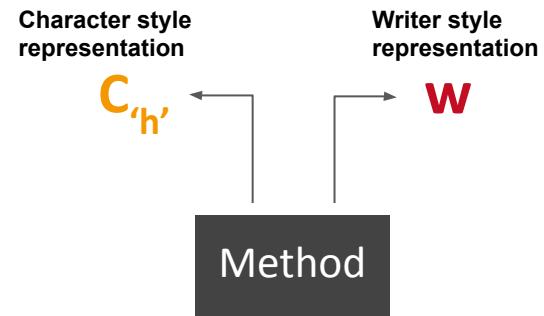
[1] Aksan, E., Pece, F., Hilliges, O.: DeepWriting: Making Digital Ink Editable via Deep Generative Modeling. SIGCHI (2018).

[2] Haines, T.S.F., Mac Aodha, O., Brostow, G.J.: My text in your handwriting. ACM Trans. Graph. 35(3) (May 2016).

# Problem statement

Desired output data:

- *Writer-independent* character style representation  $C_{h'}$
- *Character-independent* writer style representation  $w$



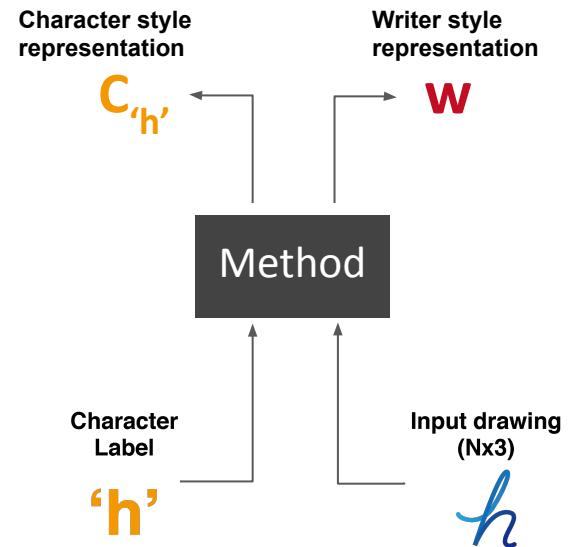
# Problem statement

Desired output data:

- *Writer-independent* character style representation  $C_{h'}$
- *Character-independent* writer style representation  $w$

Input data:

- Strokes as point sequences ( $x, y, t$ )
- Character labels as one-hot vectors



# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_h$

# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_h$

Character-DSD  
(256x256 Matrix)

$C_h$



Character  
Label

‘h’

# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_h$

Character-DSD  
(256x256 Matrix)      Writer-DSD  
(256x1 Vector)

$$C_h \quad w =$$

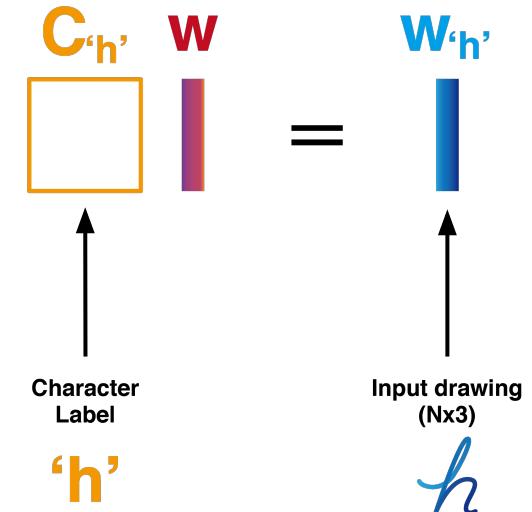
The diagram illustrates the linear relationship learned between a Character-DSD matrix  $C_h$  and a Writer-DSD vector  $w$ . The equation  $C_h \quad w =$  is shown. Below the equation, an arrow points from the character label 'h' to the matrix  $C_h$ , indicating that the character label serves as input to the matrix.

# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_{h'}$

Character-DSD  
(256x256 Matrix)      Writer-DSD  
(256x1 Vector)      Writer-Character-DSD  
(256x1 Vector)



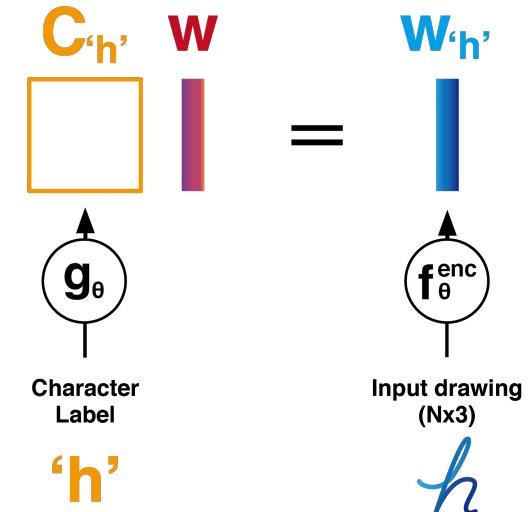
# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_{h'}$ ,

through learned LSTM encoders  $g_\theta$  and  $f_\theta$ .

Character-DSD  
(256x256 Matrix)      Writer-DSD  
(256x1 Vector)      Writer-Character-DSD  
(256x1 Vector)



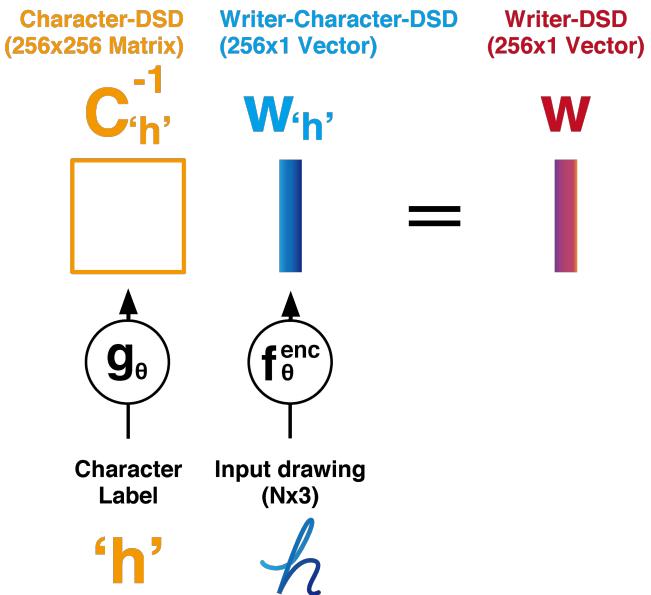
# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

- Writer-DSD  $w$  and
- Character-DSD  $C_{h'}$ ,

through learned LSTM encoders  $g_\theta$  and  $f_\theta$ .

