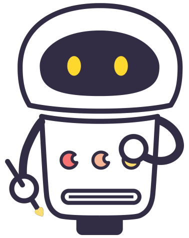
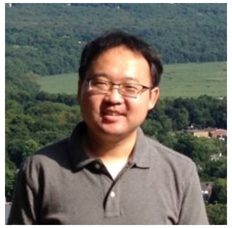
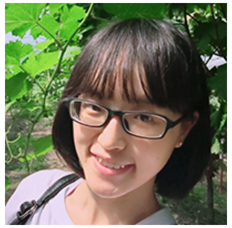


AI-Sketcher: A Deep Generative Model for Generating High Quality Sketches

Nan Cao, **Xin Yan**, Yang Shi, Chaoran Chen

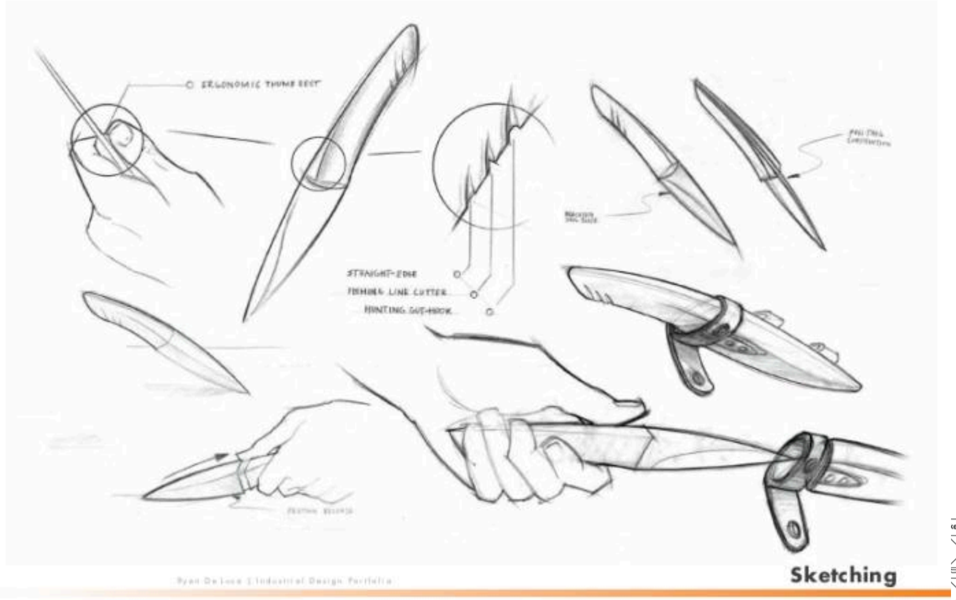
Tongji University

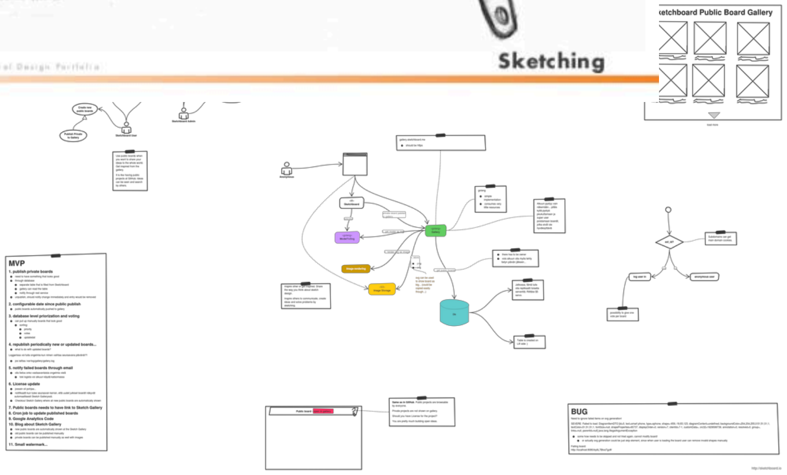
Intelligent Big Data Visualization Lab

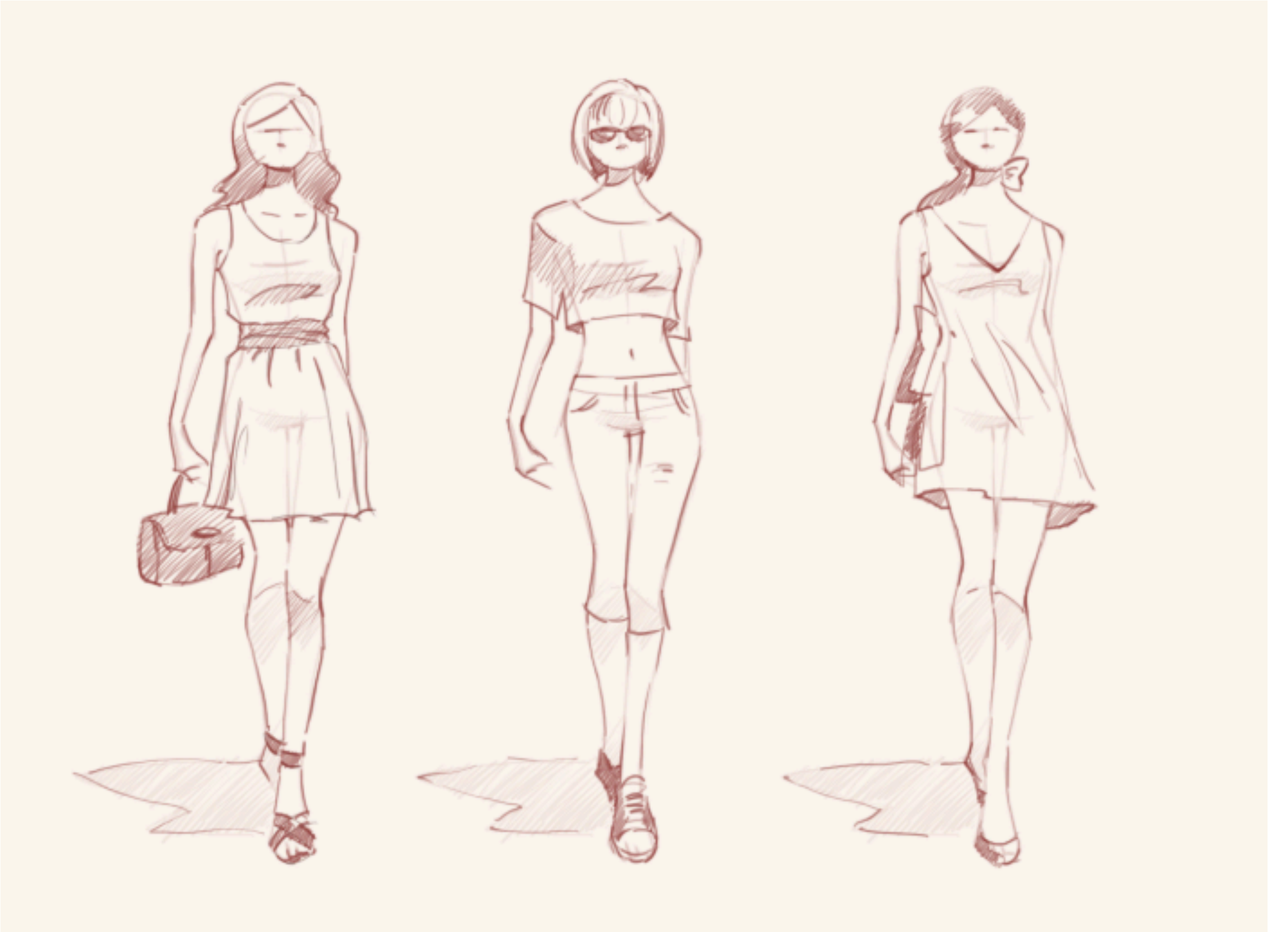
BACKGROUND - Cave Painting

BACKGROUND - Children's Drawings

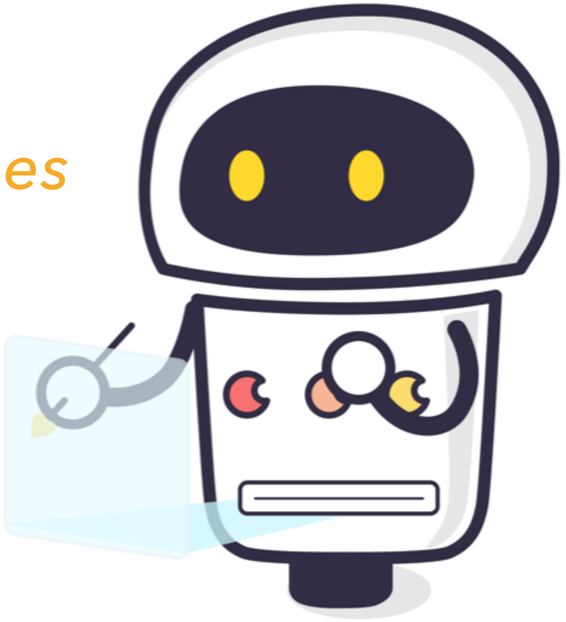
BACKGROUND - Design Drafts





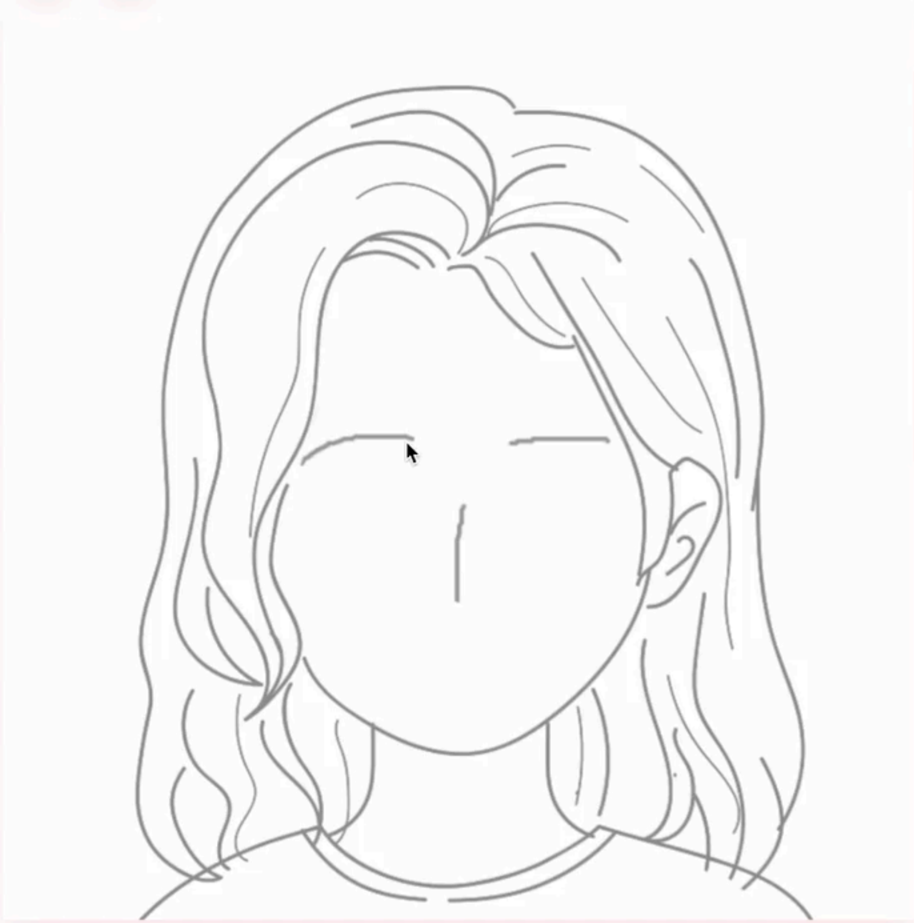


*Can **AI** help designers create high quality sketches and boost their productivity and creativity?*

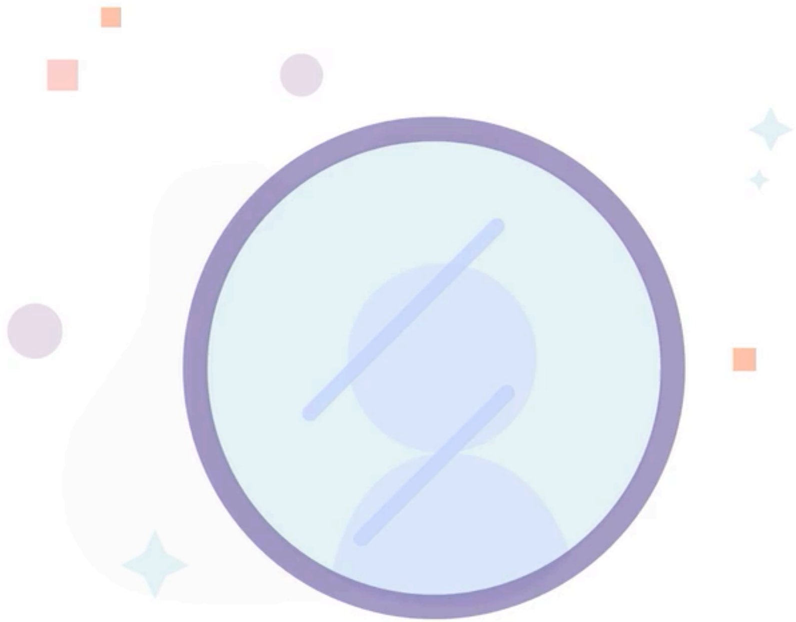


AI-Sketcher

Facial Sketch Autodraw

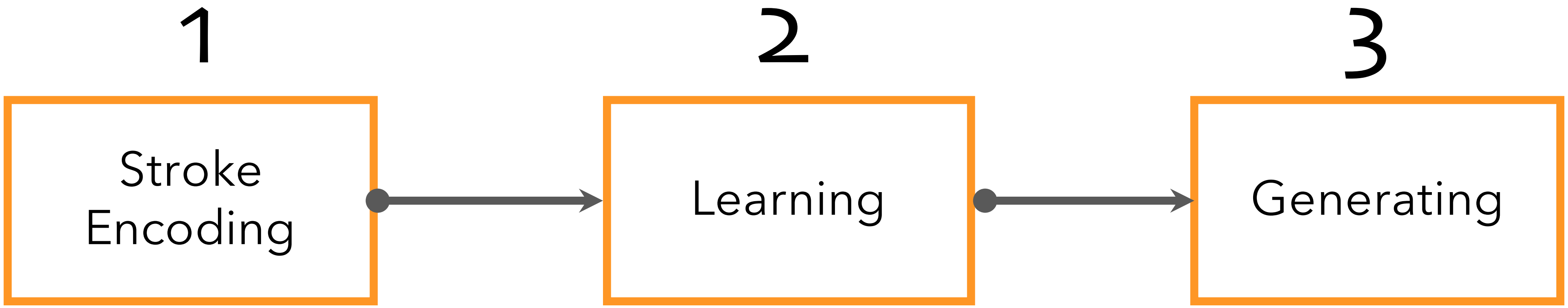


Style: Cartoon Gender: Female



Optimize your drawings here!

Reference Pipeline for AI-Supported Sketching



Stroke Encoding

Stroke Encoding

Learning

Generating

continuous one-hot

$$[\triangle x, \ \triangle y, \ p_1, \ p_2, \ p_3]$$

Start coordinates relative to the end of last stroke

continuous drawing from the last point,

end of drawing a stroke

end of drawing a sketch

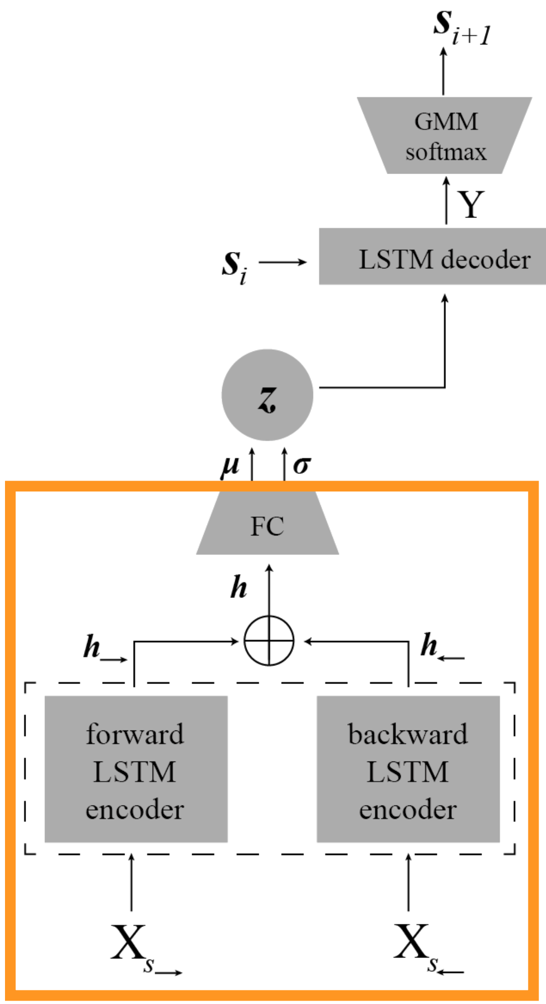
——"Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477, 2017."

Learning (Sketch-RNN)

Stroke Encoding

Learning

Generating



Encoder: Bidirectional RNN (BRNN)

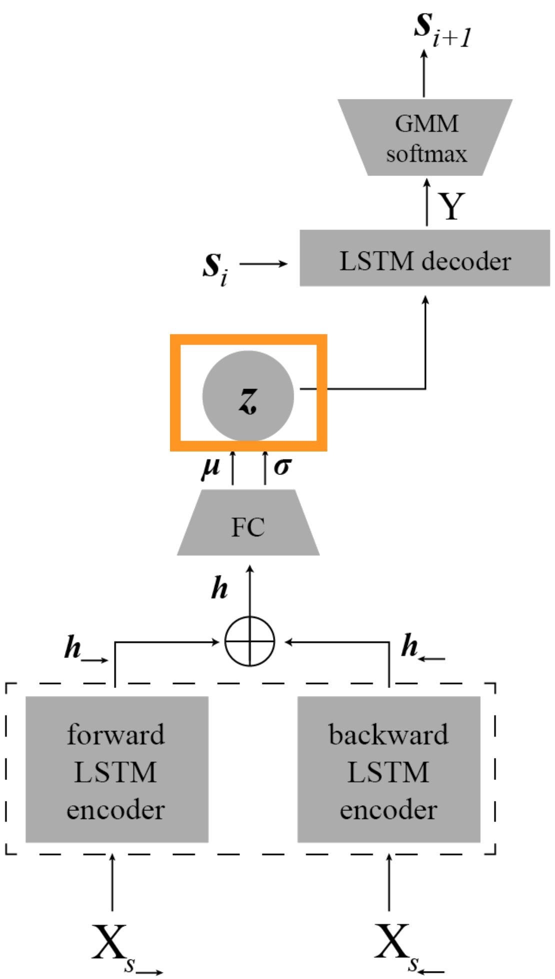
——"Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477, 2017."

Learning (Sketch-RNN)

Stroke Encoding

Learning

Generating



Random Sampling for Stroke Generation

Encoder: Bidirectional RNN (BRNN)

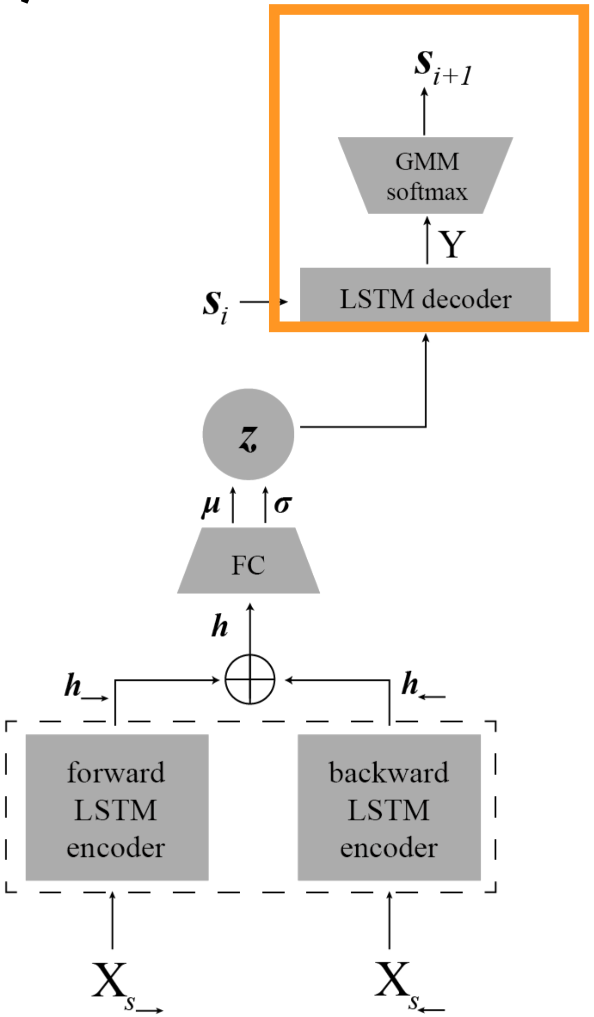
——"Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477, 2017."