*Simply invert  $C_{h'}$  to recover  $w$  from  $w_{h'}$ .*



# Decoupled Style Descriptors (DSD)

We learn a *linear* relationship between

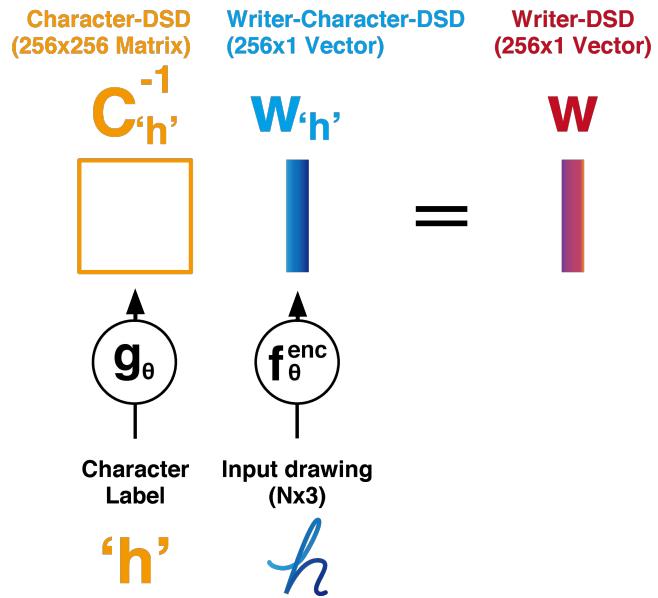
- Writer-DSD  $w$  and
- Character-DSD  $C_{h'}$ ,

through learned LSTM encoders  $g_\theta$  and  $f_\theta$ .

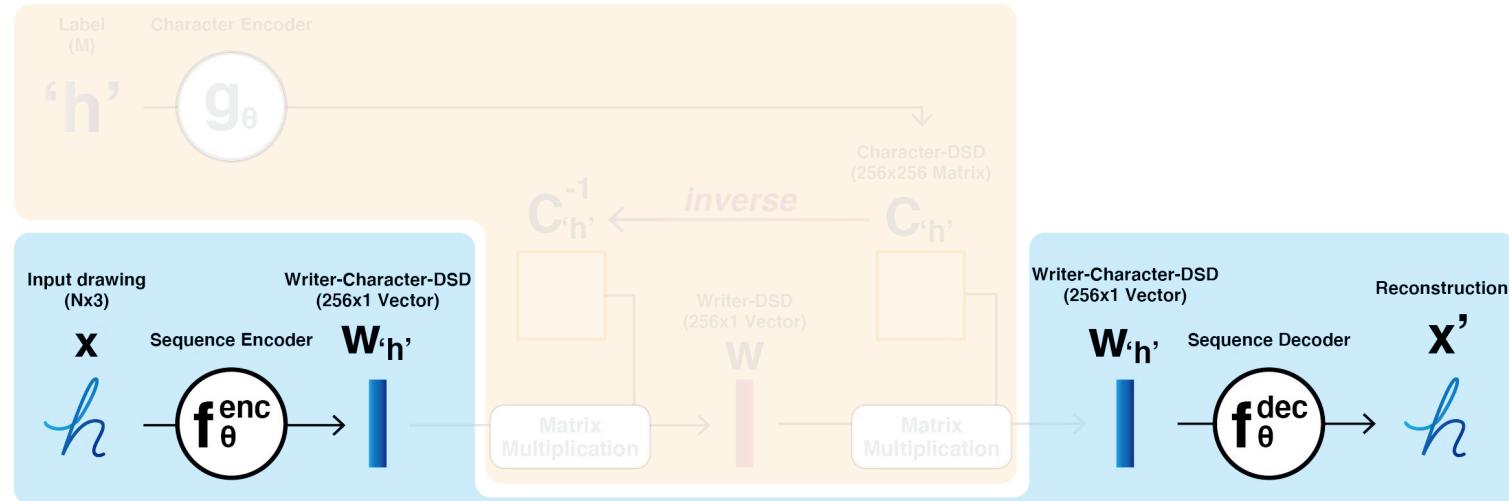
*Simply invert  $C_{h'}$  to recover  $w$  from  $w_{h'}$ .*

Retains *more fine detail*.

Allows *few-shot learning* for new characters,  
and *writer identification*.

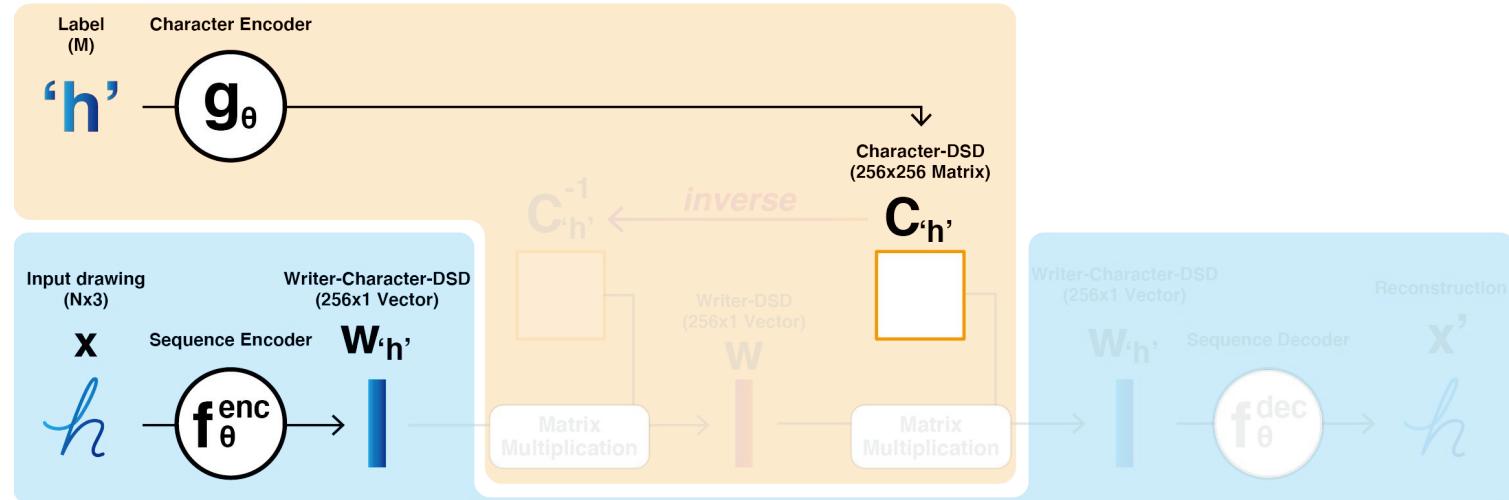


# Decoupled Style Descriptors (DSD)

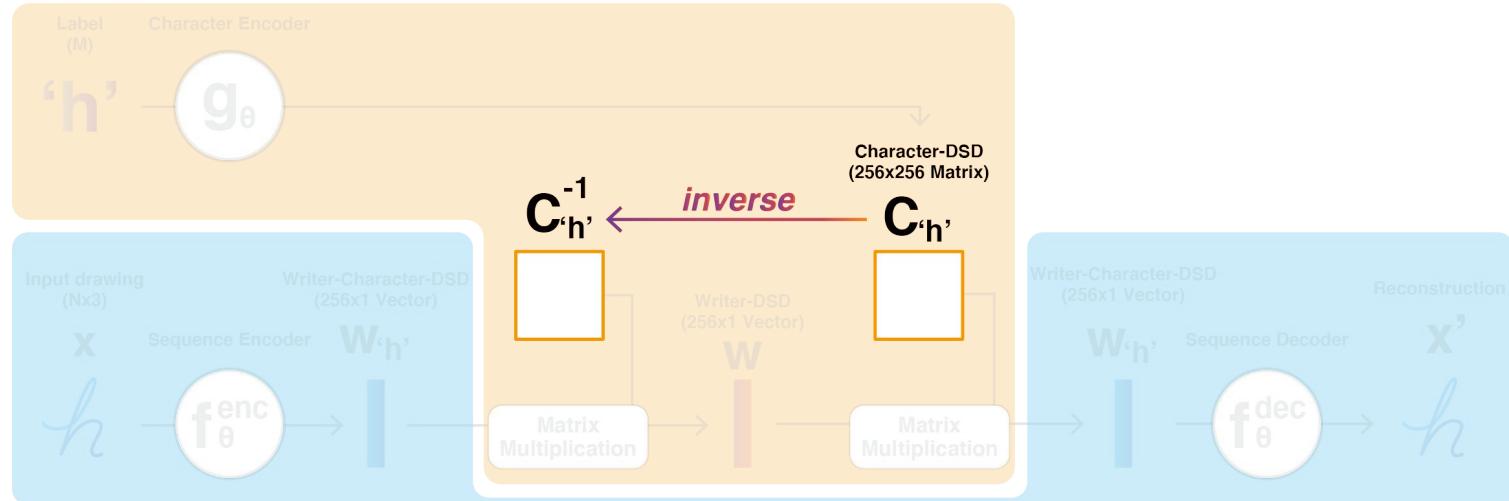


At the core, we have LSTM-based autoencoder, similar to the work by Graves [3].

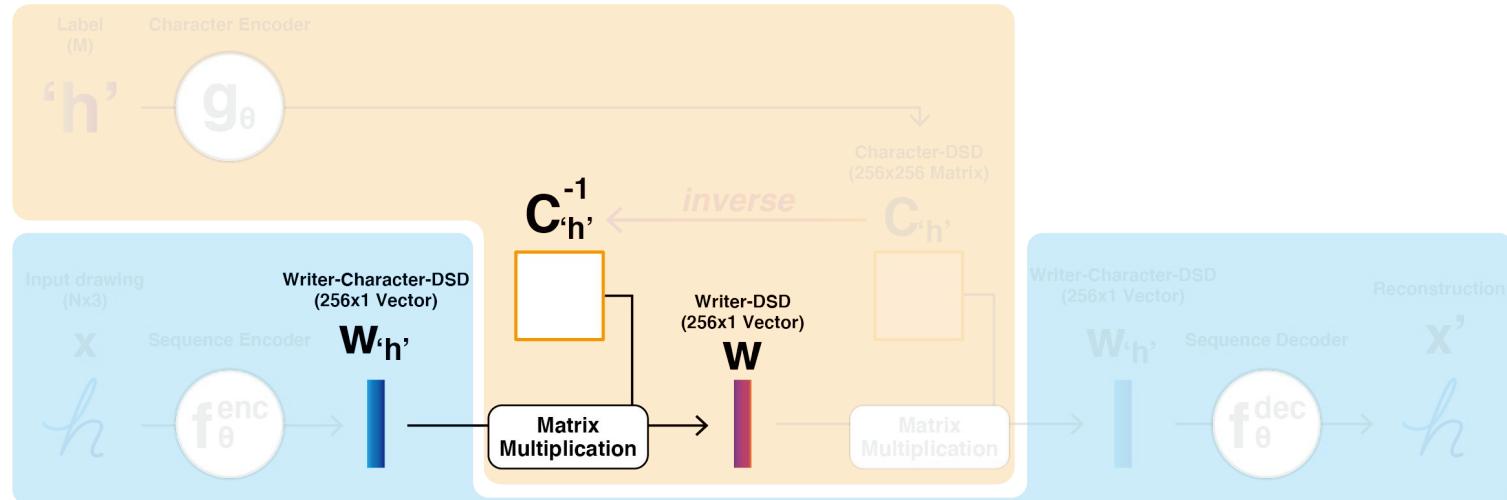
# Decoupled Style Descriptors (DSD)