Learning (Sketch-RNN)

Stroke Encoding

Learning

Generating



Decoder: Autoregressive RNN

Random Sampling for Stroke Generation

Encoder: Bidirectional RNN (BRNN)

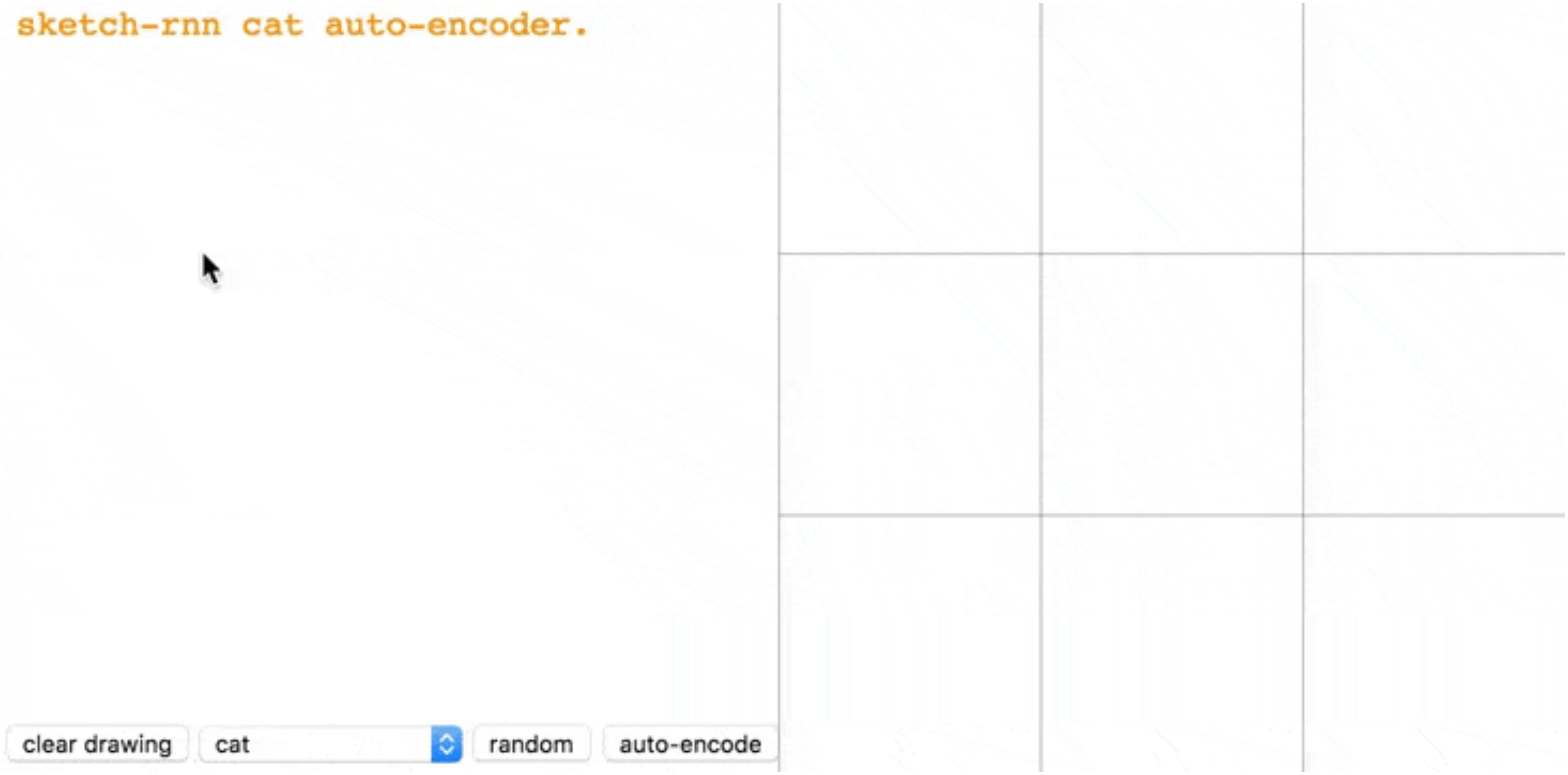
——"Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477, 2017."

Generating (Sketch-RNN)

Stroke Encoding

Learning

Generating

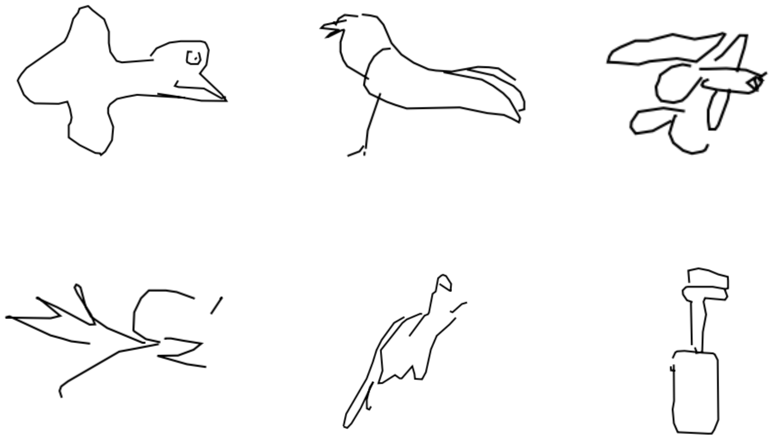


——"Ha D, Eck D. A neural representation of sketch drawings[J]. arXiv preprint arXiv:1704.03477, 2017."

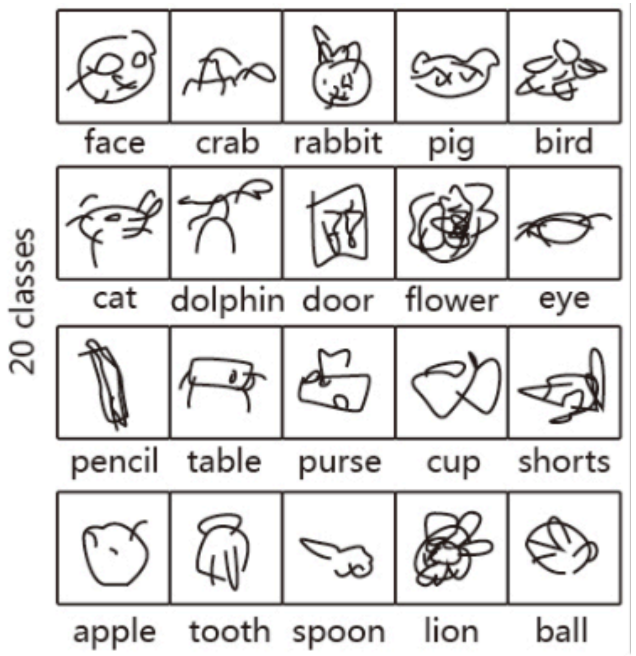
LIMITATIONS OF SKETCH-RNN

Low quality

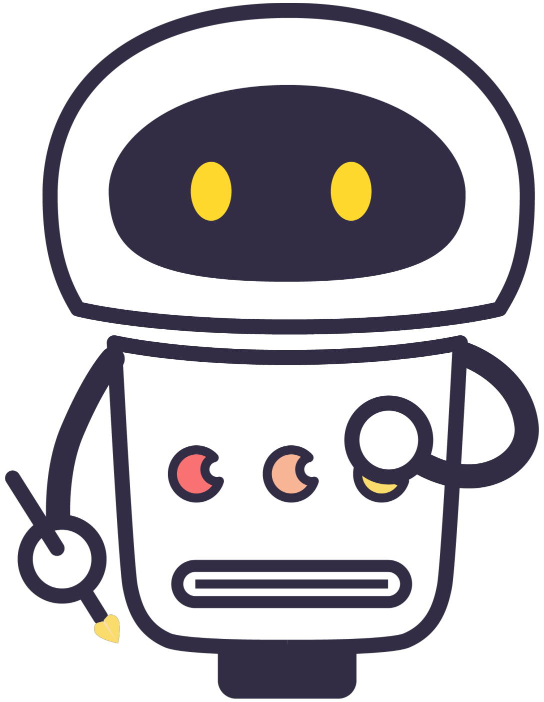
1. generating sketches in one category (bird).



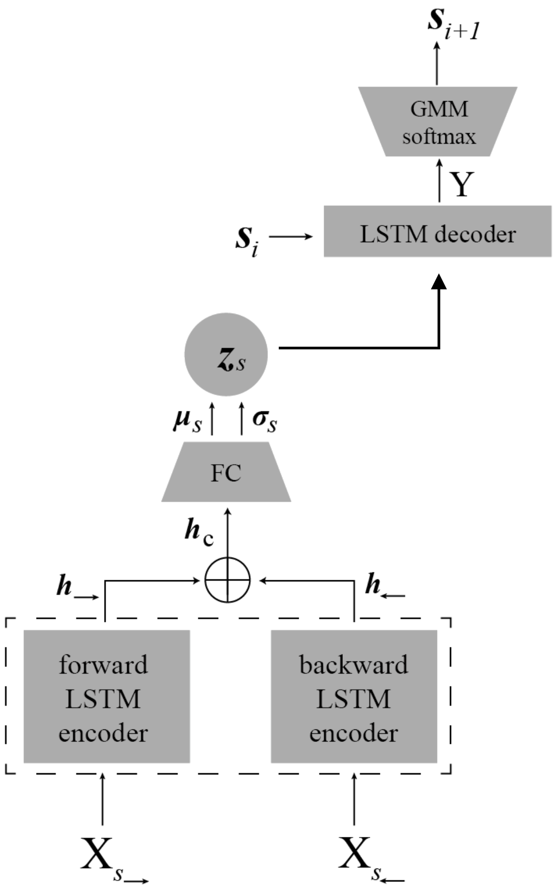
2. dealing with multi-class situations



AI-SKETCHER

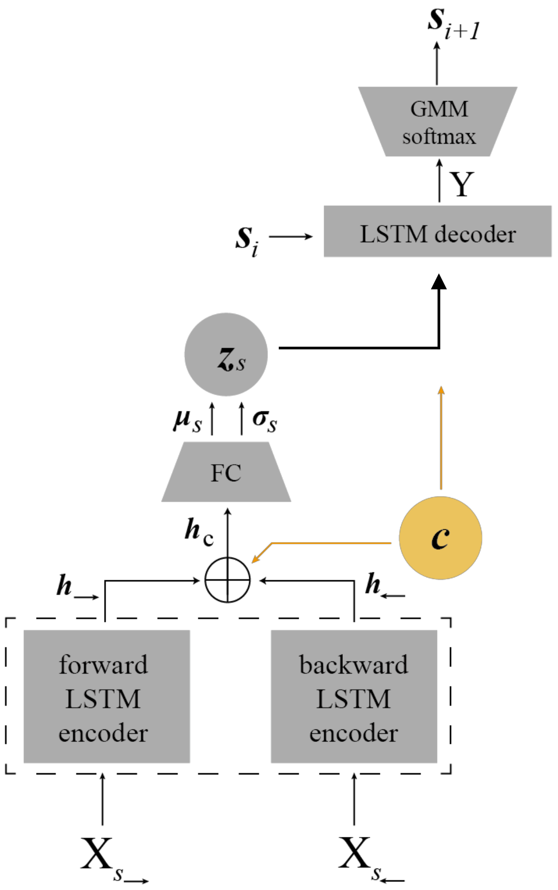


AI-SKETCHER



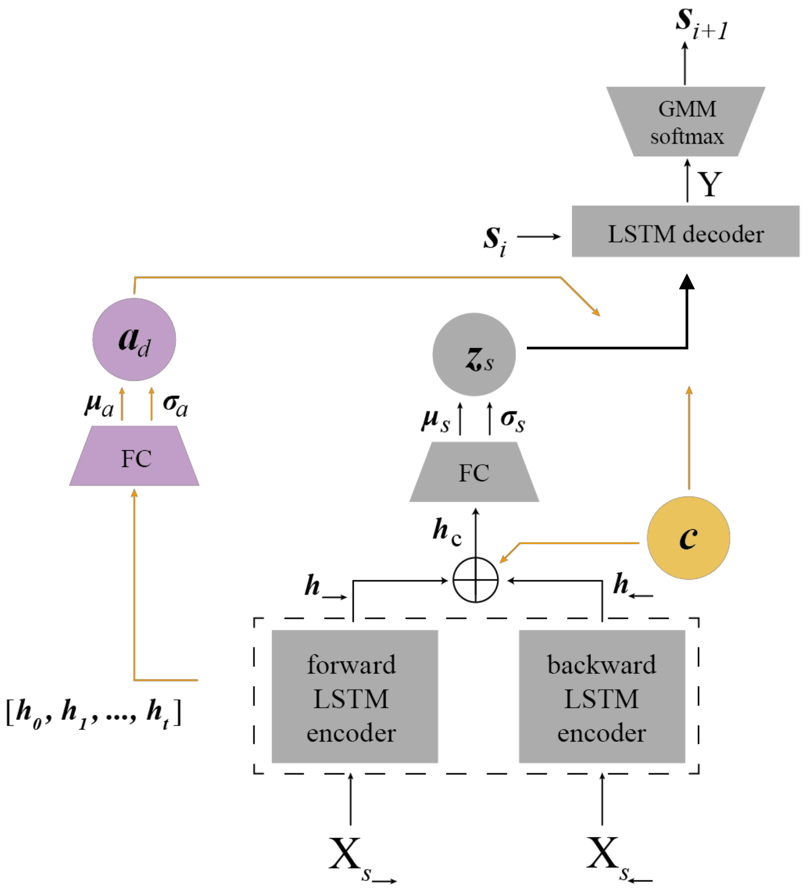
s_{i+1}
GMM softmax
Y
s_i
LSTM decoder
z_s
μ_s
σ_s
FC
h_c
$h_{\rightarrow}$
$h_{\leftarrow}$
forward LSTM encoder
backward LSTM encoder
$X_{s_{\rightarrow}}$
$X_{s_{\leftarrow}}$

AI-SKETCHER



- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

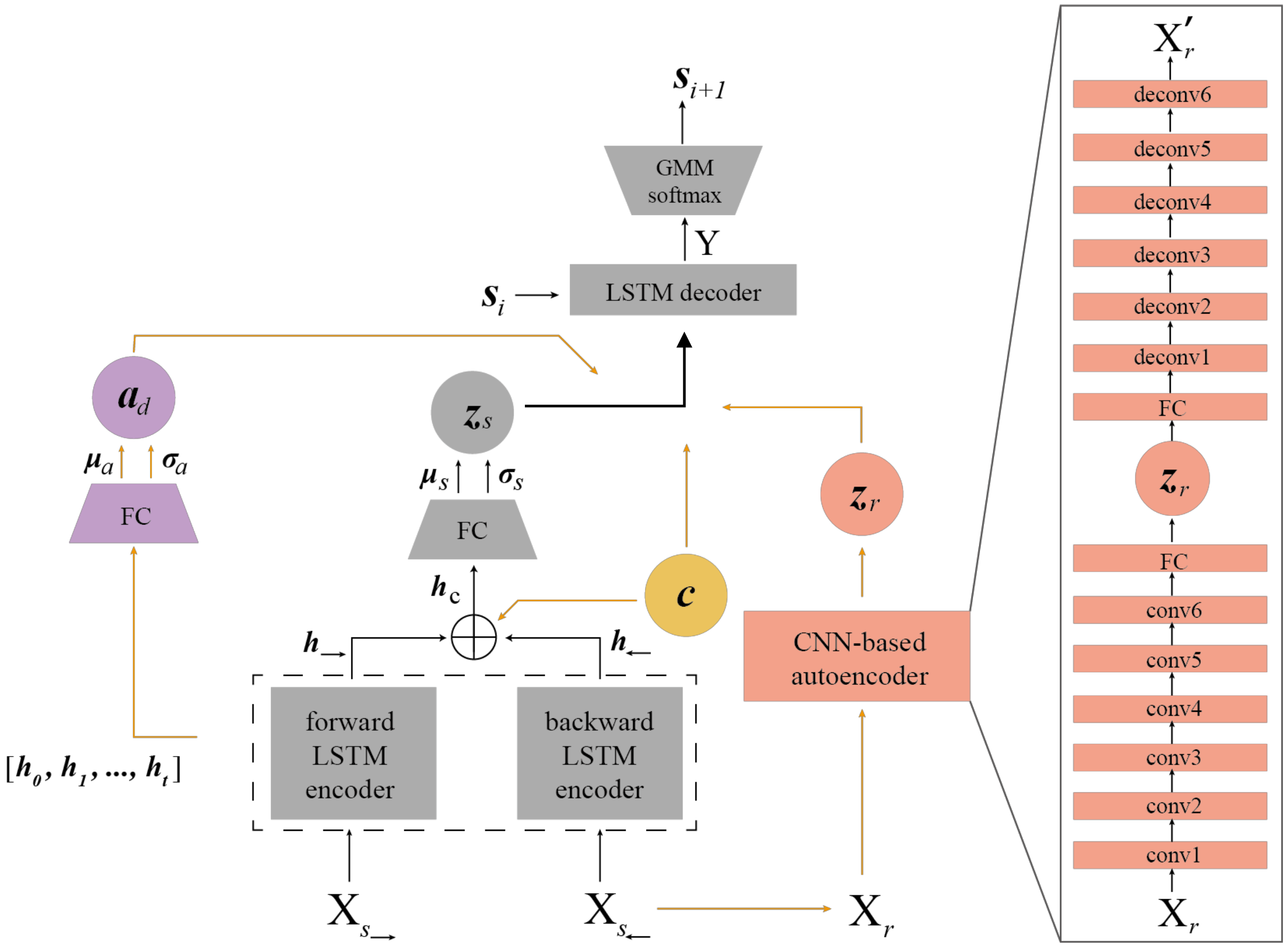
AI-SKETCHER



- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.

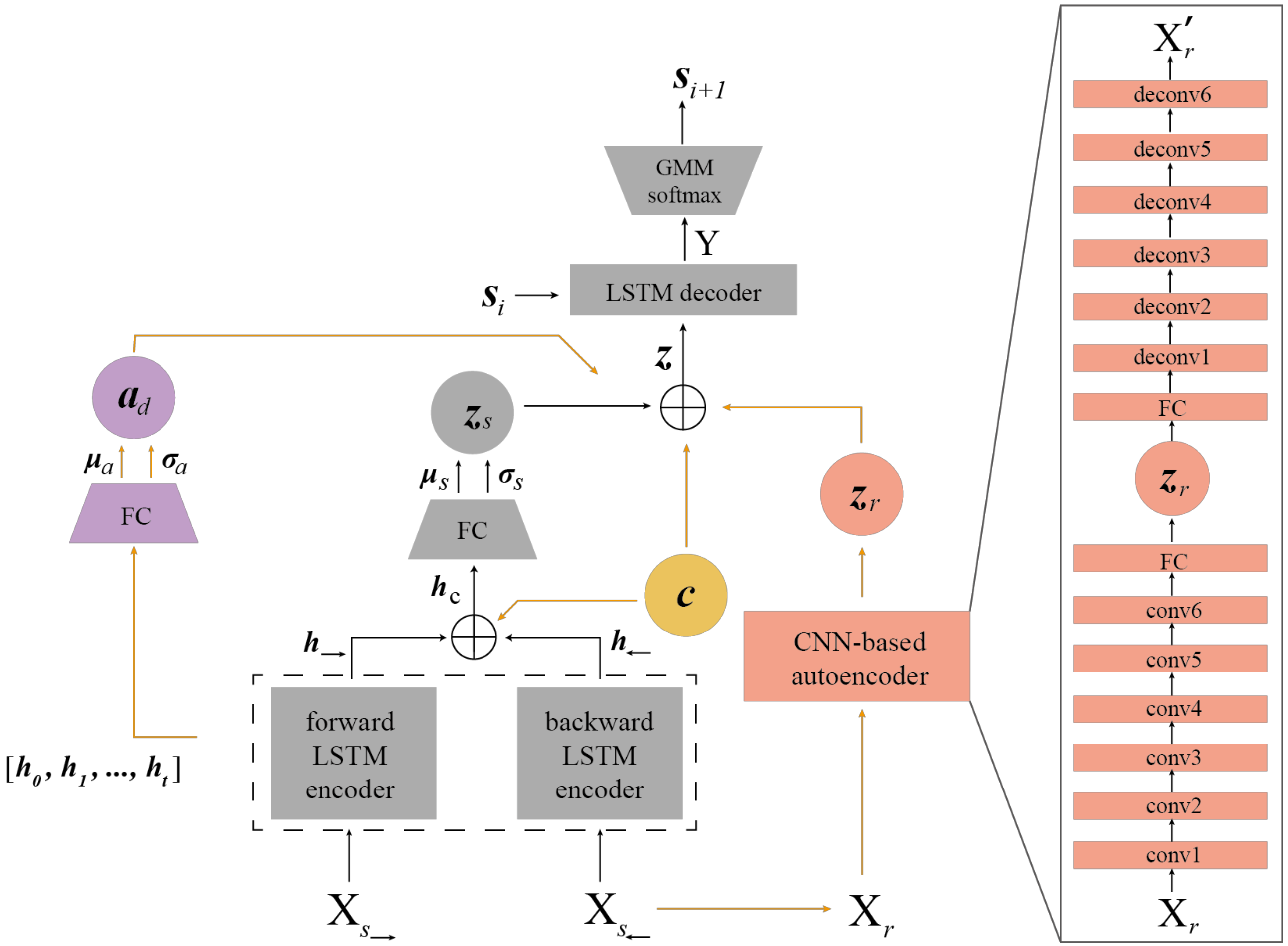
- **An influence layer is introduced to estimate how the previous strokes will influence on the next stroke.**

AI-SKETCHER



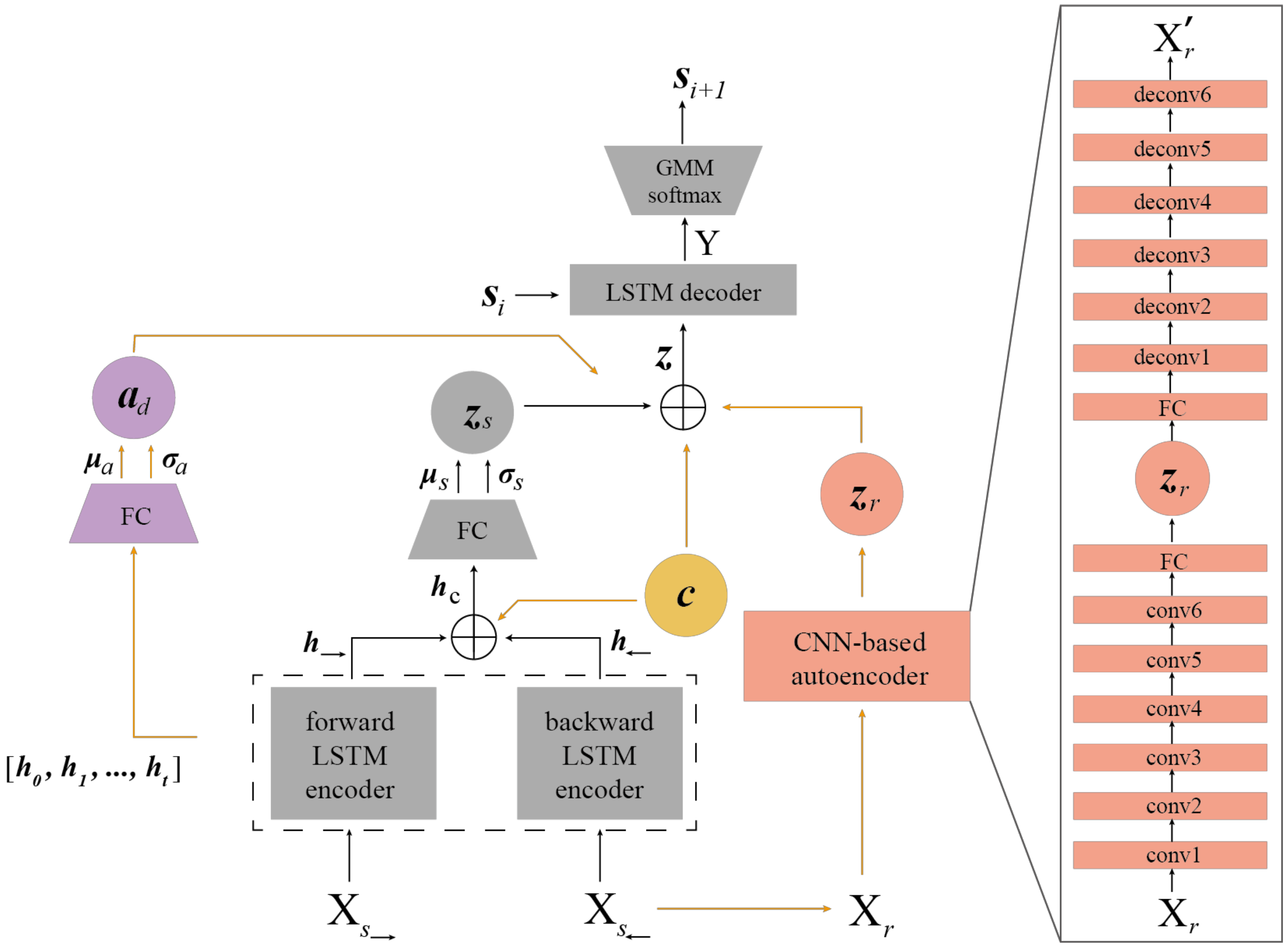
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