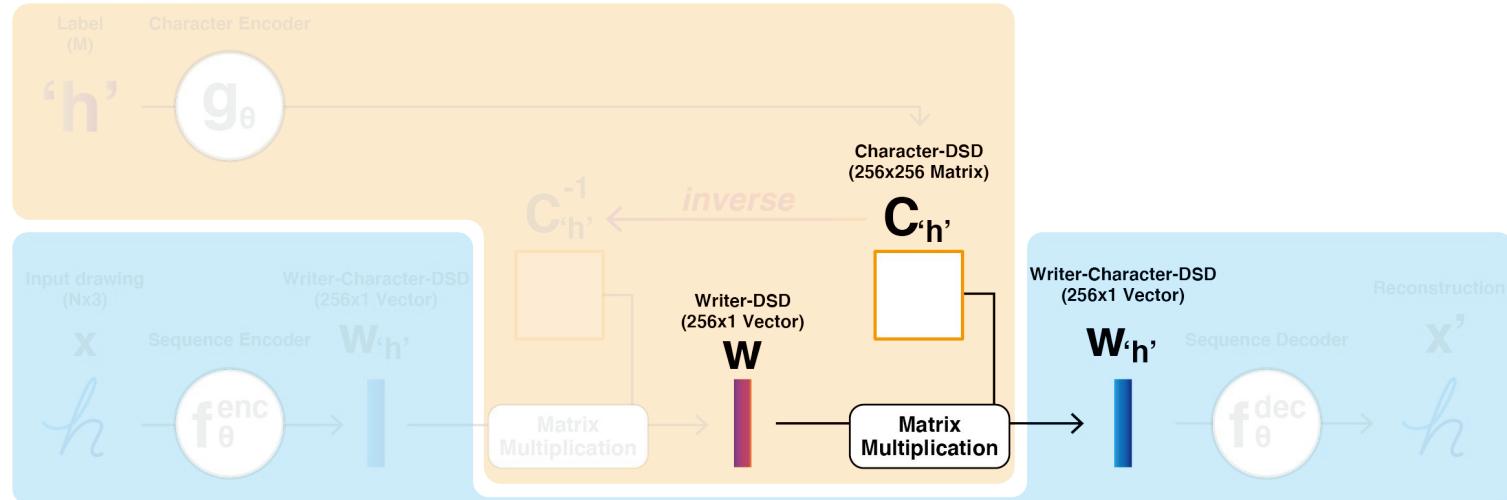
# Decoupled Style Descriptors (DSD)



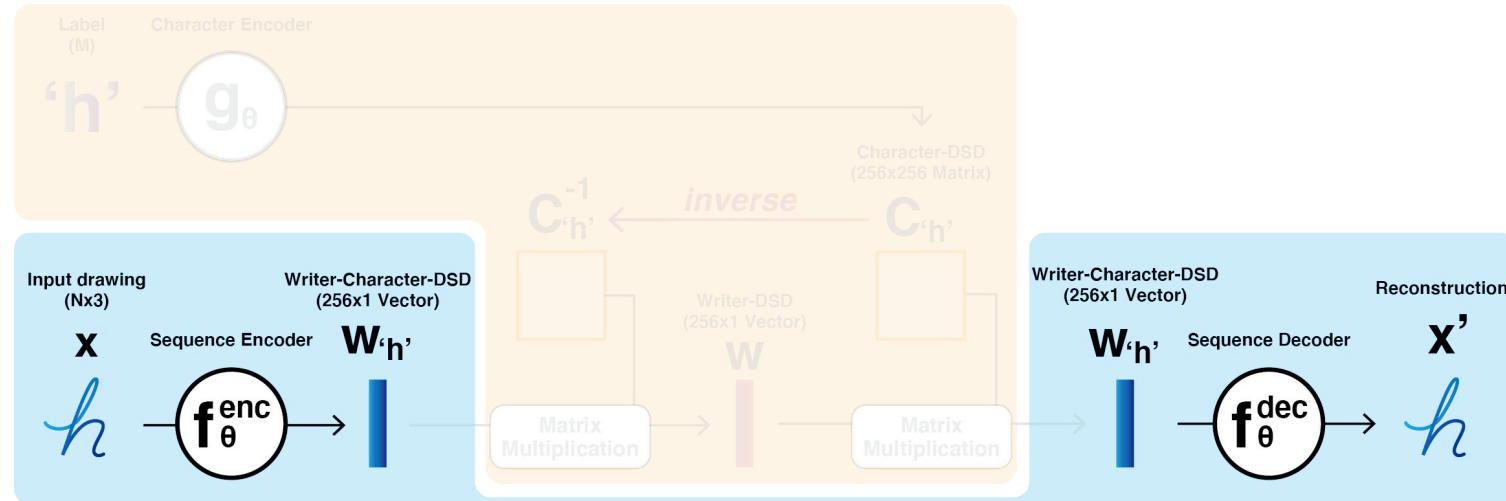
# Decoupled Style Descriptors (DSD)



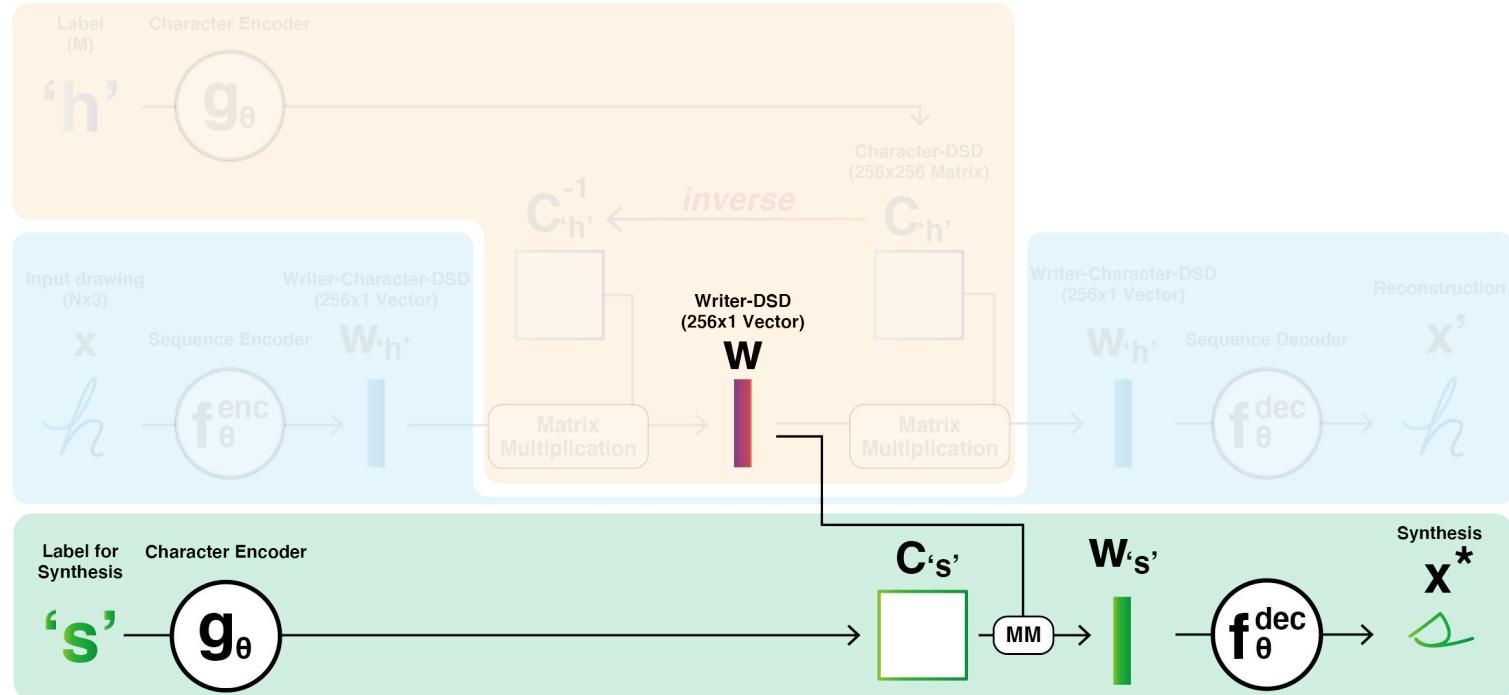
# Decoupled Style Descriptors (DSD)



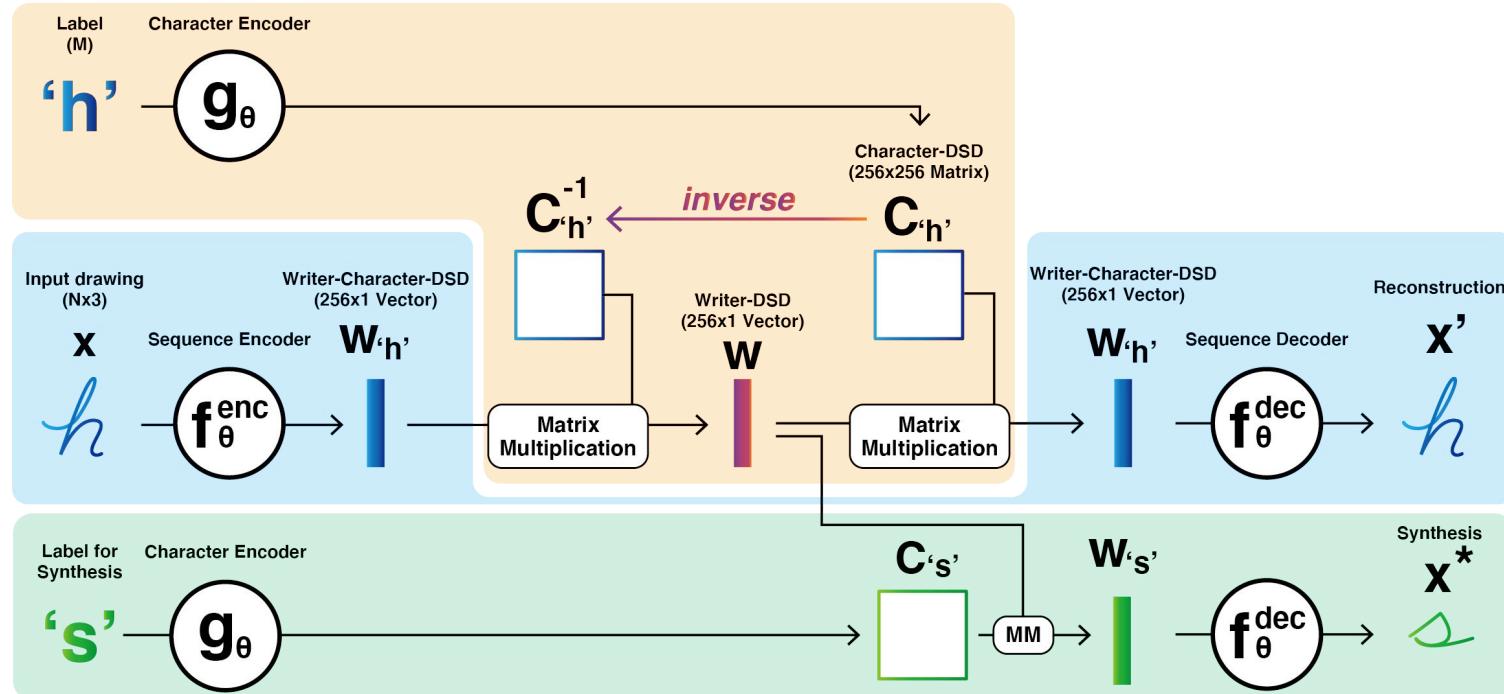
# Decoupled Style Descriptors (DSD)



# Decoupled Style Descriptors (DSD)



# Decoupled Style Descriptors (DSD)



# Subsequences rather than single characters

Writing is complex:

- Cursive
- Character pairs (ligatures)
- Delayed strokes (f, t, i, j)

# Subsequences rather than single characters

Writing is complex:

- Cursive
- Character pairs (ligatures)
- Delayed strokes (f, t, i, j)

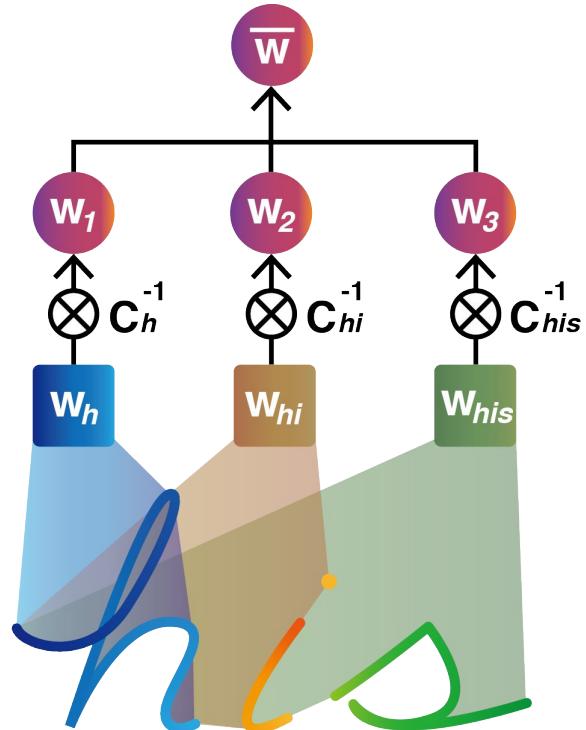
Our approach actually represents latent space of *all subsequences of characters*.

Word ‘his’ has representations     $C_{'h'}$ ,  $C_{'hi'}$ , and  $C_{'his'}$ .  
and     $w_{'h'}$ ,  $w_{'hi'}$ , and  $w_{'his'}$ .