AI-SKETCHER



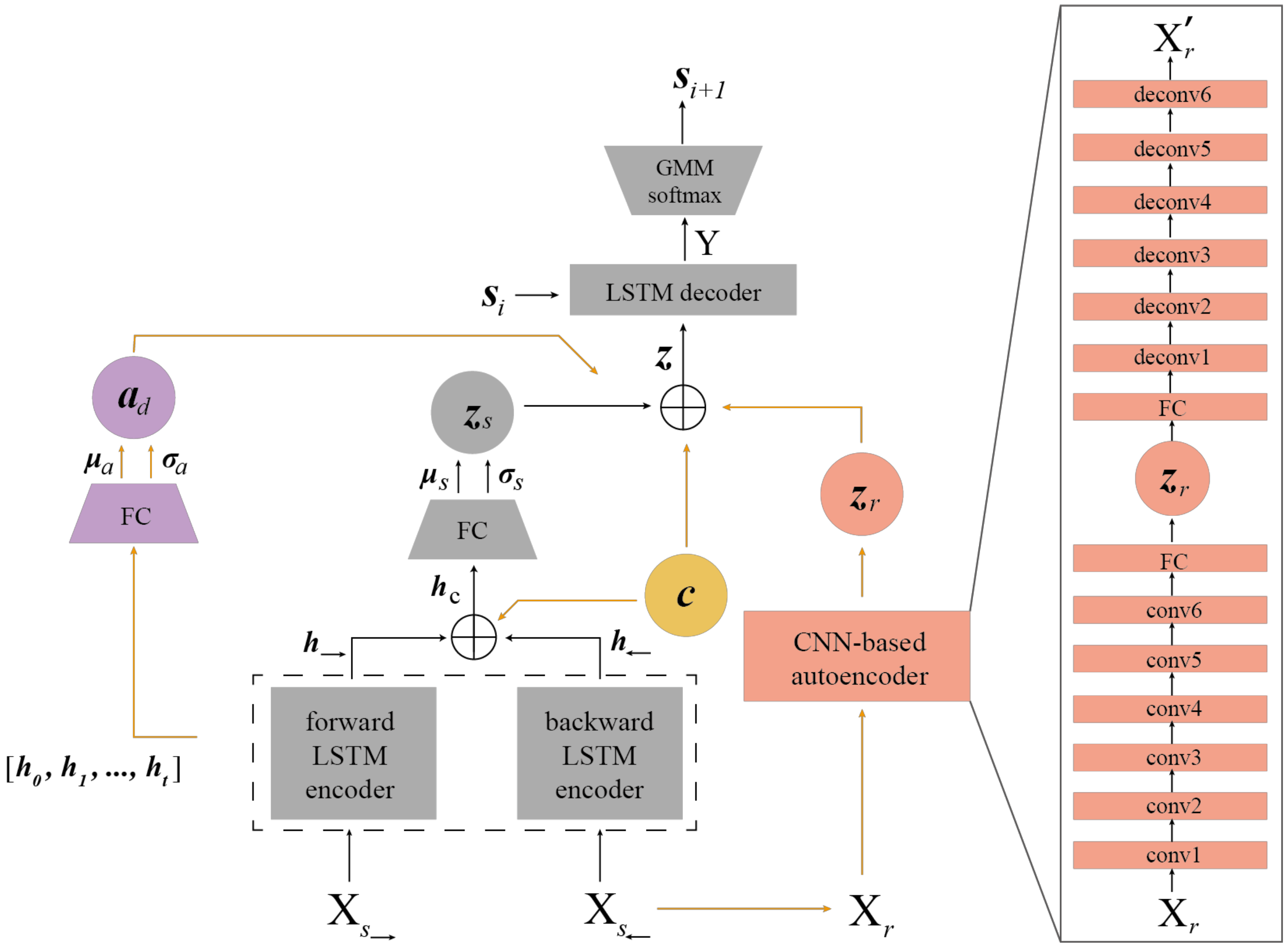
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function is modified.**

AI-SKETCHER



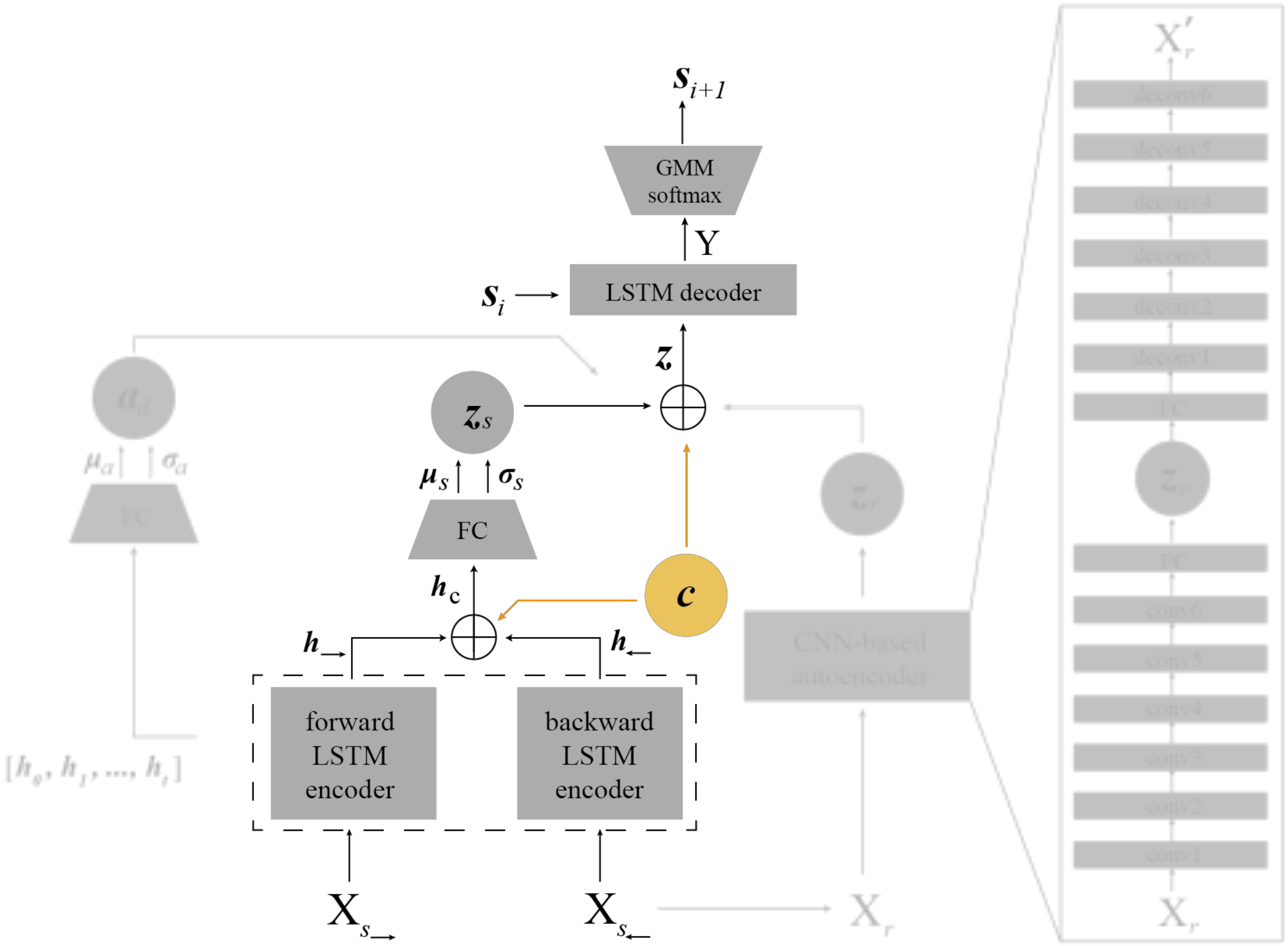
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function** is modified.

AI-SKETCHER



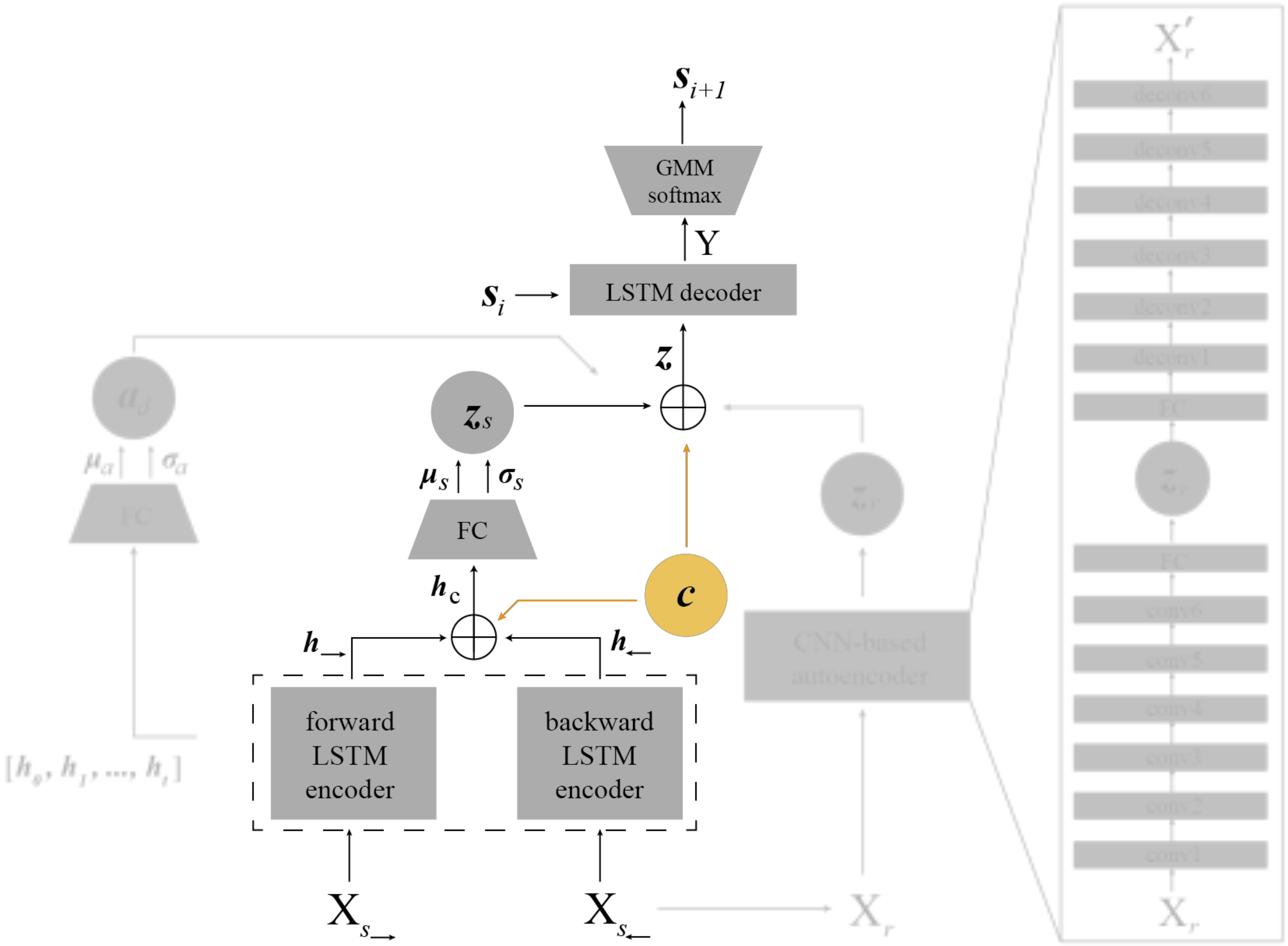
- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function** is modified.

AI-SKETCHER - Conditional Sequence-to-Sequence VAE



- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

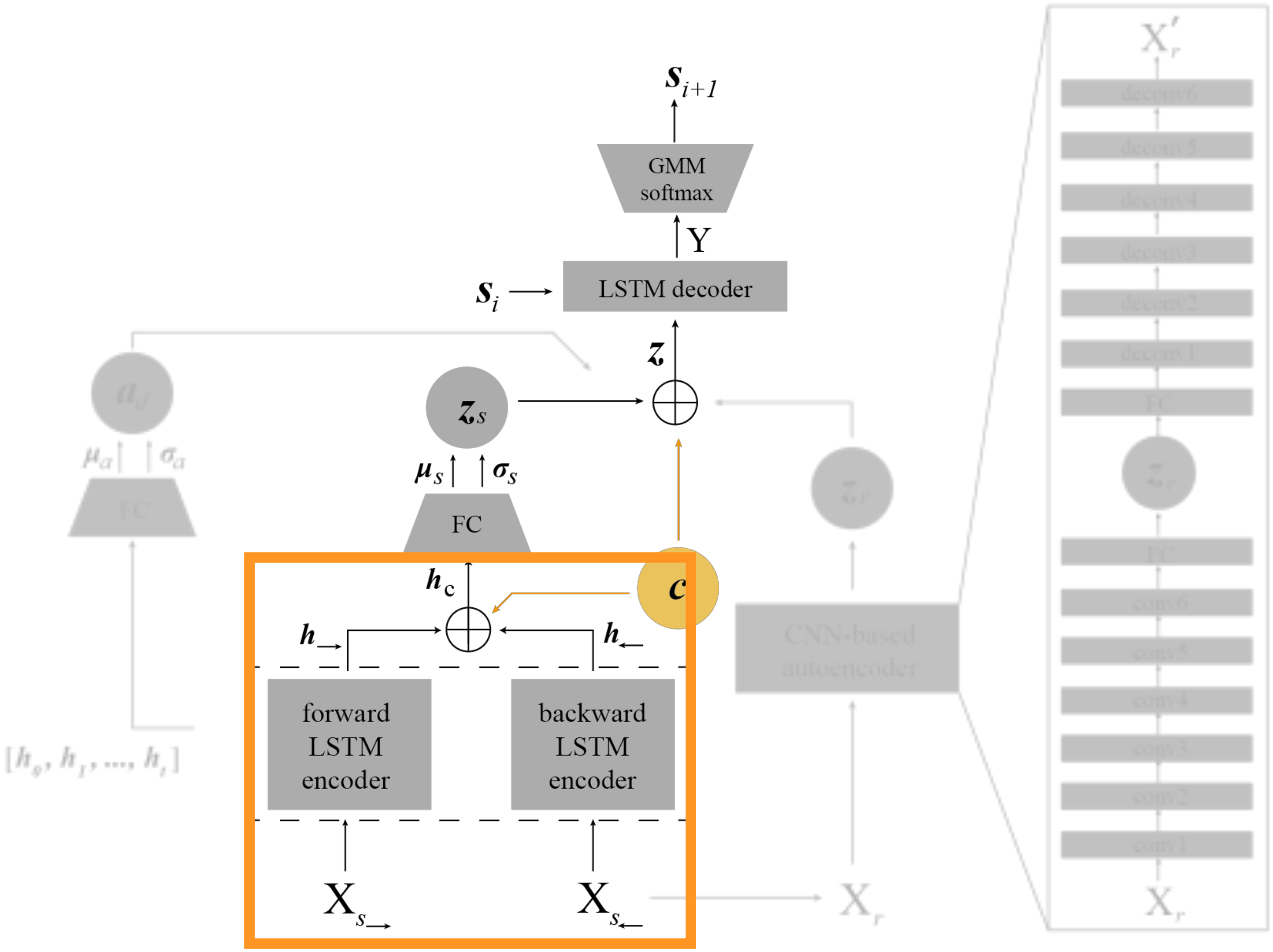
AI-SKETCHER - Conditional Sequence-to-Sequence VAE



- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

Encoder: Bidirectional RNN

AI-SKETCHER - Conditional Sequence-to-Sequence VAE

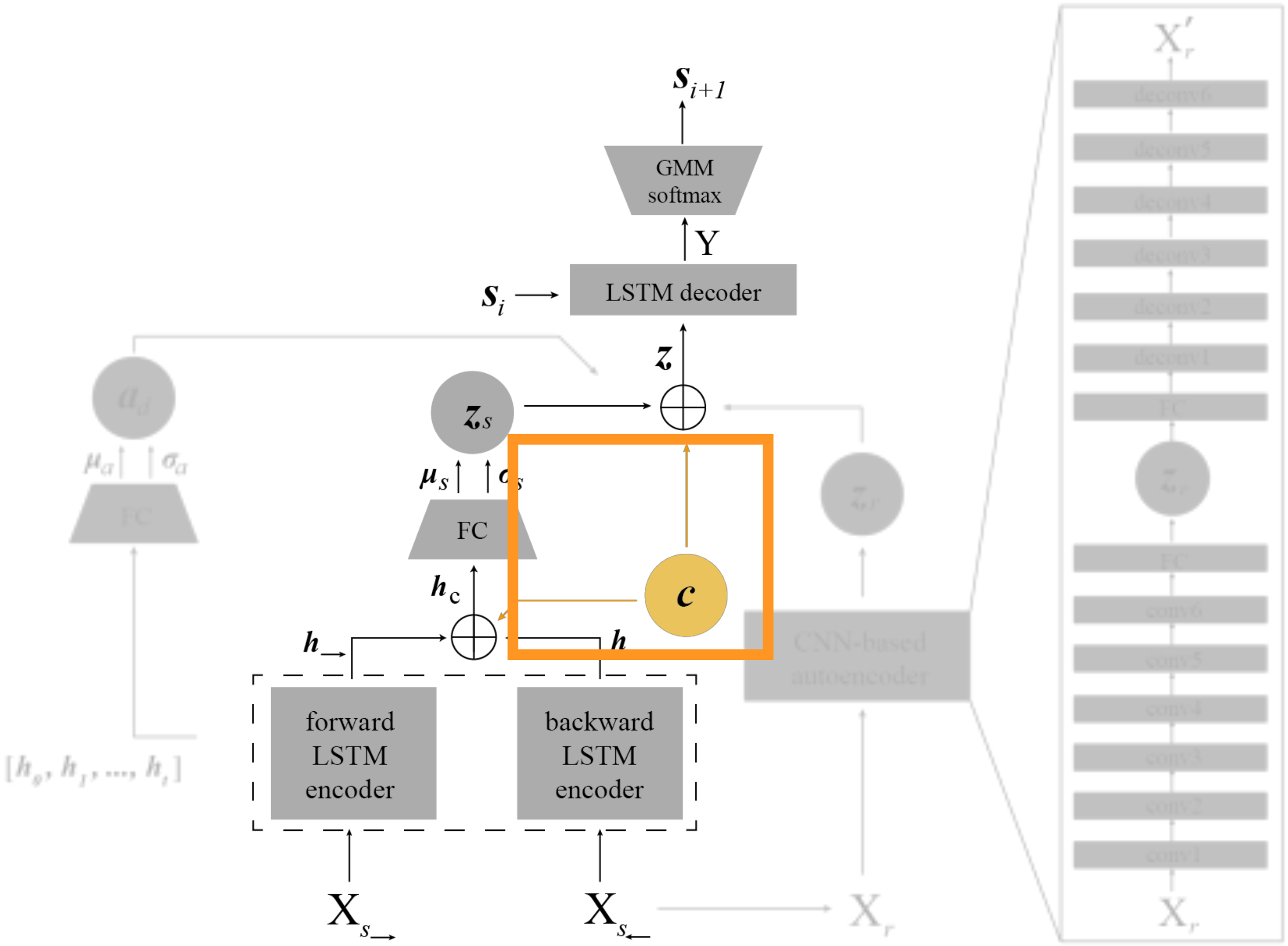


- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

X_s: the sequences of strokes.

$$\boldsymbol{h}^{enc} = encode(X_s)$$

AI-SKETCHER - Conditional Sequence-to-Sequence VAE

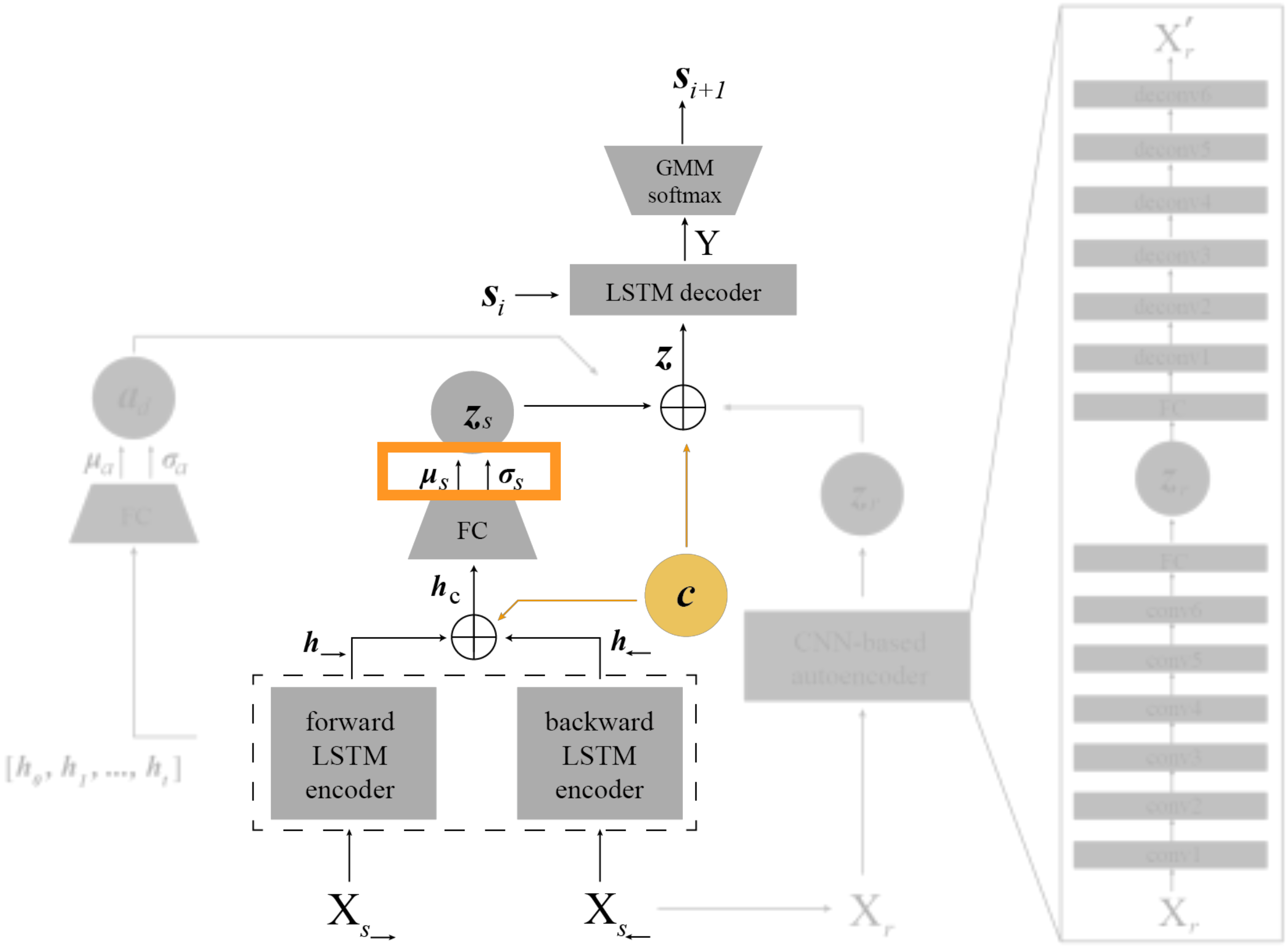


- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

$$\boldsymbol{h}_c = [\boldsymbol{h}^{enc}; \boldsymbol{c}]$$

$\boldsymbol{c}$ is a k-dimensional one-hot conditional vector with k indicates the number of conditions.

AI-SKETCHER - Conditional Sequence-to-Sequence VAE



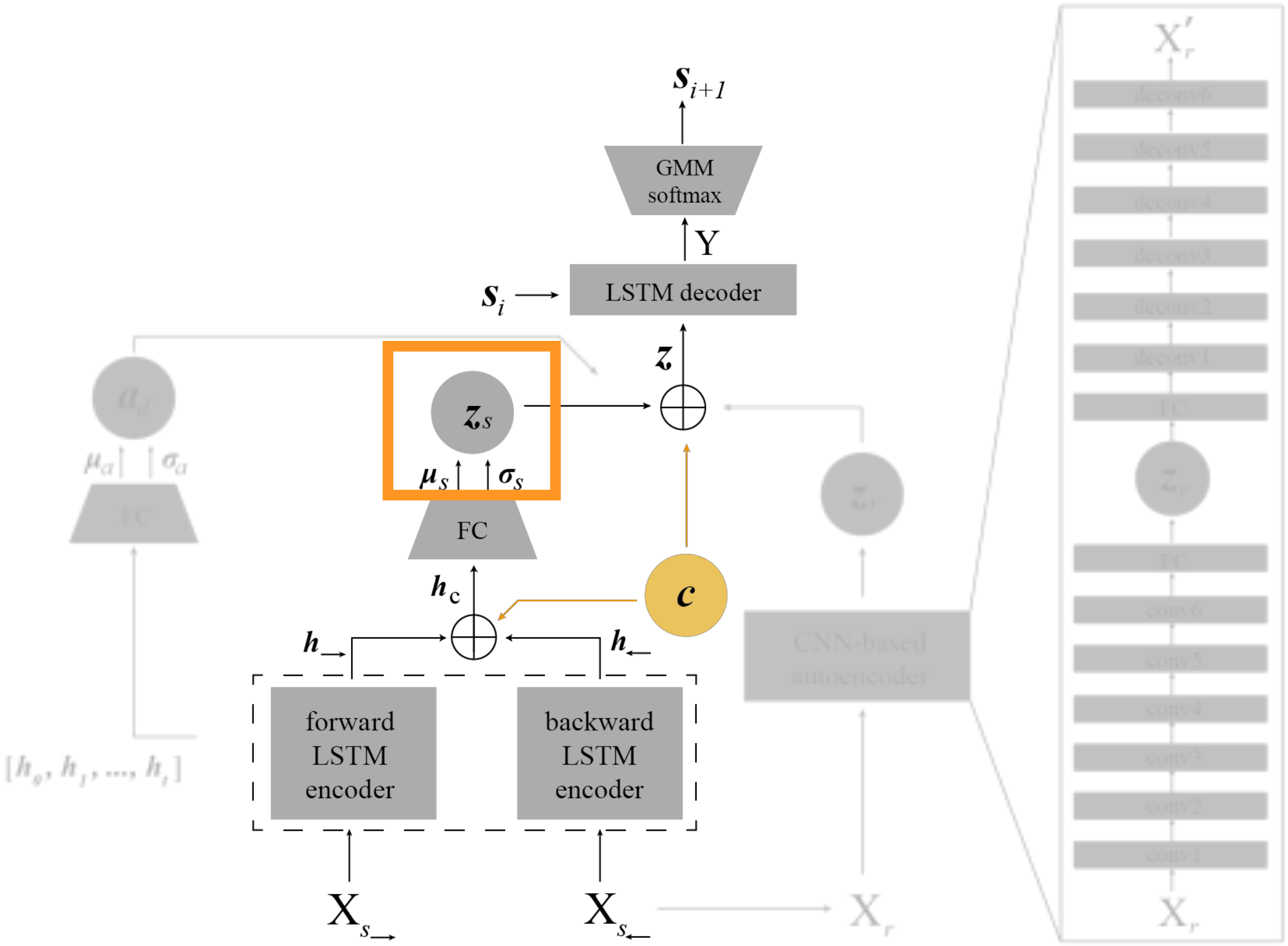
- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

$\boldsymbol{h}_c$ is further transformed into two vectors to capture the distributions of the training strokes:

$$\boldsymbol{\mu}_s = W_\mu \boldsymbol{h}_c + \boldsymbol{b}_\mu$$

$$\boldsymbol{\sigma}_s = exp(\frac{W_\sigma \boldsymbol{h}_c + \boldsymbol{b}_\sigma}{2})$$

AI-SKETCHER - Conditional Sequence-to-Sequence VAE

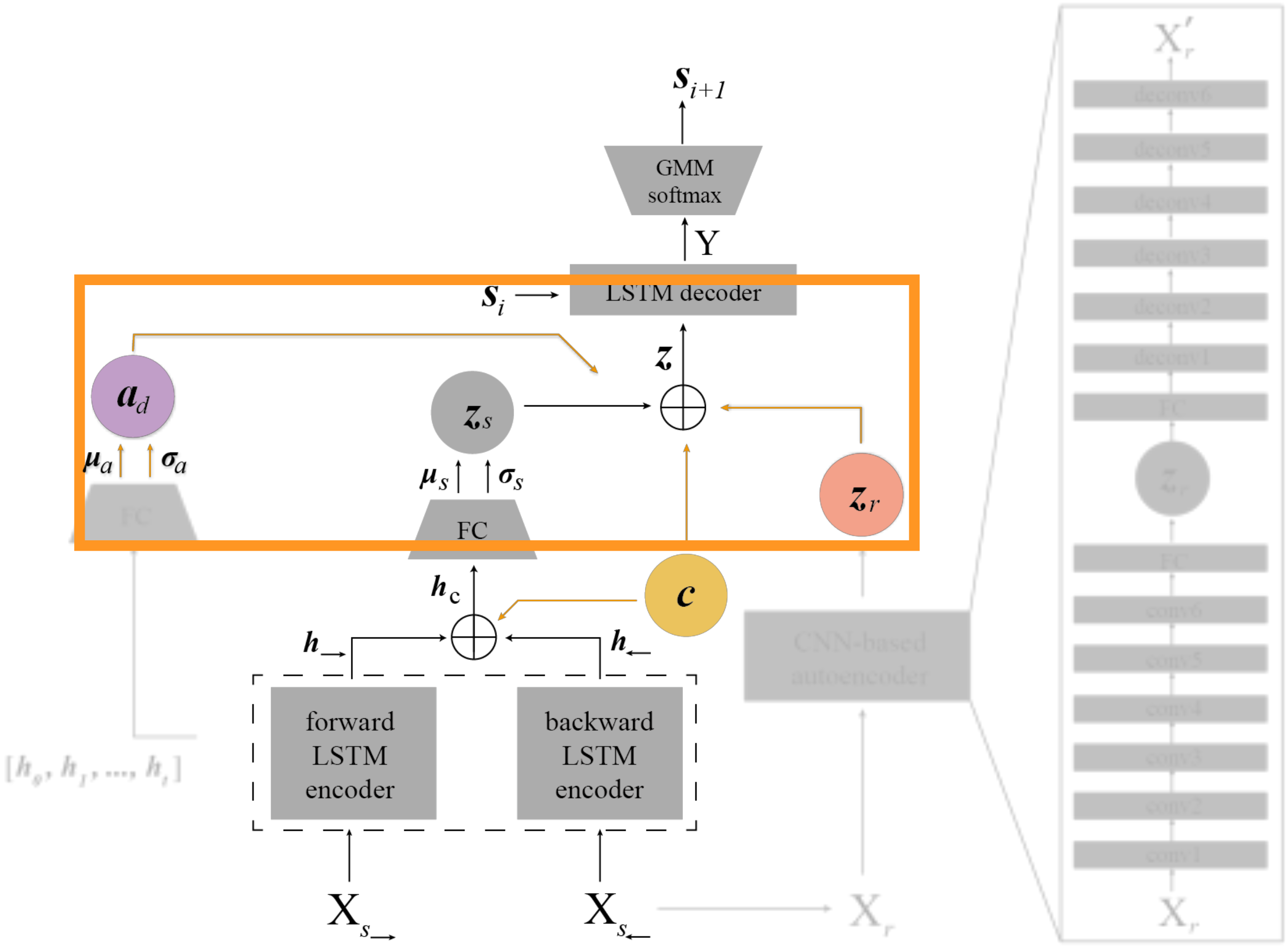


- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

A latent vector has been randomly sampled from the distributions for generating the next strokes:

$$z_s = \boldsymbol{\mu}_s + \boldsymbol{\sigma}_s \cdot \boldsymbol{\lambda}$$

AI-SKETCHER - Conditional Sequence-to-Sequence VAE



- **A conditional vector is used to ensure a high quality generation of sketches from multiple categories.**

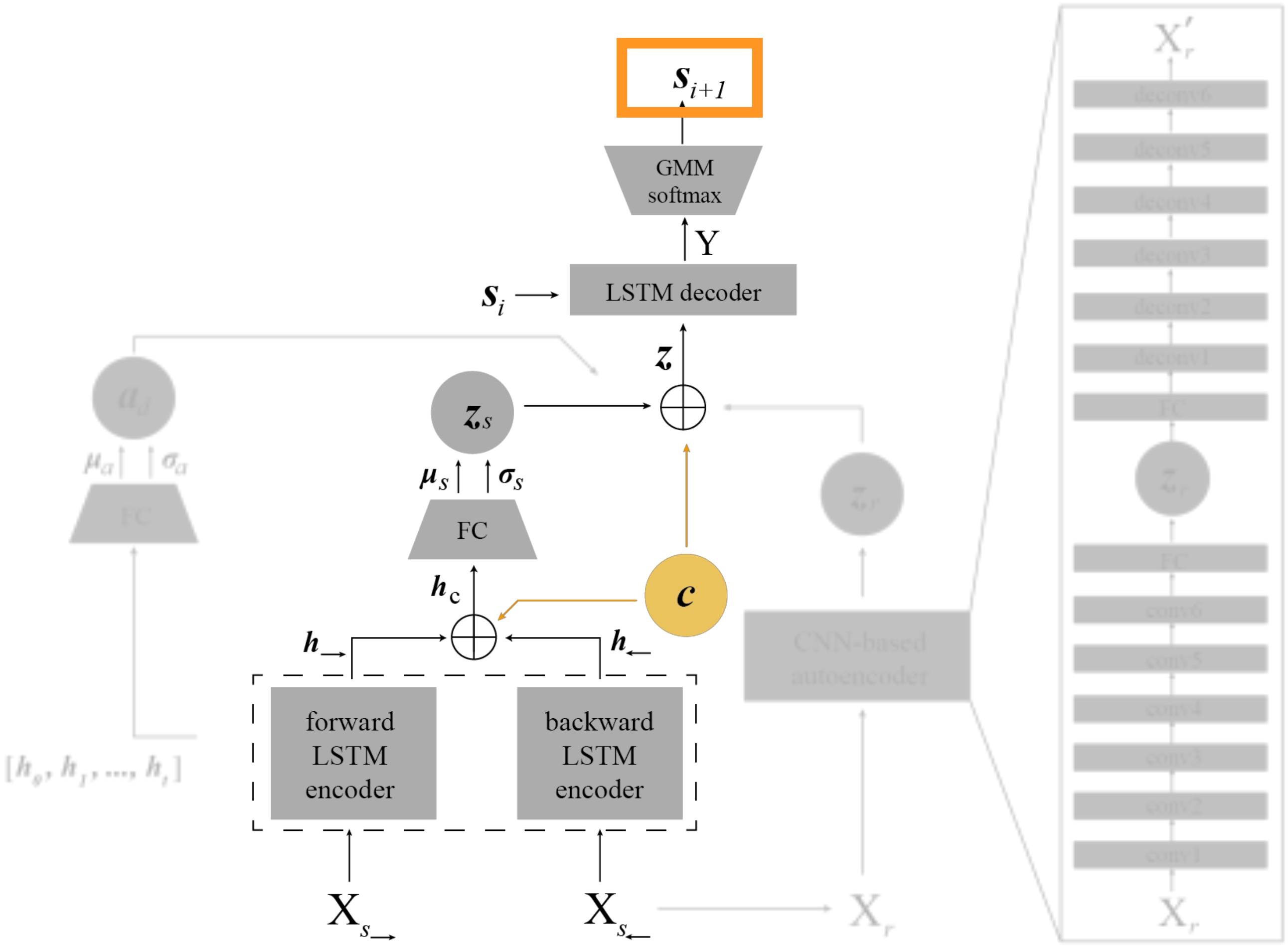
For decoding

z_s is concatenated together with the image feature vector z_r, the latent influence vector a_d, the conditional vector c and the last stroke vector S_i:

$$z = [z_s; z_r; \ a_d; \ c; \ s_i]$$

$$h^{dec} = decode(z)$$

AI-SKETCHER - Conditional Sequence-to-Sequence VAE

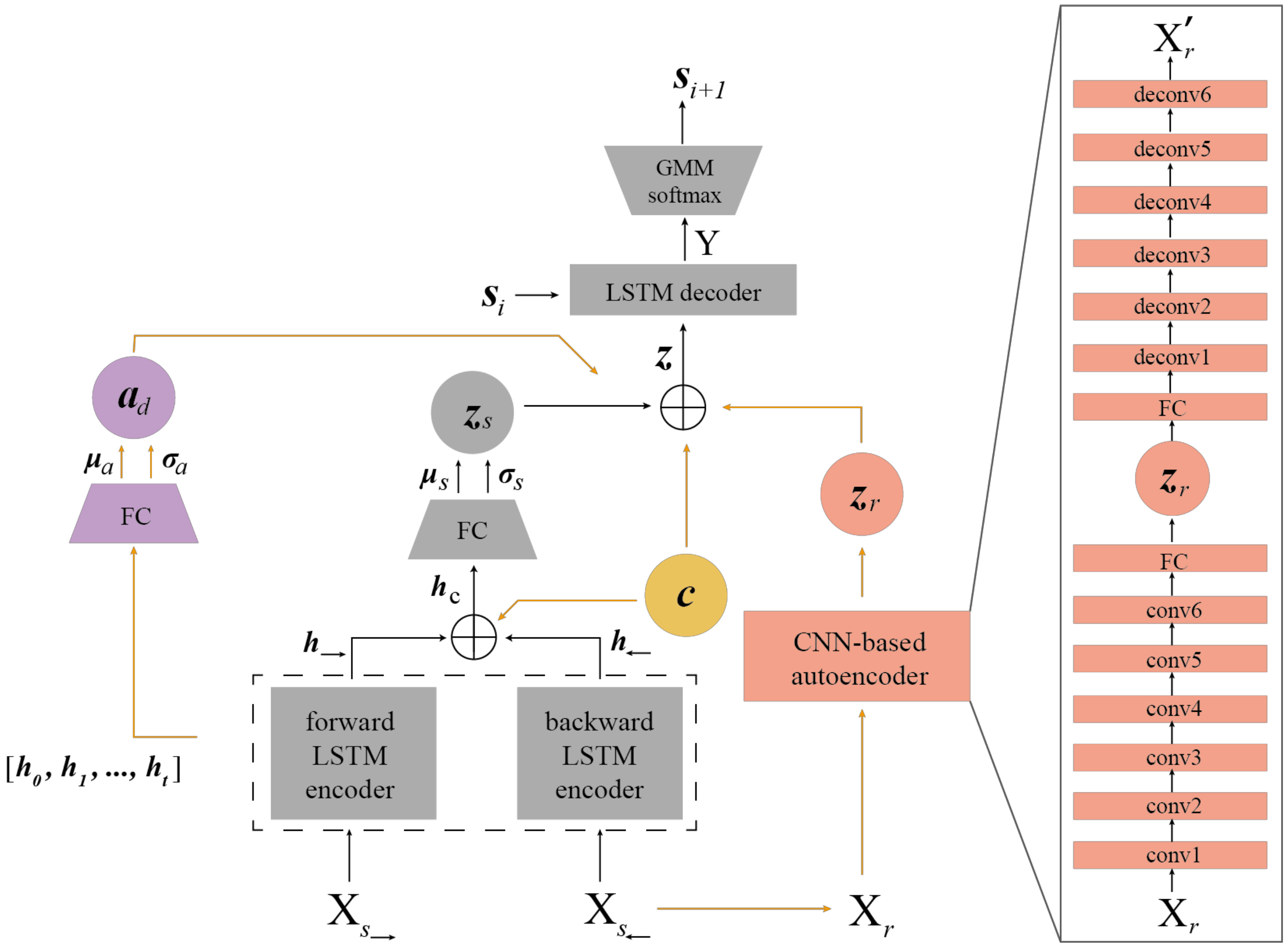


- A conditional vector is used to ensure a high quality generation of sketches from multiple categories.

predict the probabilities of the relative position of the next drawing point:

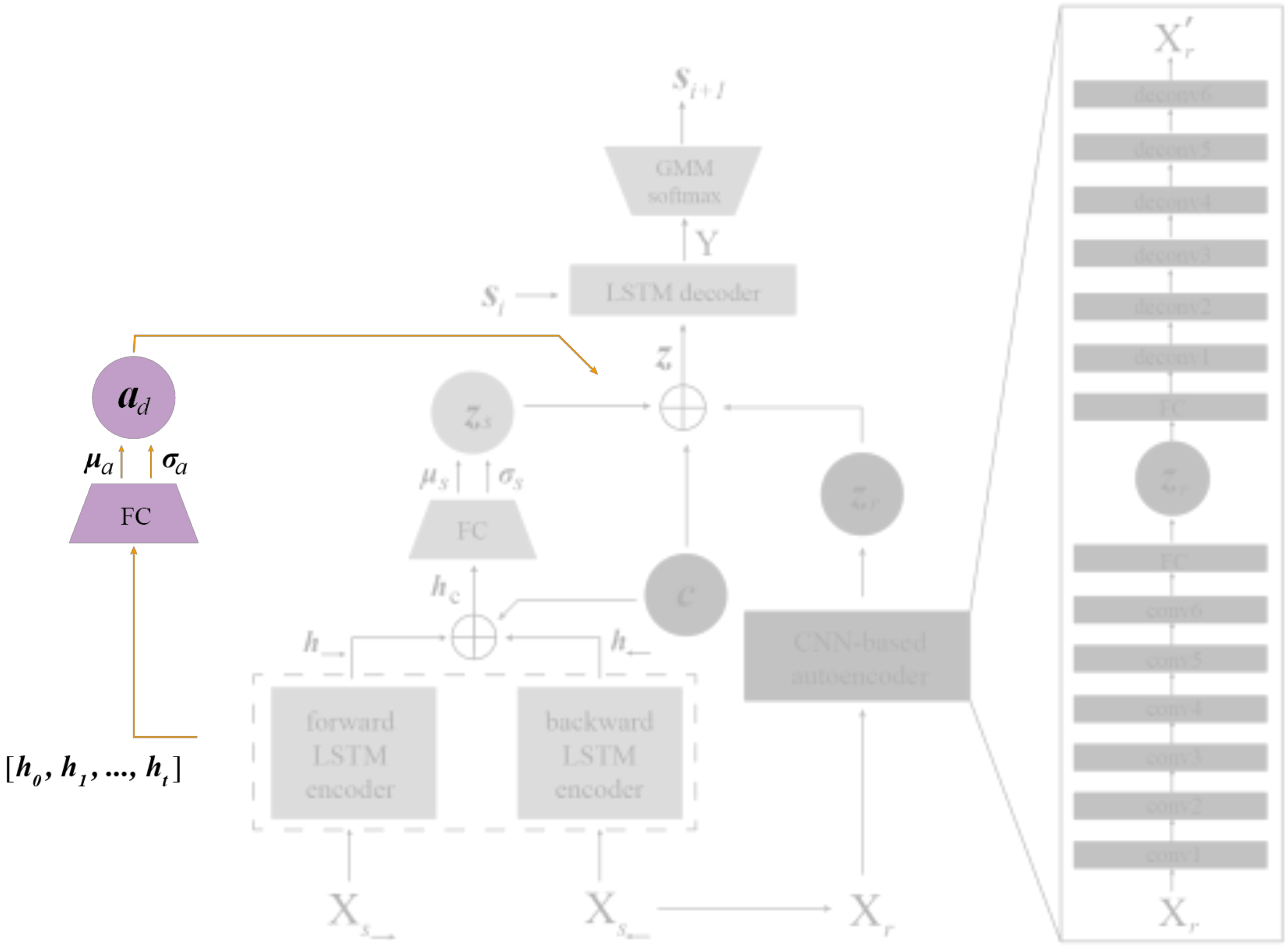
$$p(\Delta x_{i+1}, \Delta y_{i+1})$$

AI-SKETCHER



- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function** is modified.