# Recovering Writer-DSD $\mathbf{w}$ from handwriting samples

Take mean  $\mathbf{w}$  over subsequences:

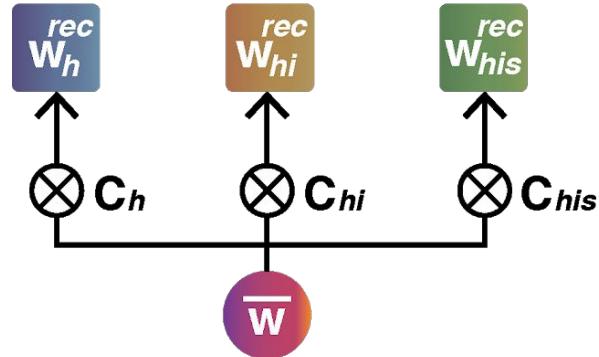
$$\overline{\mathbf{w}} = \frac{1}{M} \sum_{t=1}^M \mathbf{C}_{c_t}^{-1} \mathbf{w}_{c_t}$$



# Generating handwriting using a *global* Writer-DSD $\mathbf{w}$

Given a target word ‘his’:

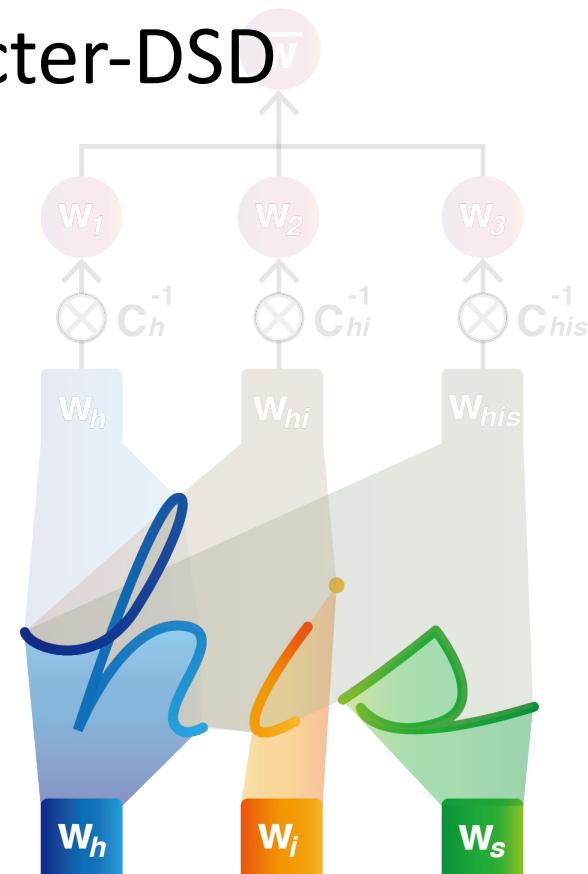
- Predict  $C_{\text{'h'}}$ ,  $C_{\text{'hi'}}$ , and  $C_{\text{'his'}}$
- Multiply by  $\mathbf{w}$  to create  $w_{\text{'h'}}$ ,  $w_{\text{'hi'}}$ , and  $w_{\text{'his'}}$
- Decode ( $w_{\text{'h'}}$ ,  $w_{\text{'hi'}}$ ,  $w_{\text{'his'}}$ ) into stroke sequence



# Single-character Writer-Character-DSD

There are relatively few single-character  $w_{\cdot h}$ .

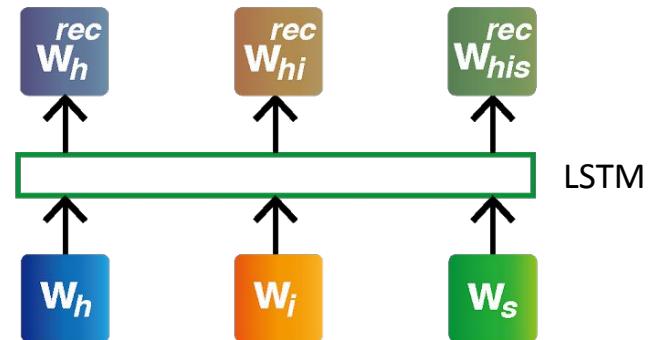
If we extract them from a writing sample,  
we can save them in a **database** and  
**sample** them during generation.



# Generating handwriting using *sampled* Writer-Character-DSD

Retrieve relevant single-characters  $w_{h'}$ ,  $w_{i'}$ ,  $w_{s'}$ .

Restore temporal dependencies via LSTM.



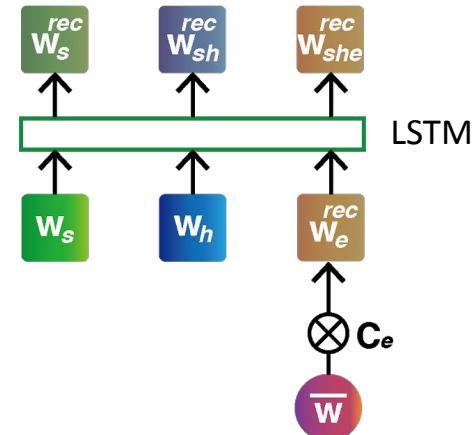
*Cannot cope with any missing characters in reference handwriting samples.*

# Combined method with *sampling*

Reference sample: 'his'

Generation target: 'she'

- Compute mean  $\mathbf{w}$  from all substrings
- Predict  $\mathbf{w}'_{e'}$  with  $\mathbf{C}'_{e'}$  and the mean  $\mathbf{w}$
- Extract single-character Writer-Character-DSDs
- *Restore temporal dependencies with LSTM.*