AI-SKETCHER



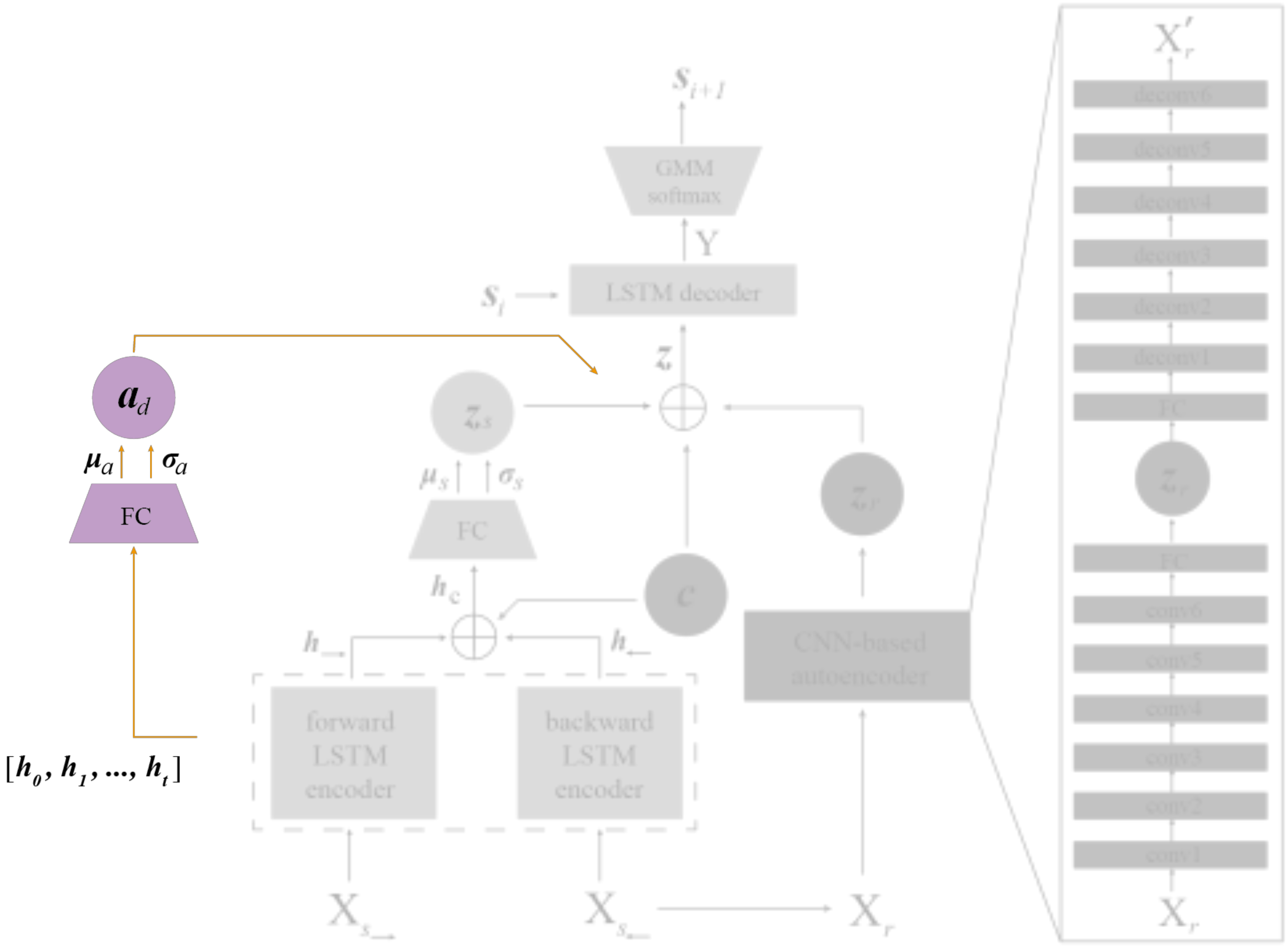
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.

- **An influence layer is introduced to estimate how the previous strokes will influence on the next stroke.**

- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.

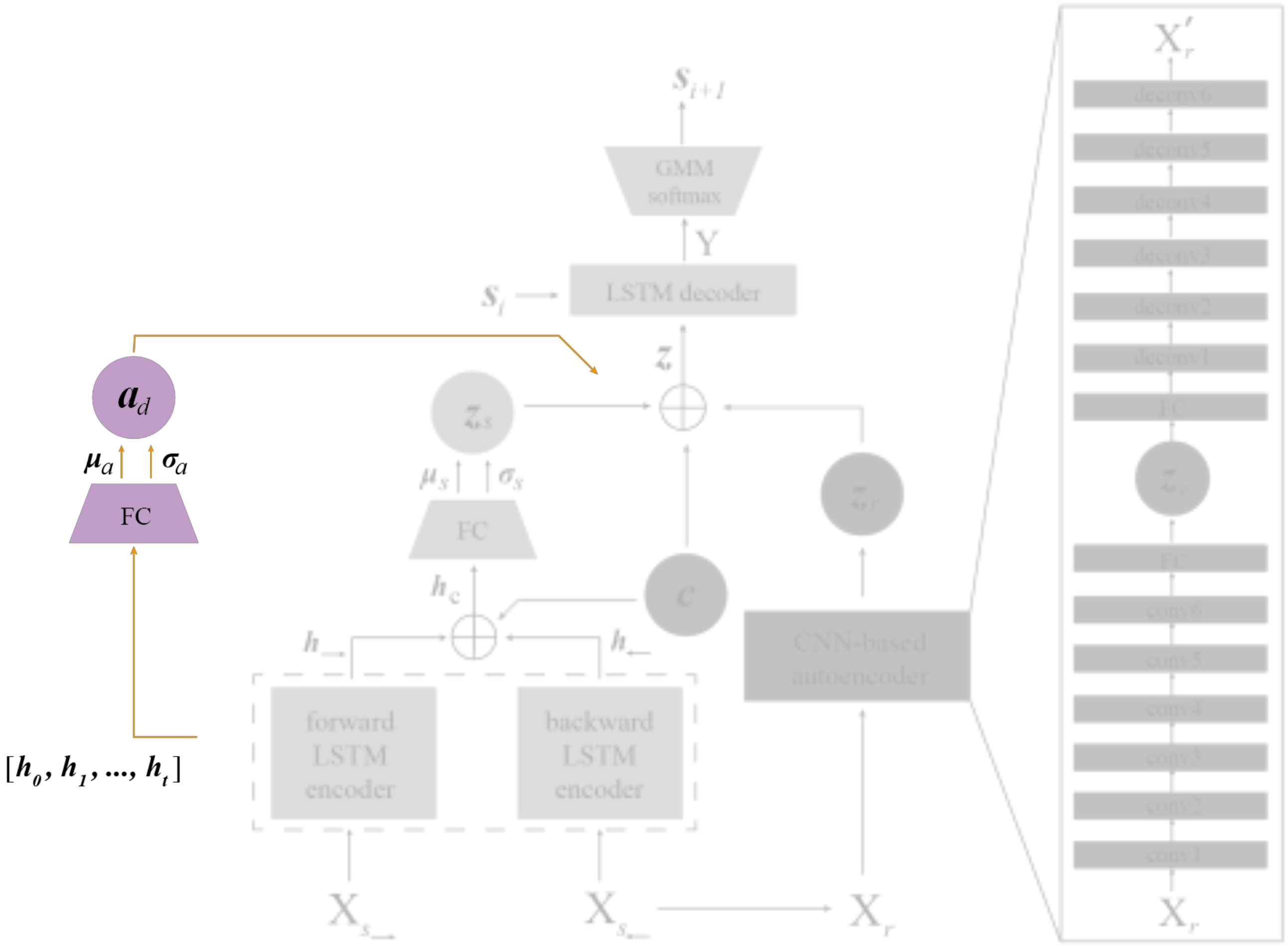
- **Loss function** is modified.

AI-SKETCHER - Influence Layer



- **An influence layer is introduced to estimate how the previous strokes will influence on the next stroke.**

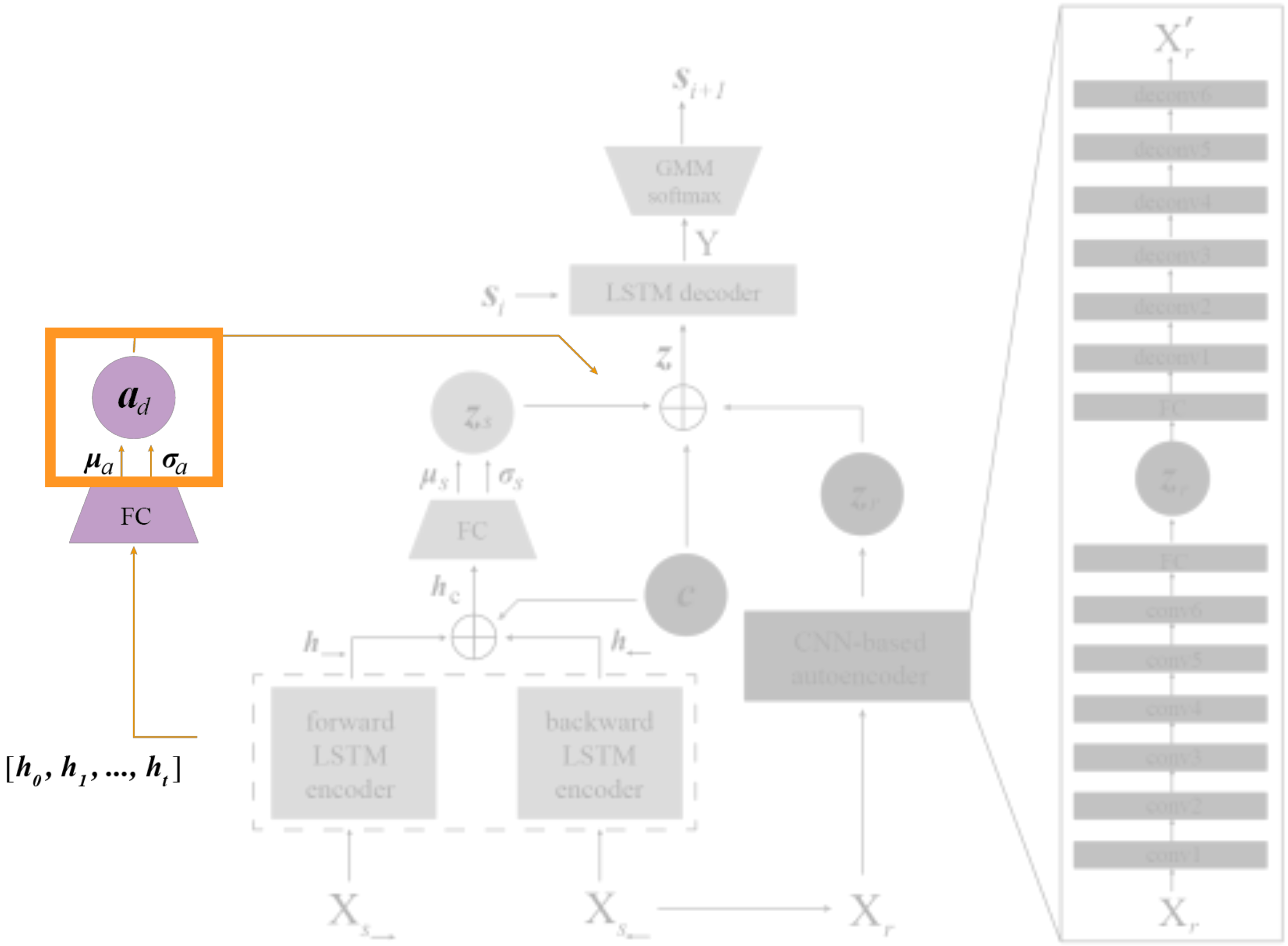
AI-SKETCHER - Influence Layer



- **An influence layer is introduced to estimate how the previous strokes will influence on the next stroke.**

It considers **all the previous hidden node values** until the latest drawing step in the RNN encoder.

AI-SKETCHER - Influence Layer

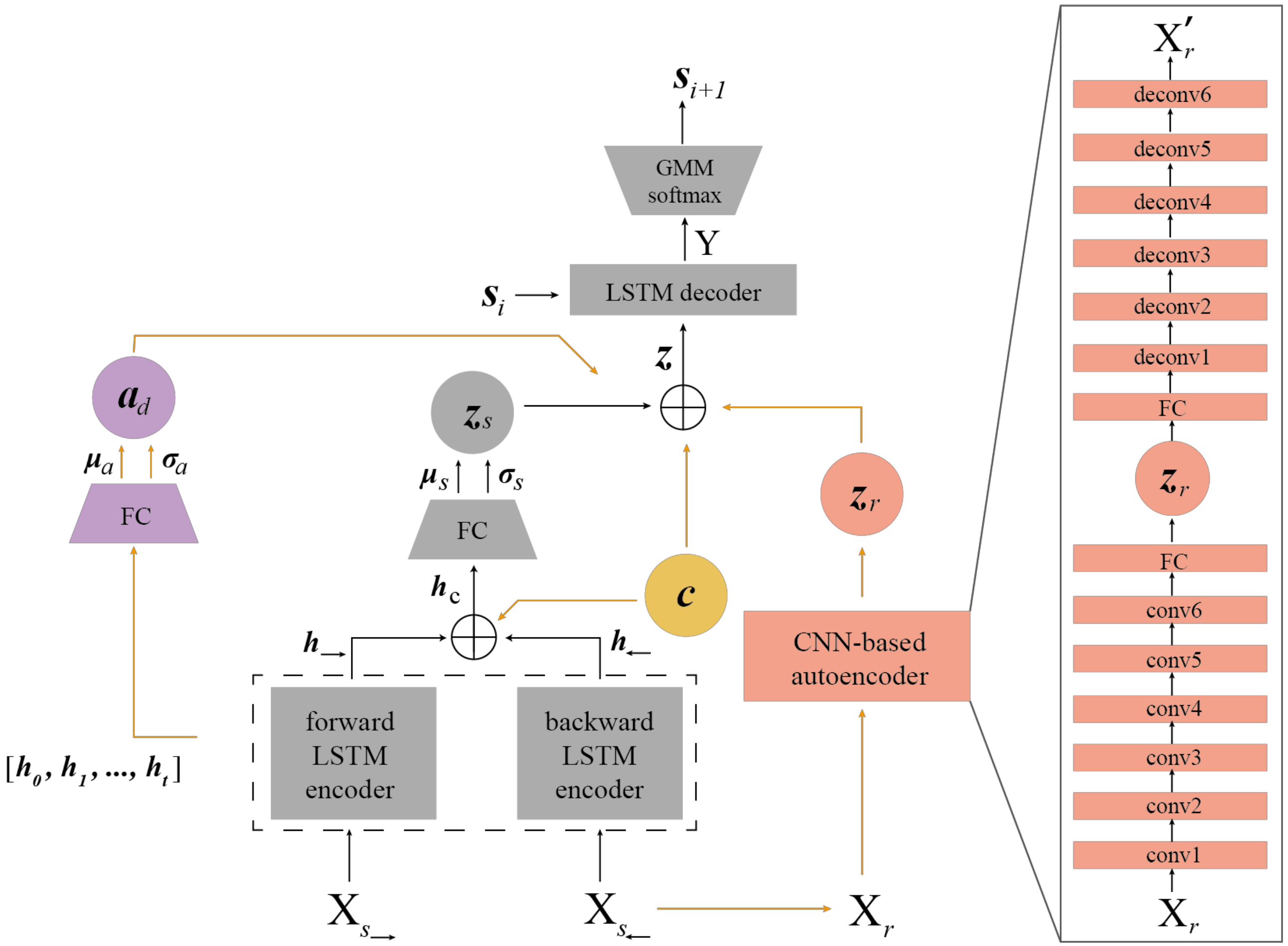


- **An influence layer is introduced to estimate how the previous strokes will influence on the next stroke.**

The influence vector $\boldsymbol{a_d}$ is a latent vector whose fields are sampled from these normal distributions:

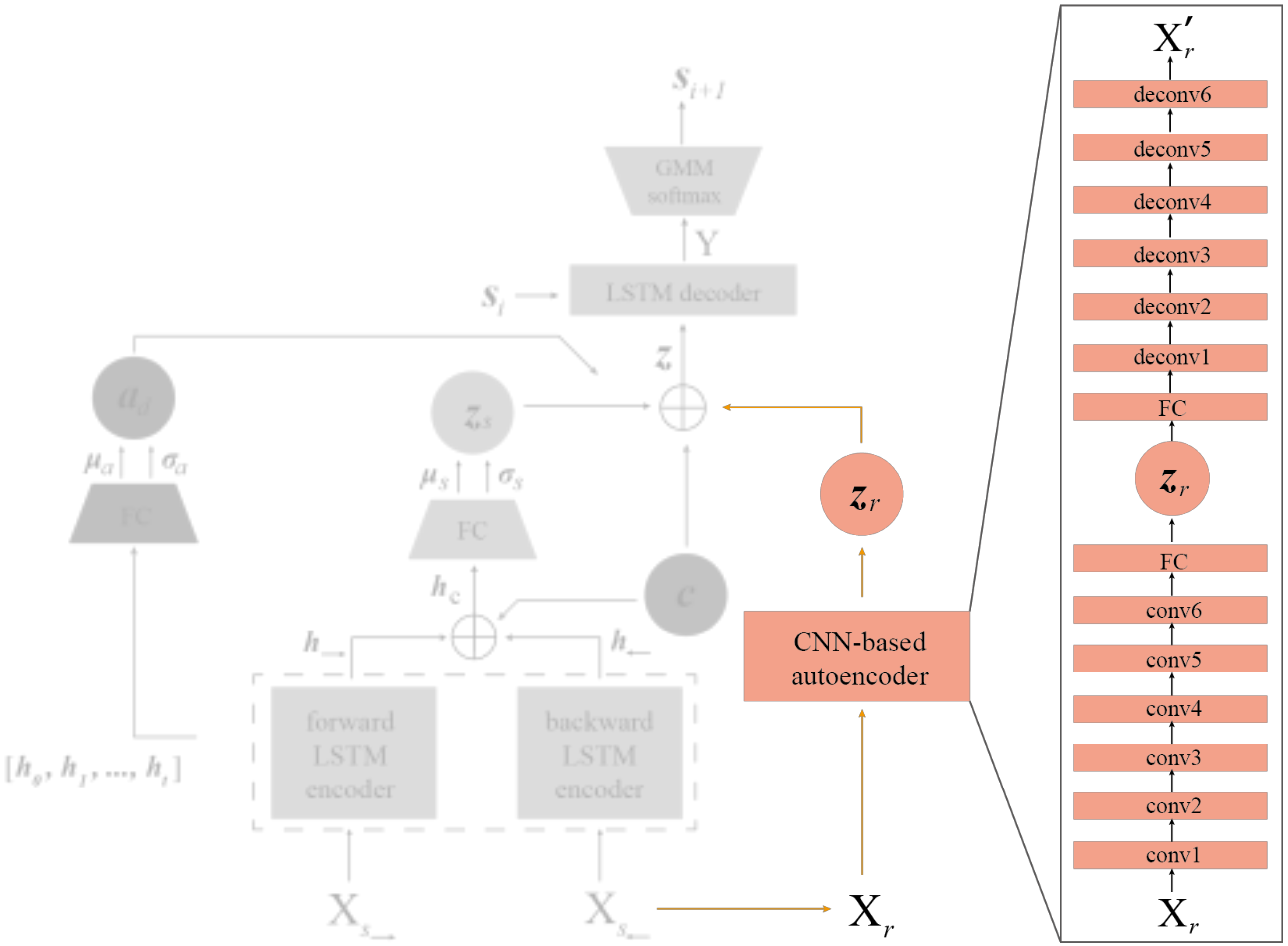
$$\boldsymbol{a}_d = \boldsymbol{\mu}_a + \boldsymbol{\sigma}_a \cdot \boldsymbol{\lambda}_a$$

AI-SKETCHER



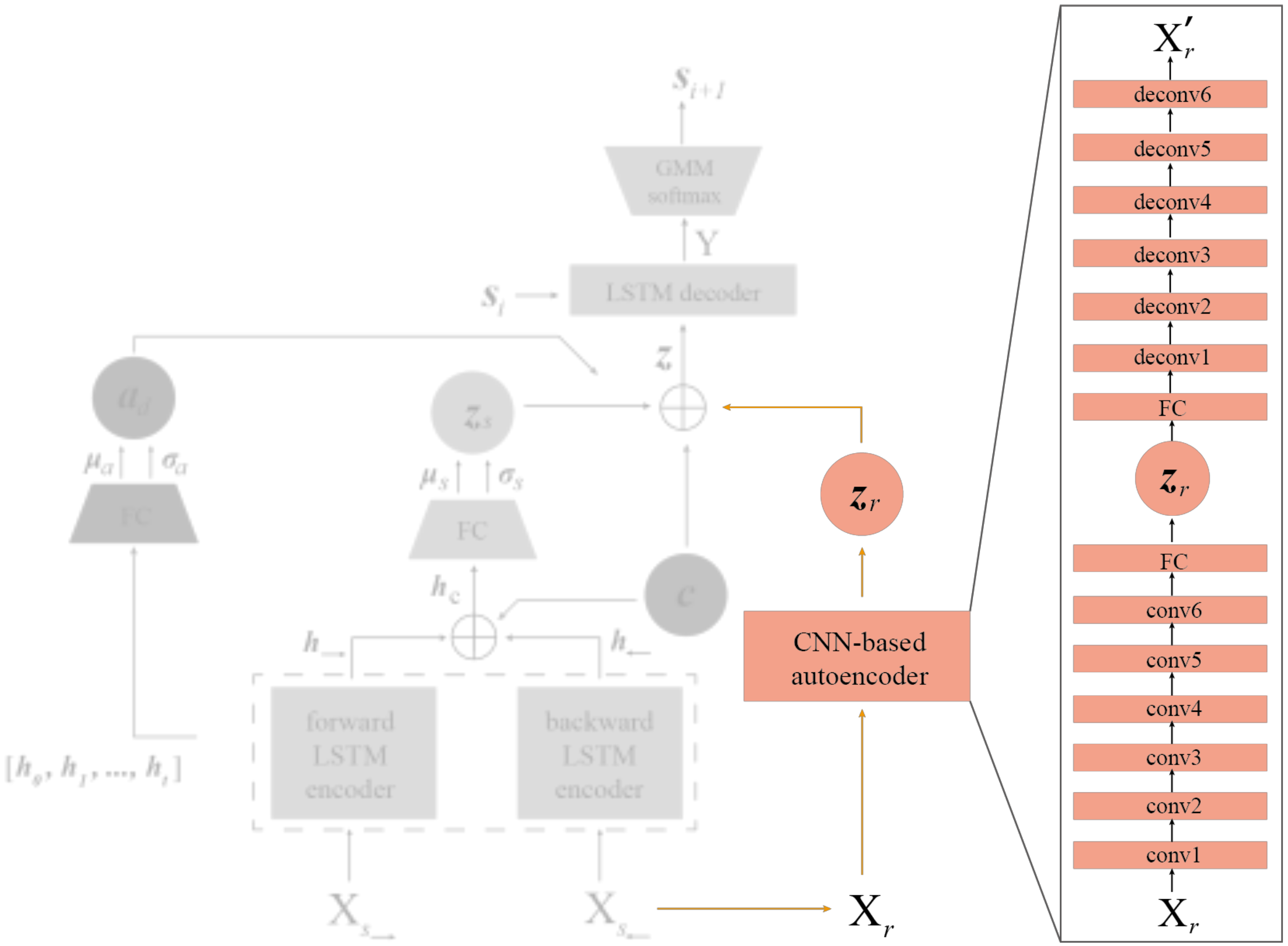
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function** is modified.

AI-SKETCHER



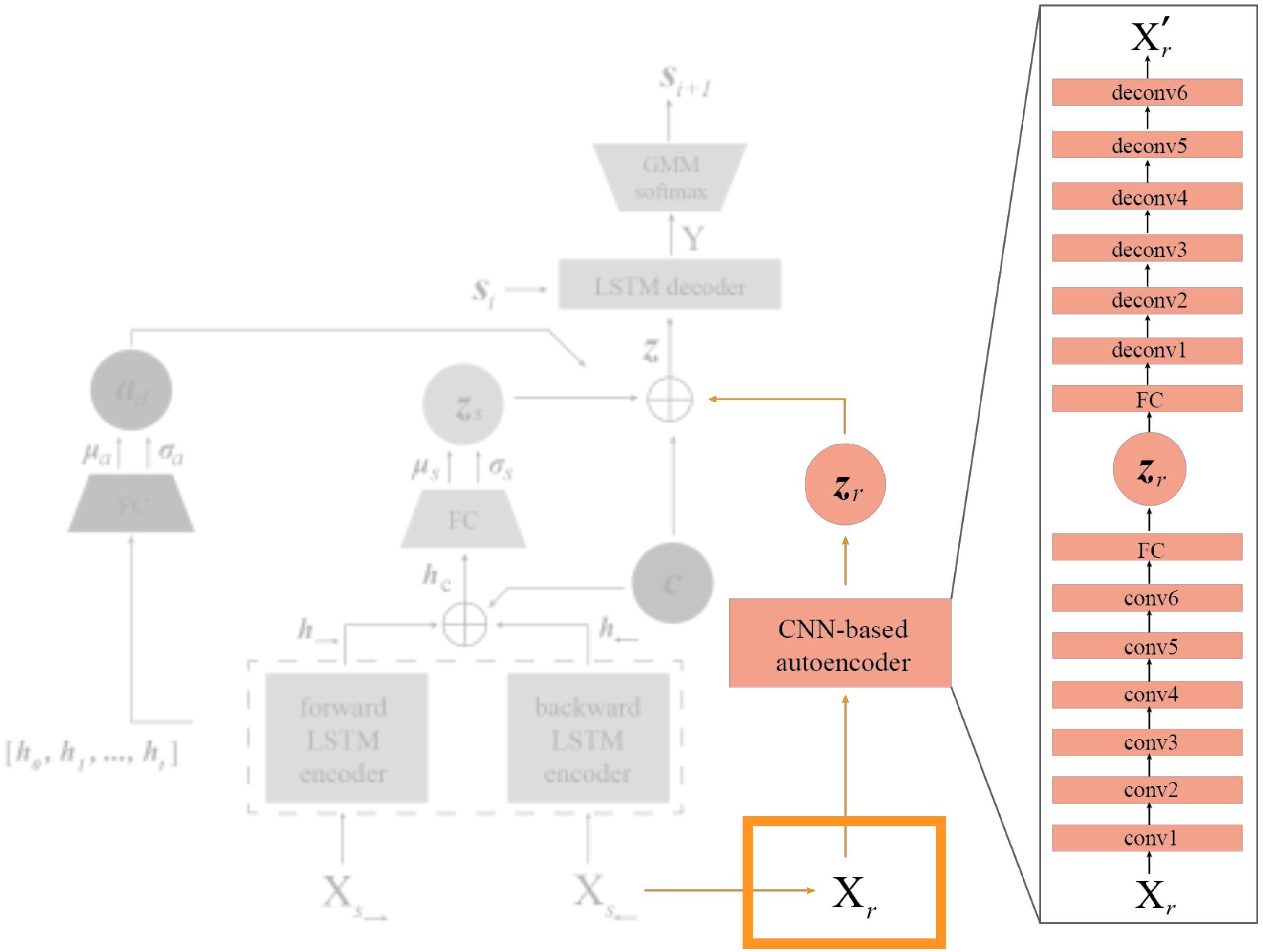
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**
- **Loss function** is modified.

AI-SKETCHER - CNN-based Autoencoder



- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

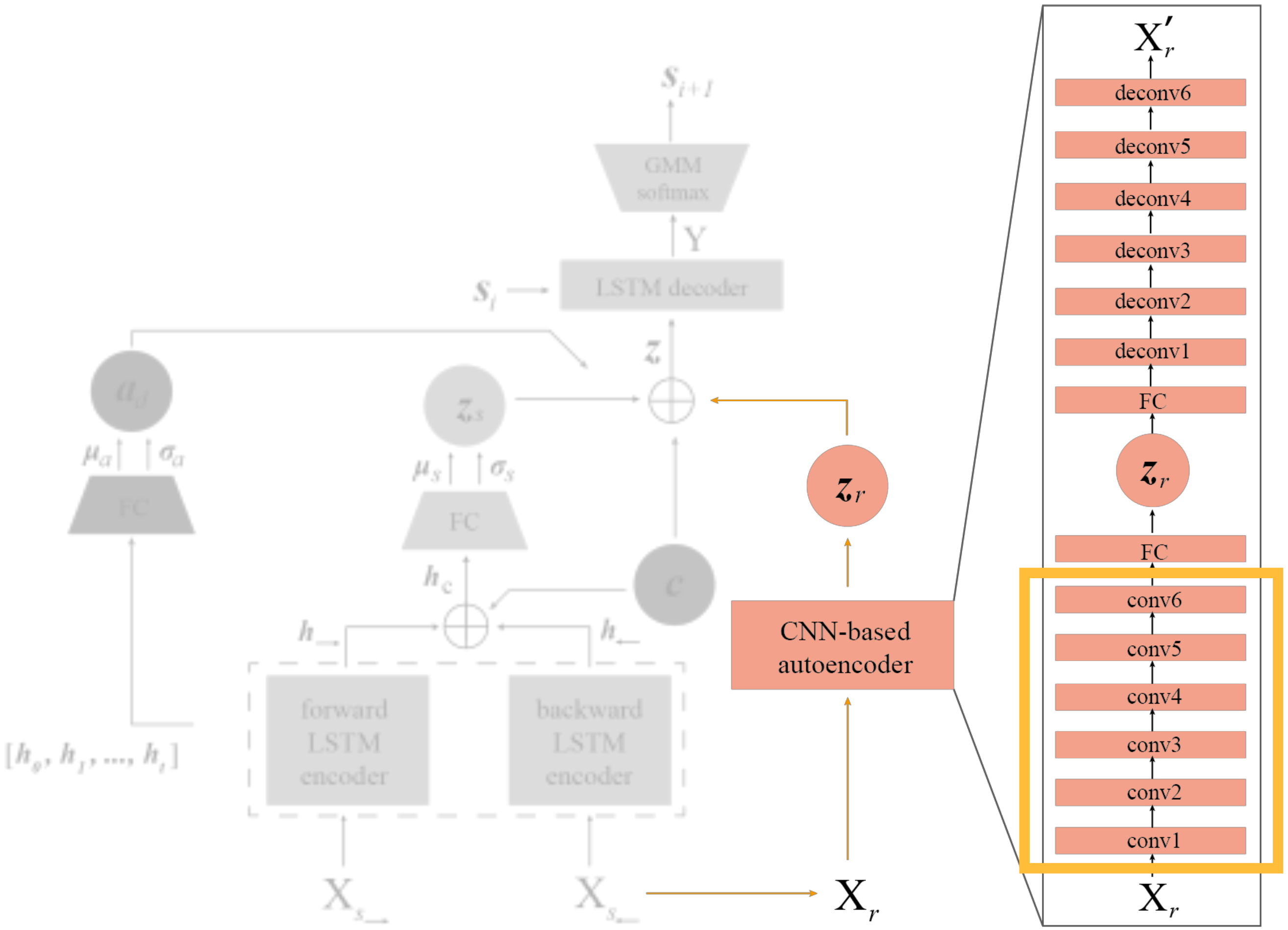
AI-SKETCHER - CNN-based Autoencoder



- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

X_r: the input raster image matrix.

AI-SKETCHER - CNN-based Autoencoder

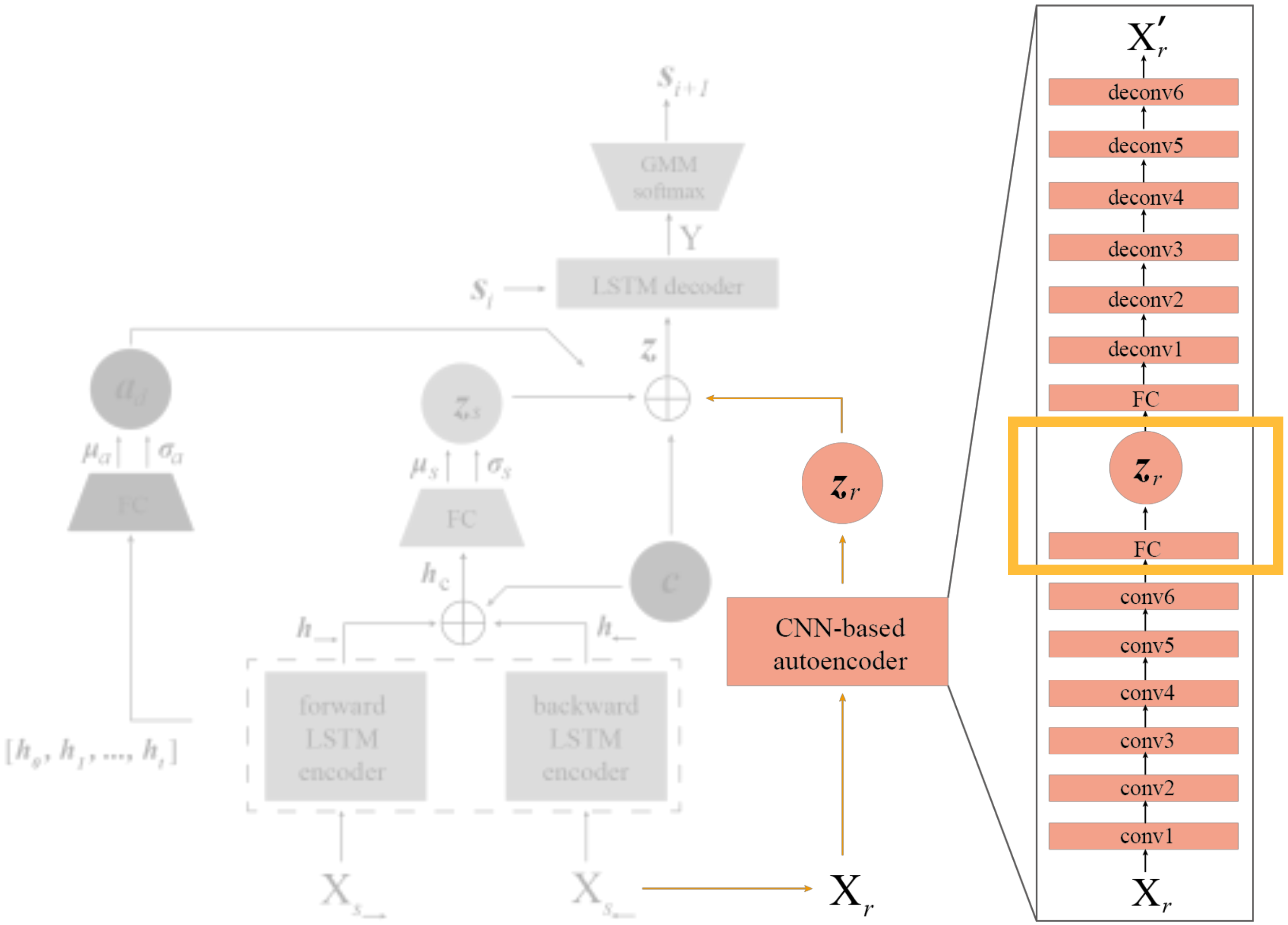


- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

Encoder:

- three convolutional layers with the stride size as 2.
- the other three layers with the stride size as 1.

AI-SKETCHER - CNN-based Autoencoder

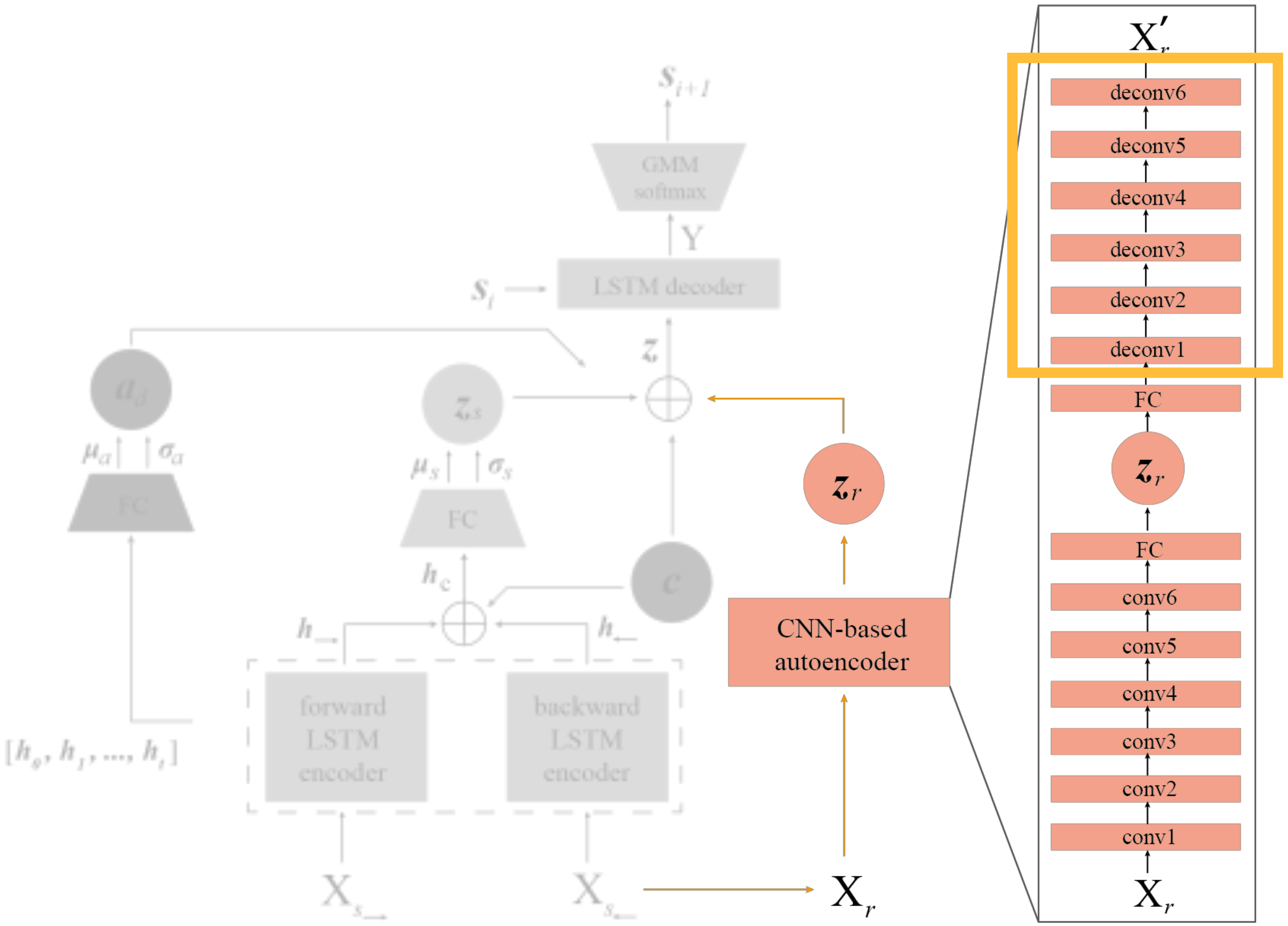


- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

Encoder:

- The last layer is a fully-connected neural network to produce the latent feature vector z_r with 128 dimensions.

AI-SKETCHER - CNN-based Autoencoder

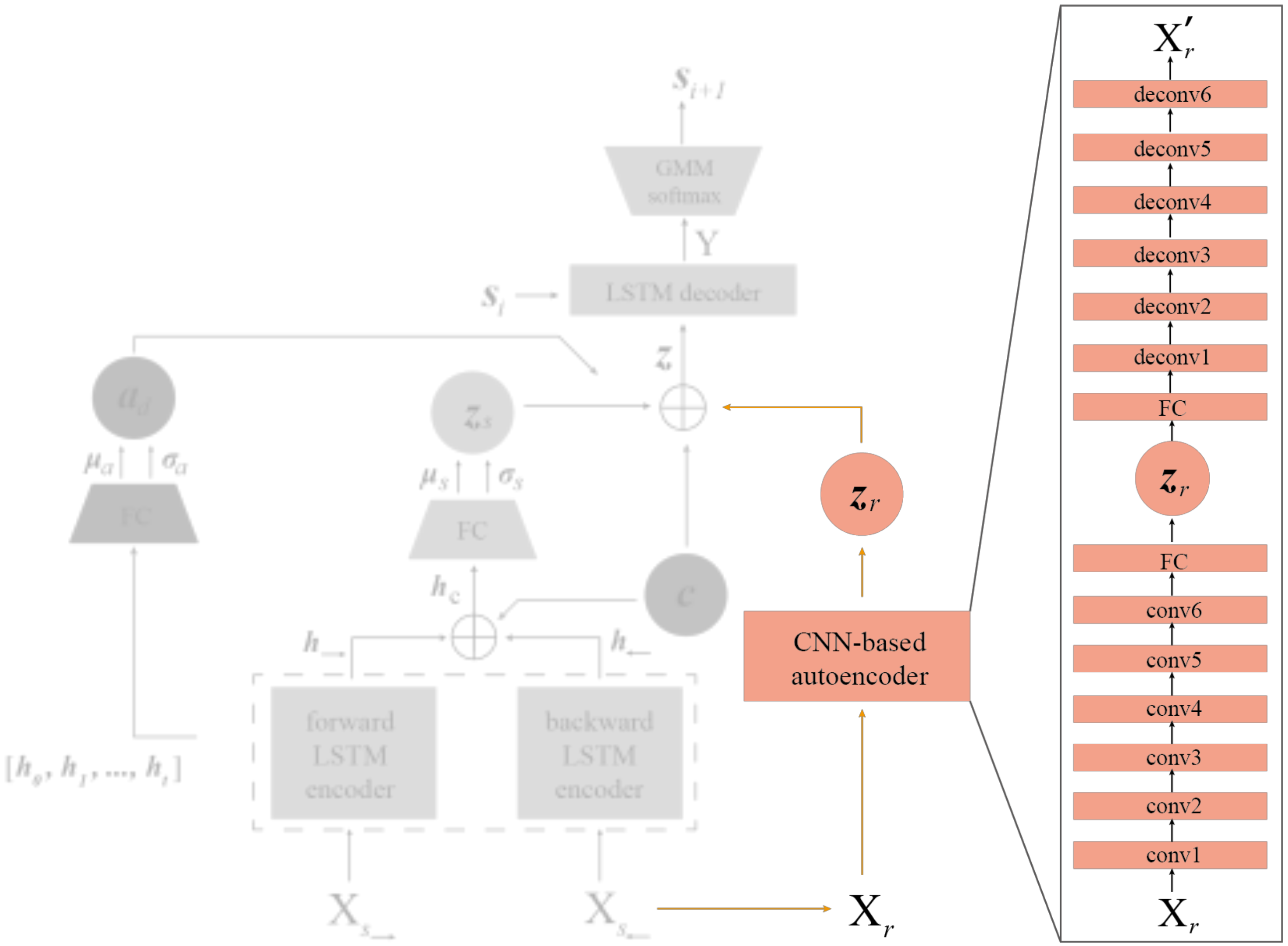


- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

Decoder:

- three deconvolutional layers with stride sizes equal to 2.
- the other three layers with stride sizes equal to 1.