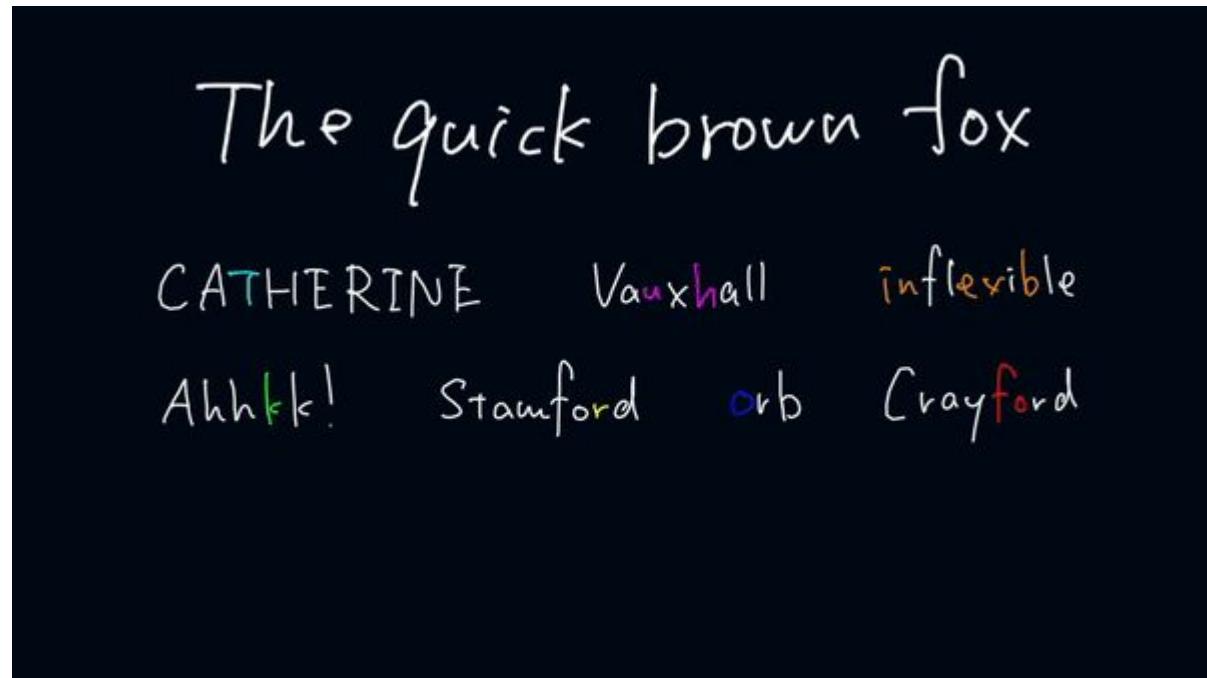


# Generated Results

Colored characters match between provided writing samples and desired output.

These are generated from retrieved  $w_h'$

Missing characters are generated from global  $w$



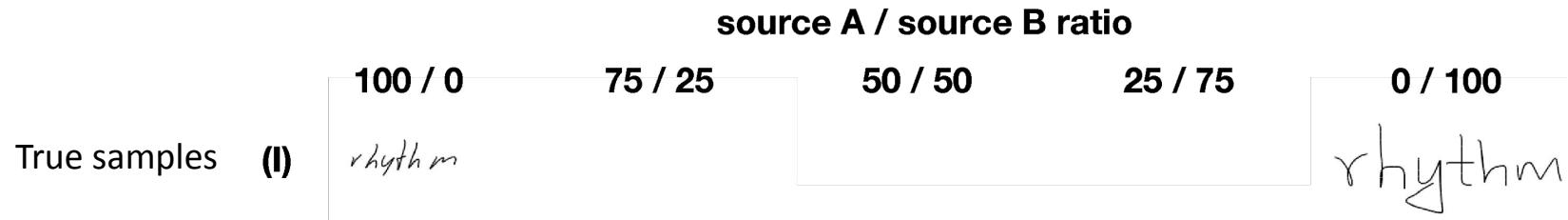
# Generated Results

<b>Target Image</b>	particularly	1 to see her.	the Connecticut
 <b>Ours w/ global DSD</b>	particularly	d to see her	the Connecticut
 <b>Ours w/ sampling</b>	particularly	d to see her	the Connecticut
<b>Target Image</b>	gether with the	young offende	earth, enrit
 <b>Ours w/ global DSD</b>	gether with the	young offende	earth mort
 <b>Ours w/ sampling</b>	gether with the	young offende	earth mort

# Generated Results - Comparison

<b>Target Image</b>	mentioned the f together with the ft of mainly
 <b>DeepWriting [1]</b>	mentioned the f together with the ft of mainly
 <b>Ours w/ global DSD</b>	mentioned the f together with the ft of mainly
 <b>Ours w/ sampling</b>	mentioned the f together with the ft of mainly

# Interpolation



# Interpolation

		source A / source B ratio				
		100 / 0	75 / 25	50 / 50	25 / 75	0 / 100
True samples	(I)	rhythm				rhythm
	(II)	rhythm	rhythm	rhythm	rhythm	rhythm

# Interpolation

		source A / source B ratio				
		100 / 0	75 / 25	50 / 50	25 / 75	0 / 100
True samples (I)		rhythm				rhythm
Writer-DSD level (II)		rhythm	rhythm	rhythm	rhythm	rhythm
Writer-Character-DSD level (III)		rhythm	rhythm	rhythm	rhythm	rhythm

# Interpolation

		source A / source B ratio				
		100 / 0	75 / 25	50 / 50	25 / 75	0 / 100
True samples	(I)	rhythm				rhythm
	(II)	rhythm	rhythm	rhythm	rhythm	rhythm
Writer-DSD level	(III)	rhythm	rhythm	rhythm	rhythm	rhythm
Writer-Character-DSD level	(IV)					
Character-DSD level	(IV)					

# Few-shot learning of New Characters

	Writer A	Writer B
Source for W	qualms polities; EA	qualms polities; EA
C from 1 sample	6 u 0 > 4 < 6 T o h	6 i 0 -> b o 6 T o h
C from 10 samples	0 1 2 3 4 < 6 7 s 9	0 1 2 3 4 > 6 T o 9
C from 100 samples	0 1 2 3 4 s 6 7 8 9	0 1 2 3 4 s 6 7 8 9