AI-SKETCHER - CNN-based Autoencoder

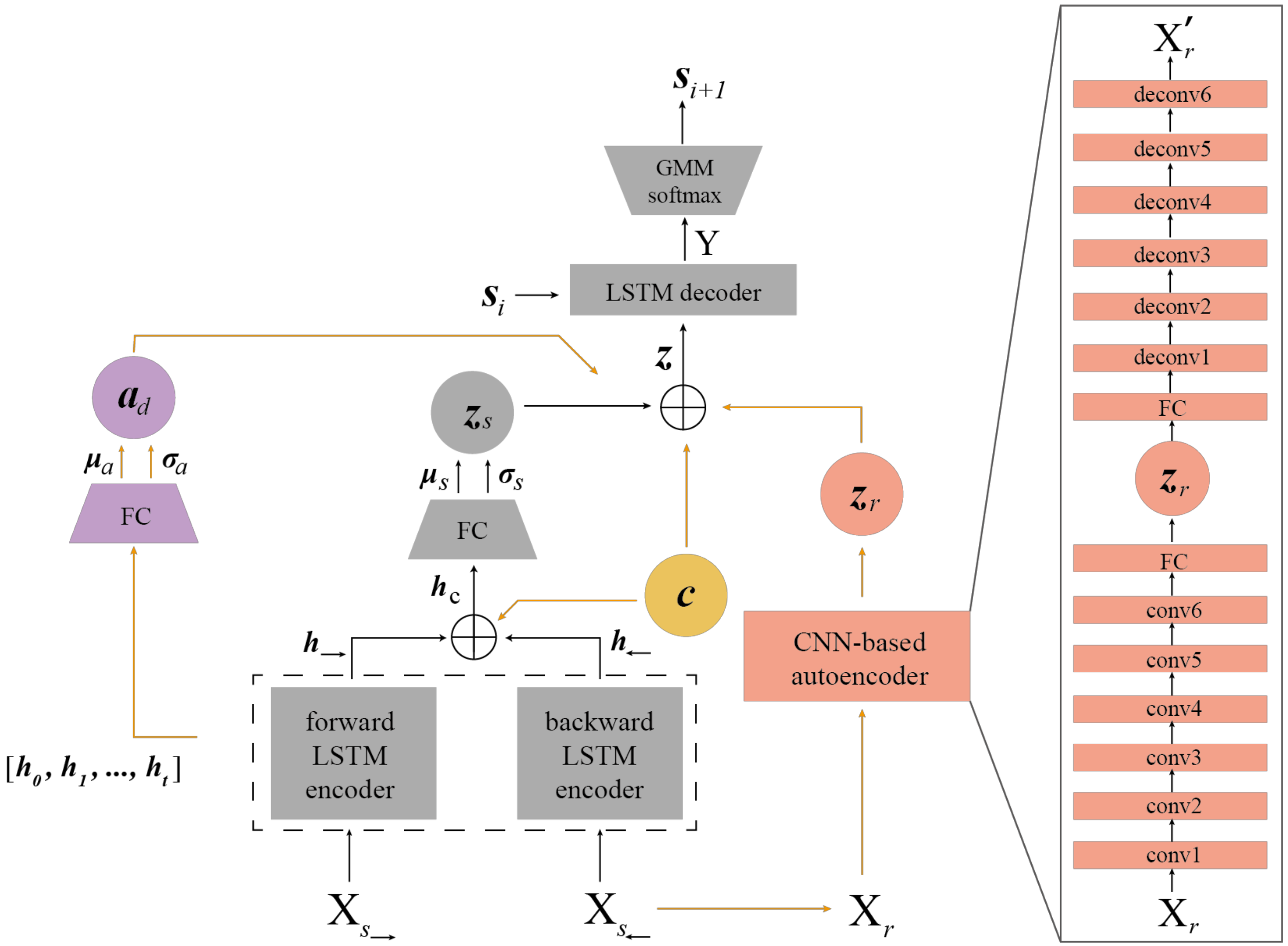


- **A CNN-based autoencoder is employed to capture the spatial information of a training set.**

ReLU is used as the activation function in both convolutional and deconvolutional layers.

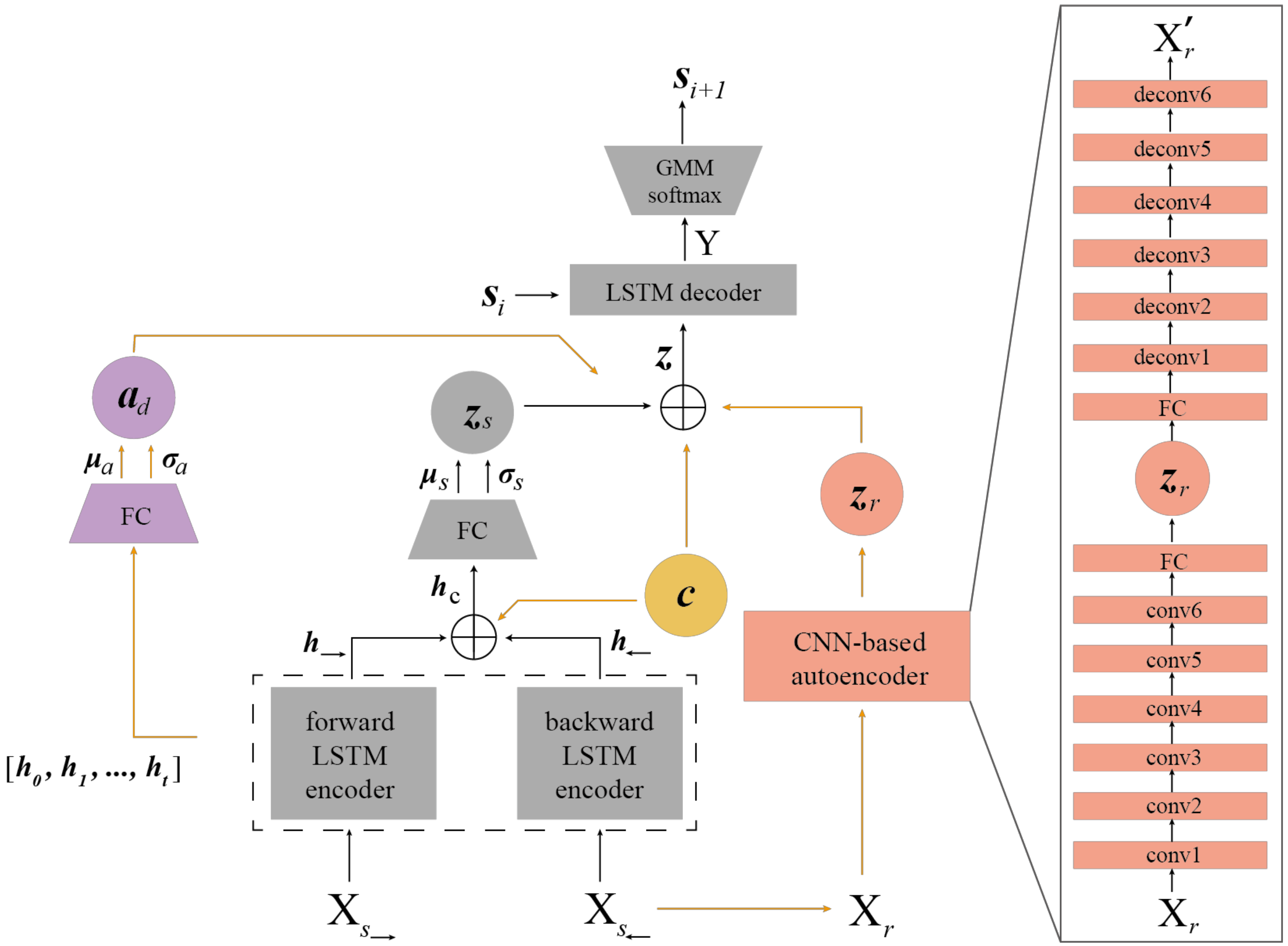
tanh is used as the activation function of the fully-connected neural network.

AI-SKETCHER



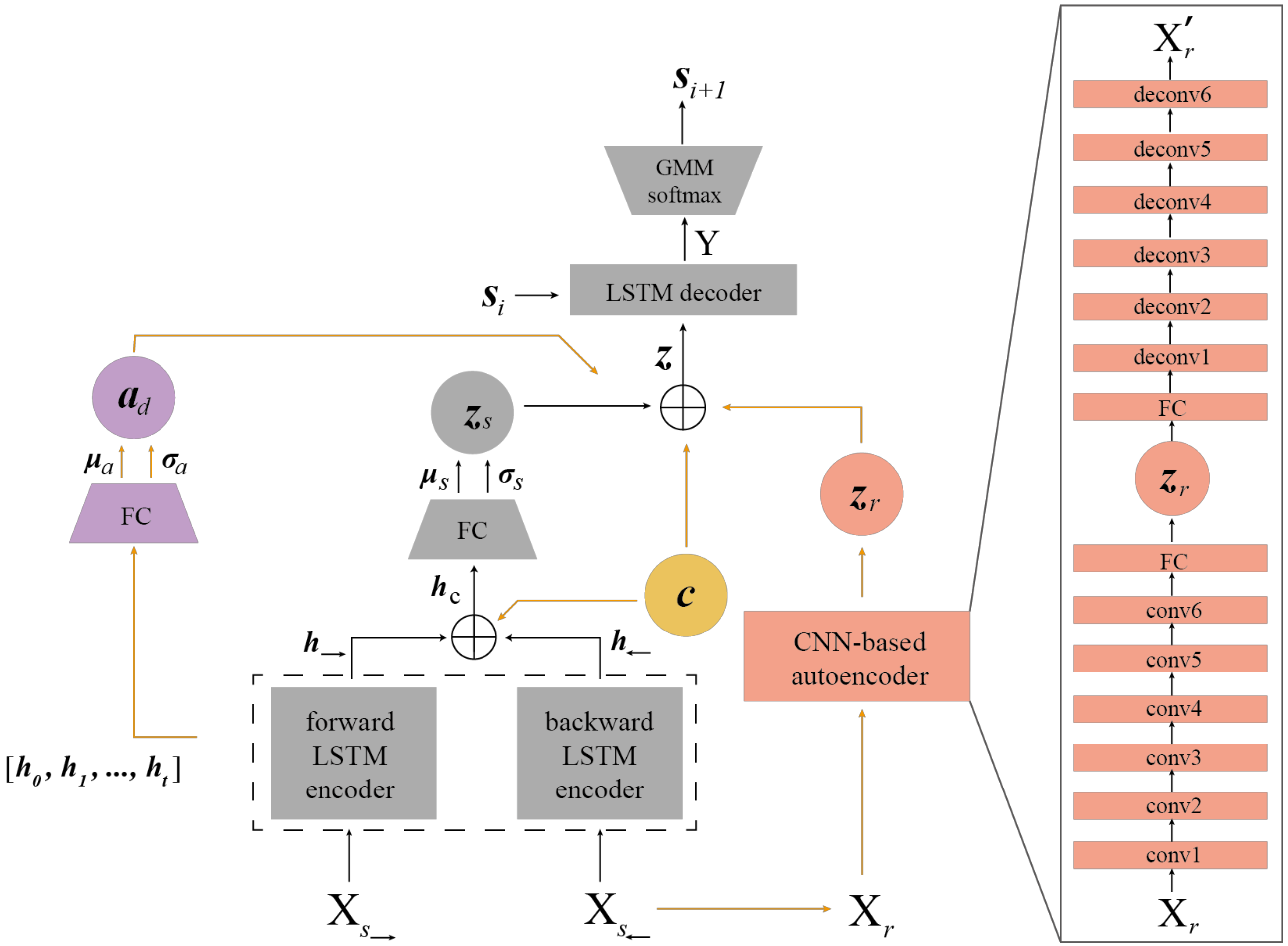
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function** is modified.

AI-SKETCHER



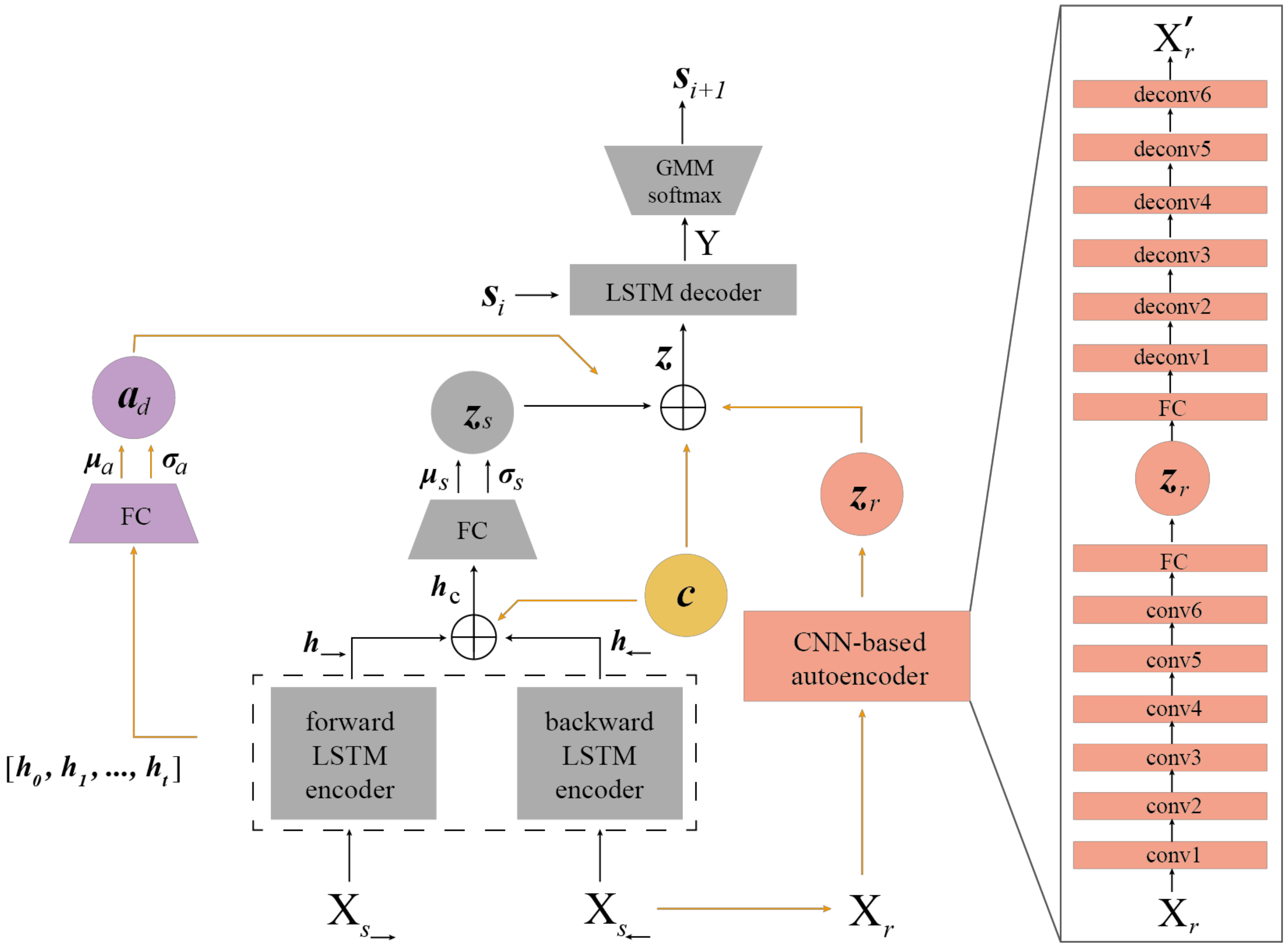
- **A conditional vector** is used to ensure a high quality generation of sketches from multiple categories.
- **An influence layer** is introduced to estimate how the previous strokes will influence on the next stroke.
- **A CNN-based autoencoder** is employed to capture the spatial information of a training set.
- **Loss function is modified.**

AI-SKETCHER - Loss function



- **Loss function is modified.**

AI-SKETCHER - Loss function



- **Loss function is modified.**

$$Loss = l_r + \alpha \cdot max(l_{kl}, \epsilon)$$

AI-SKETCHER - Loss function

- **Loss function is modified.**

$$Loss = l_r + \alpha \cdot max(\boxed{l_{kl}}, \epsilon)$$

the reconstruction loss estimates the differences between the generated strokes and the training samples.

estimates the distribution differences between the generated strokes and the strokes in the training set modeled by the standard normal distribution.

AI-SKETCHER - Loss function

- **Loss function is modified.**

$$Loss = l_r + \alpha \cdot max(\boxed{l_{kl}}, \epsilon)$$

$$\begin{aligned}
l_z &= -\frac{1}{2n_z}\sum_{i=1}^{n_z}(1 + \boldsymbol{\sigma}_{s_i} - exp(\boldsymbol{\sigma}_{s_i}) - \boldsymbol{\mu}_{s_i}^2) \\
l_a &= -\frac{1}{2n_a}\sum_{j=1}^{n_a}(1 + \boldsymbol{\sigma}_{a_j} - exp(\boldsymbol{\sigma}_{a_j}) - \boldsymbol{\mu}_{a_j}^2) \\
l_{kl} &= l_z + \beta l_a
\end{aligned}$$

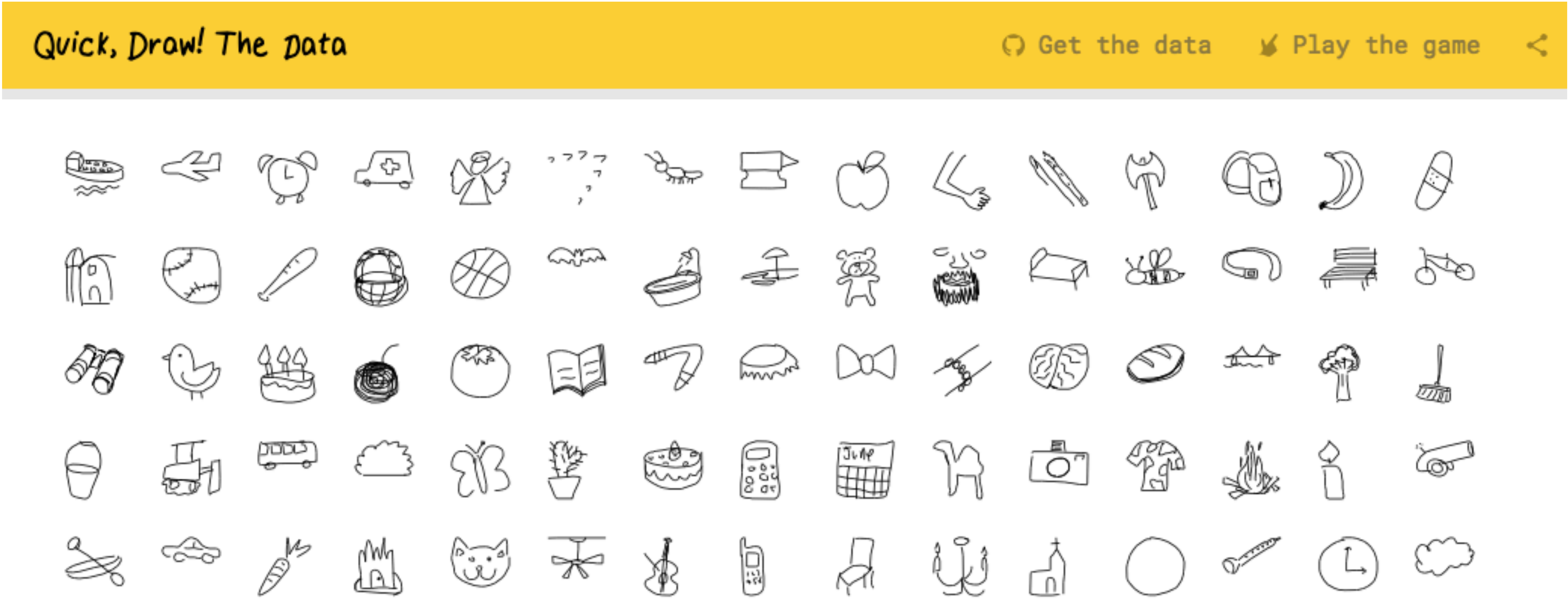
EVALUATION

We performed three experiments in purpose of validating the AI-Sketcher's

- **drawing quality**
- capability of generating sketches from **multiple classes**
- generation **diversity**

EVALUATION - Dataset

The QuickDraw dataset contains over 50 million sketches in 75 object categories and originally used for training Sketch-RNN.



https://quickdraw.withgoogle.com/

EVALUATION - Dataset

The FaceX dataset consists of 5 million sketches of both male's and female's facial expressions showing seven different types of emotions.

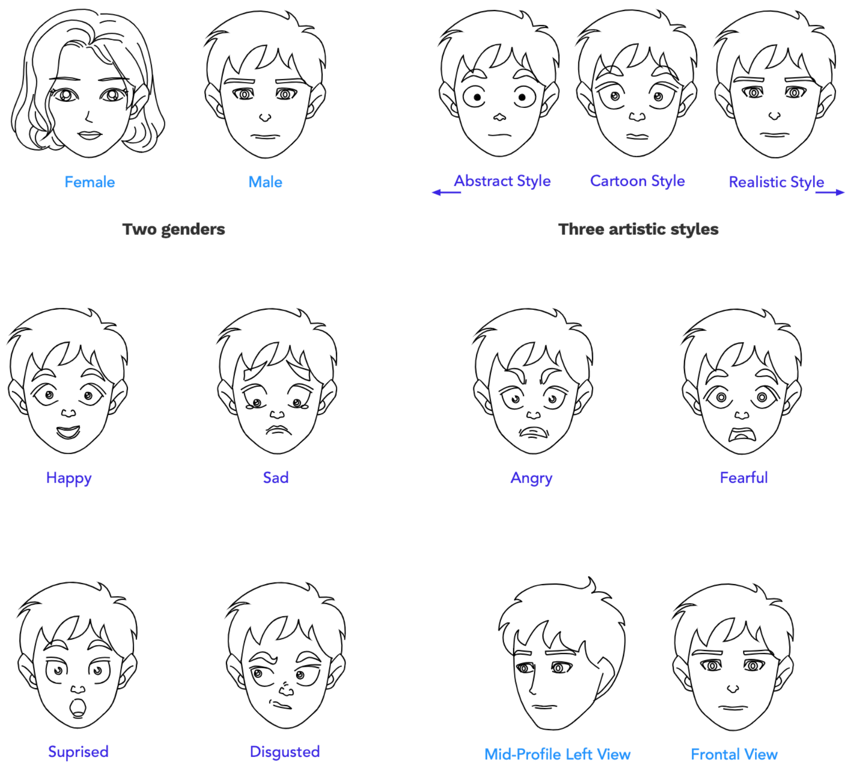
FaceX

A Dataset Containing 5,240,088 Hand-Drawing Sketches

The dataset contains over 5 million labeled facial sketches categorized by genders (male, female), viewing angles (frontal, mid-profile left view), emotions (neutral, happy, sad, angry, fearful, surprised, disgusted), and artistic styles (realistic, cartoon, abstract styles).