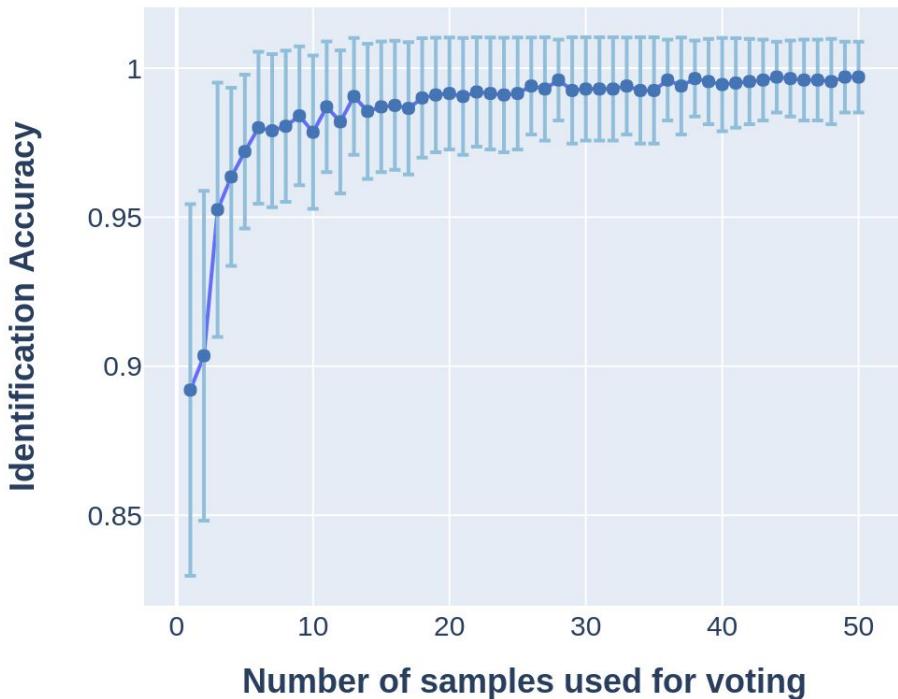
# Writer Identification

In a 20-writer identification task,

our model achieved:

**89.20%  $\pm$  6.23** for **1** word sample

**99.70%  $\pm$  1.18** for **50** word samples



# BRUSH dataset

**BRUSH** dataset contains handwriting of:

- 170 writers, 86 Latin alphabet characters
- 488 common words written by all writers
  - 99.5% coverage of two-character letter space
- + 3668 rarer words written across writers
  - 99.9% coverage in total

Data Collection      Log Out

When you scroll the page, please not to scroll on the drawing boxes but the side space.

1 / 20 pages

qualms politics; [A

CLEAR UNDO

baseline

Islington, Hamlin

CLEAR UNDO

baseline

dwelling. Frankfurt

CLEAR UNDO

The interface shows two rows of handwritten text. The first row contains the text 'qualms politics; [A' with a red 'baseline' label below it. The second row contains the text 'Islington, Hamlin' with a red 'baseline' label below it. Each row has a red 'CLEAR' button on the left and a red 'UNDO' button on the right. The top of the interface has a dark header bar with 'Data Collection' and 'Log Out' buttons. Below the header is a red warning message about not scrolling on the drawing boxes. The page number '1 / 20 pages' is at the top center. The bottom of each row shows a red 'baseline' label.

# BRUSH dataset

**BRUSH** dataset contains handwriting of:

- 170 writers, 86 Latin alphabet characters
- 488 common words written by all writers
  - 99.5% coverage of two-character letter space
- + 3668 rarer words written across writers
  - 99.9% coverage in total

Common word samples & character-level labels  
are not present in IAM dataset [4].

The screenshot shows a user interface for 'Data Collection' with a 'Log Out' button at the top right. A red banner at the top states: 'When you scroll the page, please not to scroll on the drawing boxes but the side space.' Below it, a status bar shows '1 / 20 pages'. The main area displays three handwritten samples with their corresponding labels and clear/undo buttons.

- Sample 1:** The label 'qualms politics; [A' is shown above the handwritten text 'qualms politics; [A'. The text is written in cursive on a grid baseline. Buttons for 'CLEAR' and 'UNDO' are to the right.
- Sample 2:** The label 'Islington, Hamlin' is shown above the handwritten text 'Islington, Hamlin'. The text is written in cursive on a grid baseline. Buttons for 'CLEAR' and 'UNDO' are to the right.
- Sample 3:** The label 'dwelling. Frankfurt' is shown above the handwritten text 'dwelling. Frankfurt'. The text is written in cursive on a grid baseline. Buttons for 'CLEAR' and 'UNDO' are to the right.

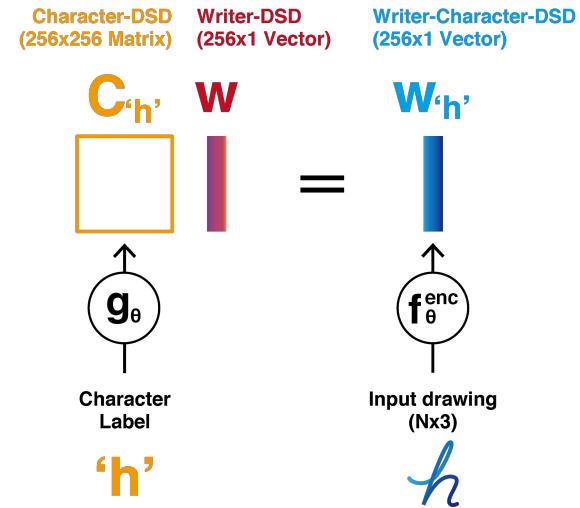
# Generating Handwriting via Decoupled Style Descriptors

Atsunobu Kotani, Stefanie Tellex, James Tompkin



## Contributions:

- DSDs decouple character style from writer style.
- Allow flexible handwriting generation.
- BRUSH: new dataset for online handwriting.



Code & dataset available at  
<http://dsd.cs.brown.edu>

# Generating Handwriting via Decoupled Style Descriptors

Atsunobu Kotani, Stefanie Tellex, James Tompkin

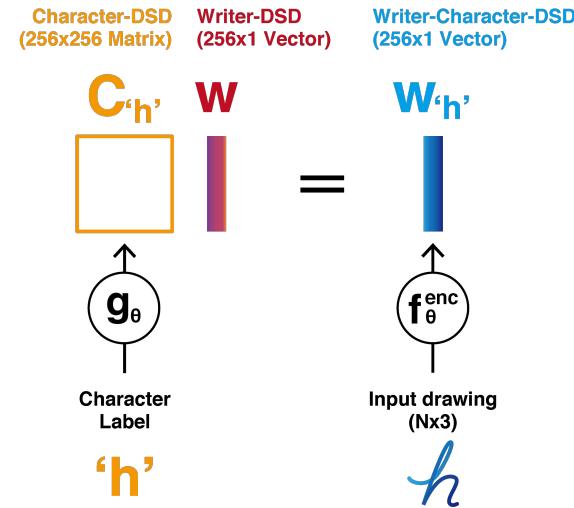


## Contributions:

- DSDs decouple character style from writer style.
- Allow flexible handwriting generation.
- BRUSH: new dataset for online handwriting.

## Potential Extension:

- Explicit decoupling may work better than implicit decoupling / disentanglement.
- DSDs could be applied to other sequential data (e.g. speech, motion capture data)



Code & dataset available at  
<http://dsd.cs.brown.edu>