DOWNLOAD CONTRIBUTE

SVG format: 72 NPZ format: 63



https://facex.idvxlab.com/

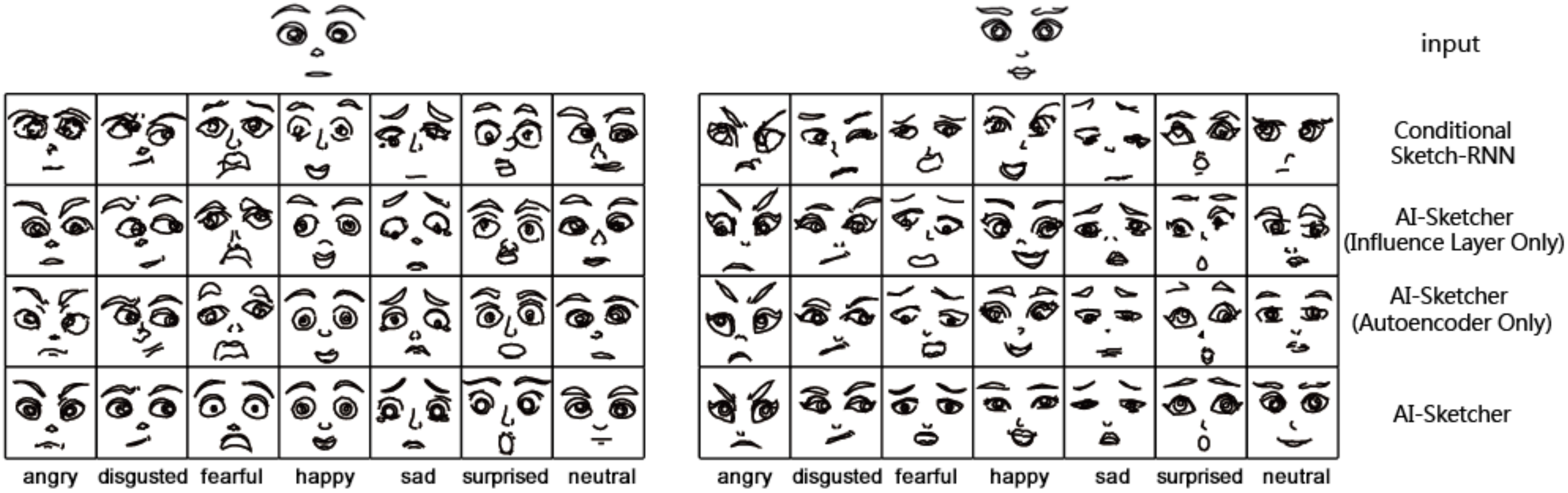
EVALUATION - Dataset

The FaceX dataset consi cial expressions
showing seven different t

https://facex.idvxlab.com/

EVALUATION - 1st Experiment

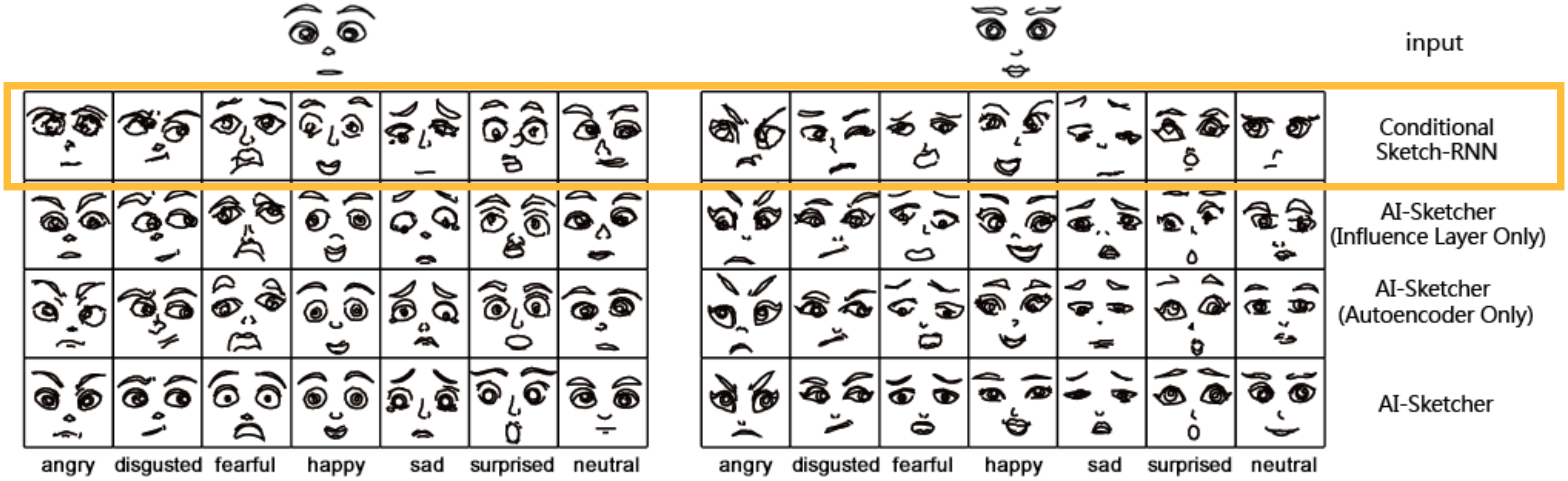
Drawing quality



Experiments based on FaceX Dataset

EVALUATION - 1st Experiment

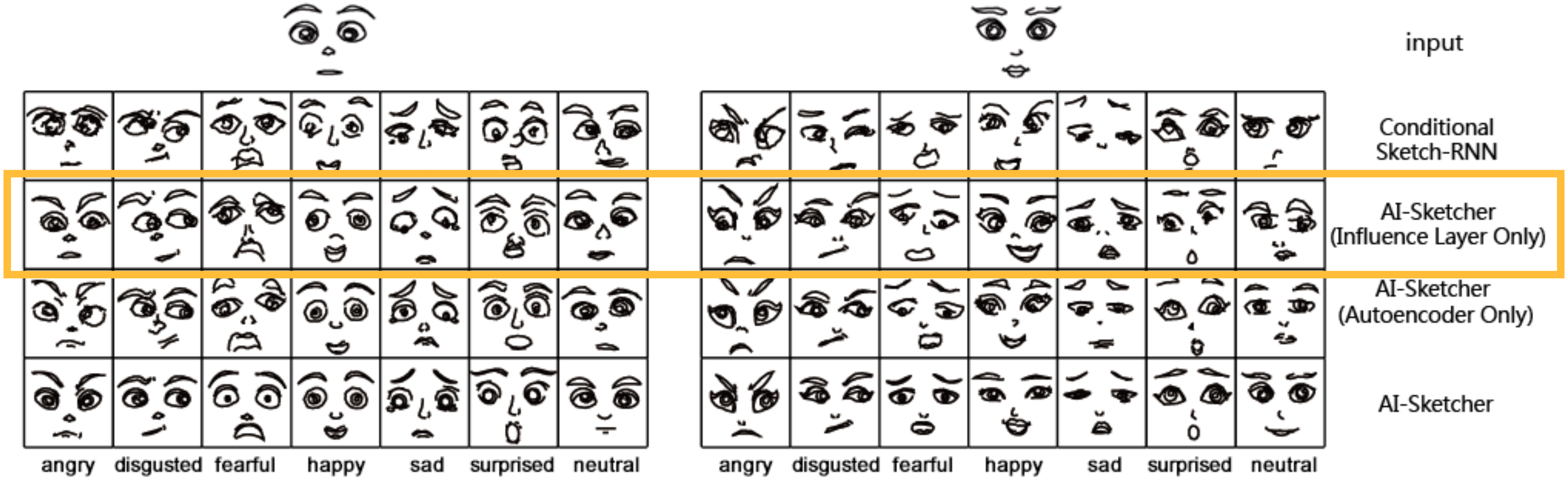
Drawing quality



Experiments based on FaceX Dataset

EVALUATION - 1st Experiment

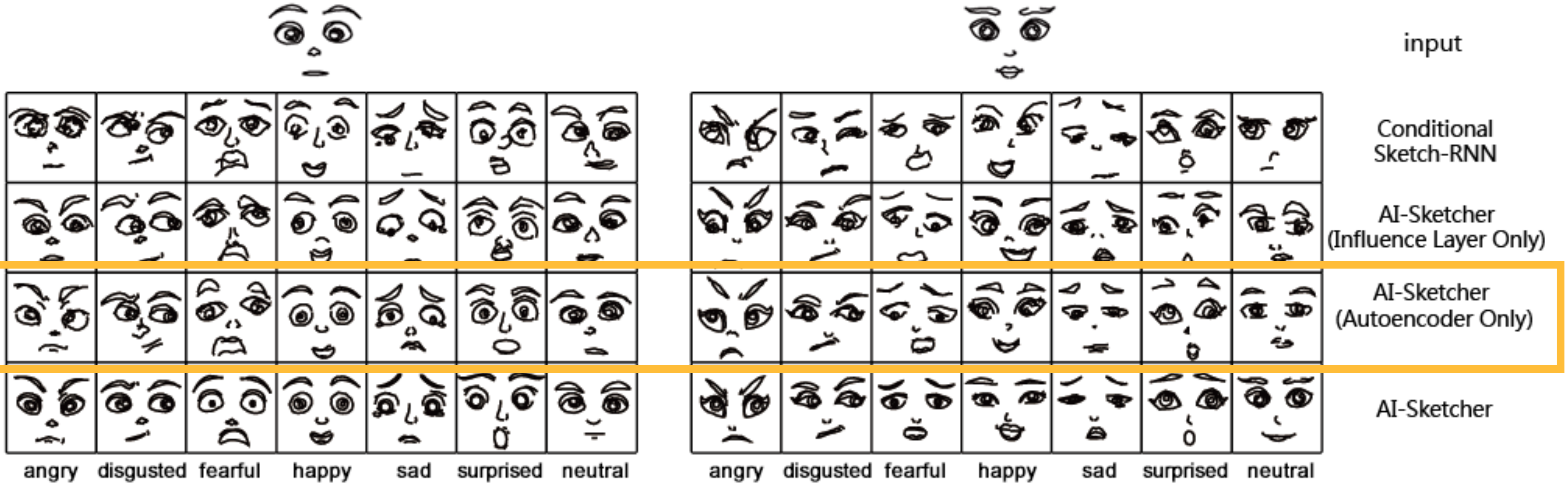
Drawing quality



Experiments based on FaceX Dataset

EVALUATION - 1st Experiment

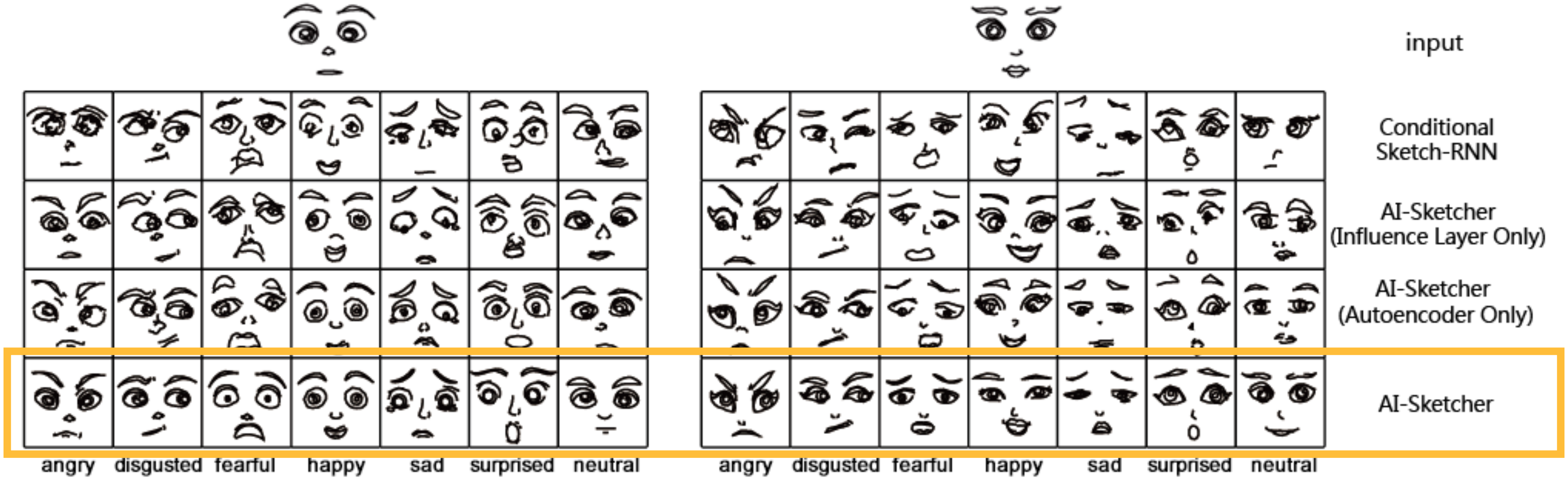
Drawing quality



Experiments based on FaceX Dataset

EVALUATION - 1st Experiment

Drawing quality



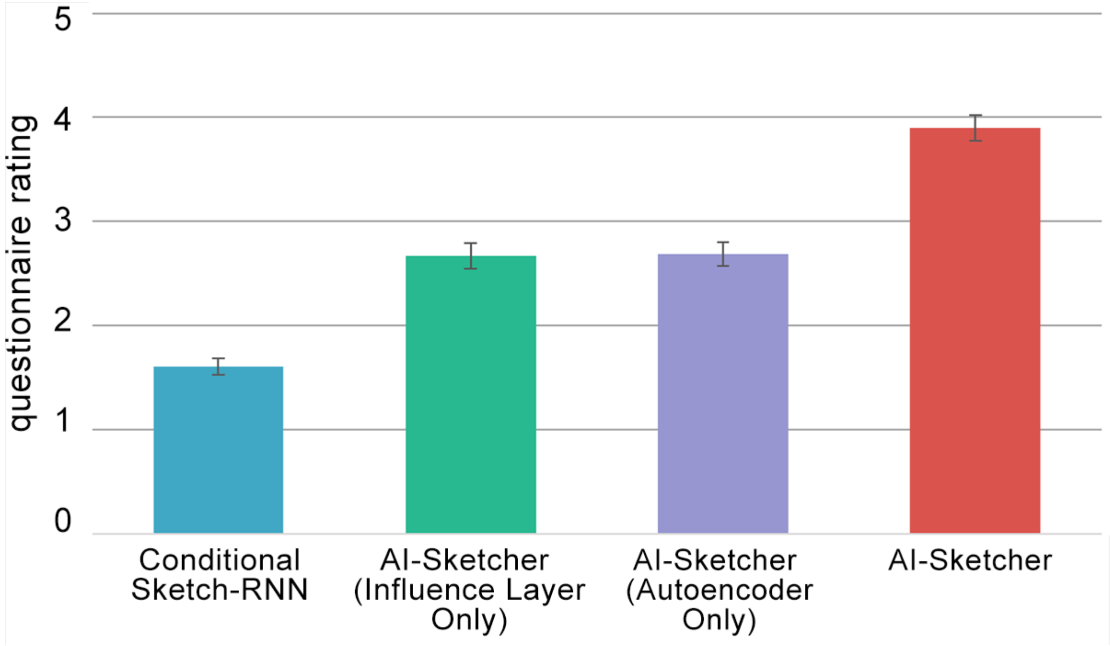
Experiments based on FaceX Dataset

EVALUATION - 1st Experiment

Drawing quality

A within-subject user study

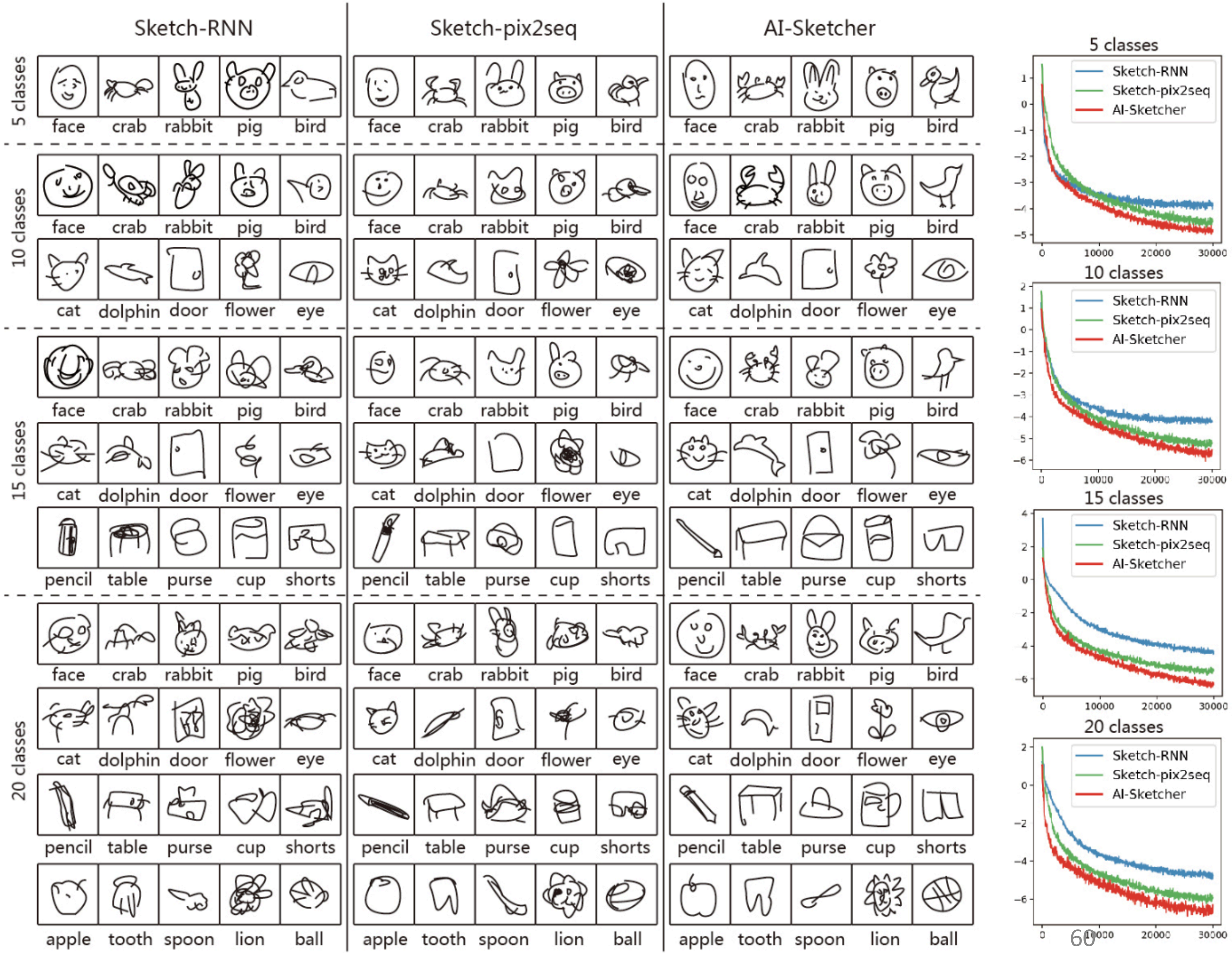
- 20 participants (10 females)
- The repeated measures one way ANOVA analysis showed that the generation quality of AI-Sketcher had an average rating of **3.9** and was significantly better than that of the baseline models (with all $p<.01$).



EVALUATION - 2nd Experiment

Generating sketches from multiple classes

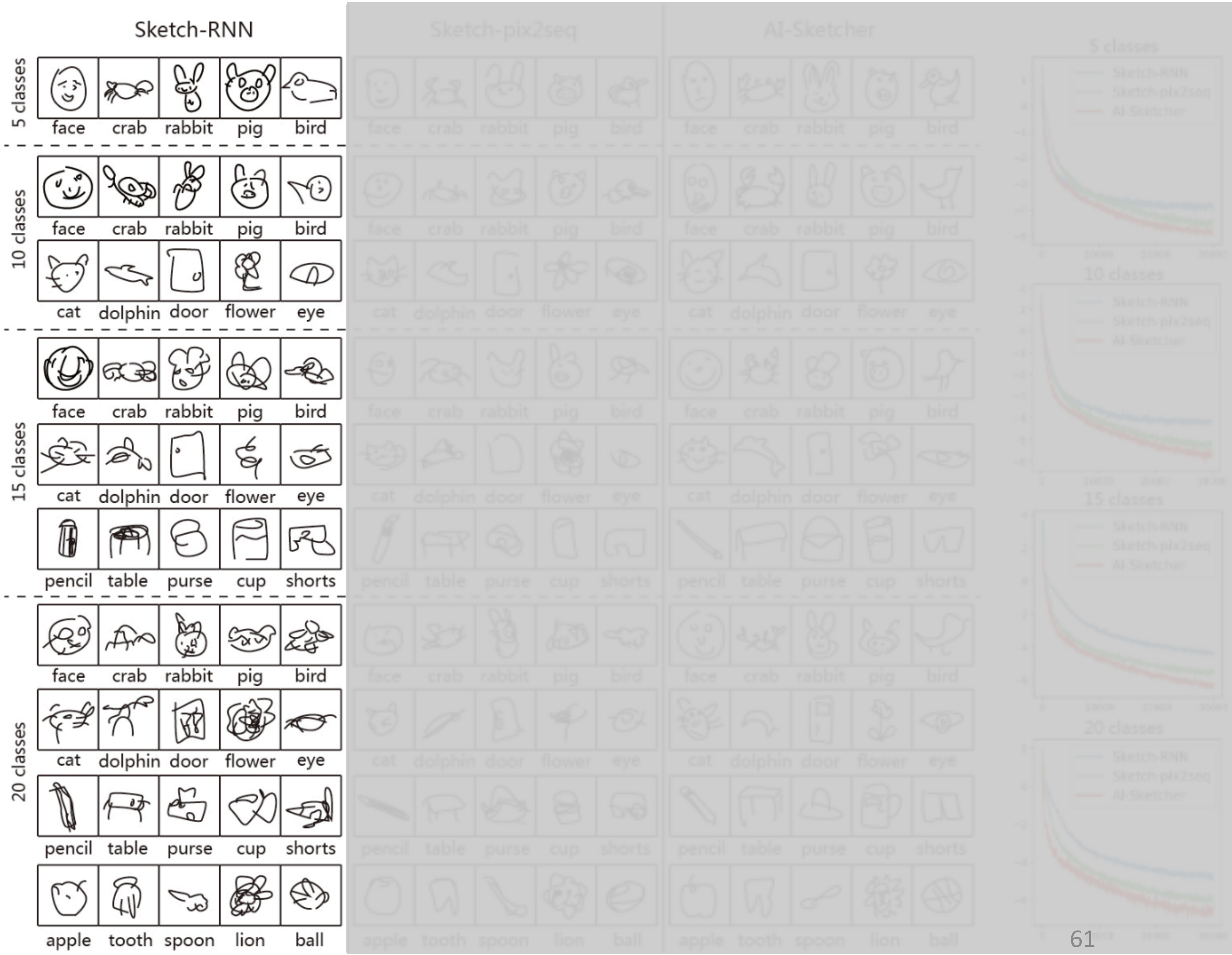
Experiments
based on QuickDraw Dataset



EVALUATION - 2nd Experiment

Generating sketches from
multiple classes

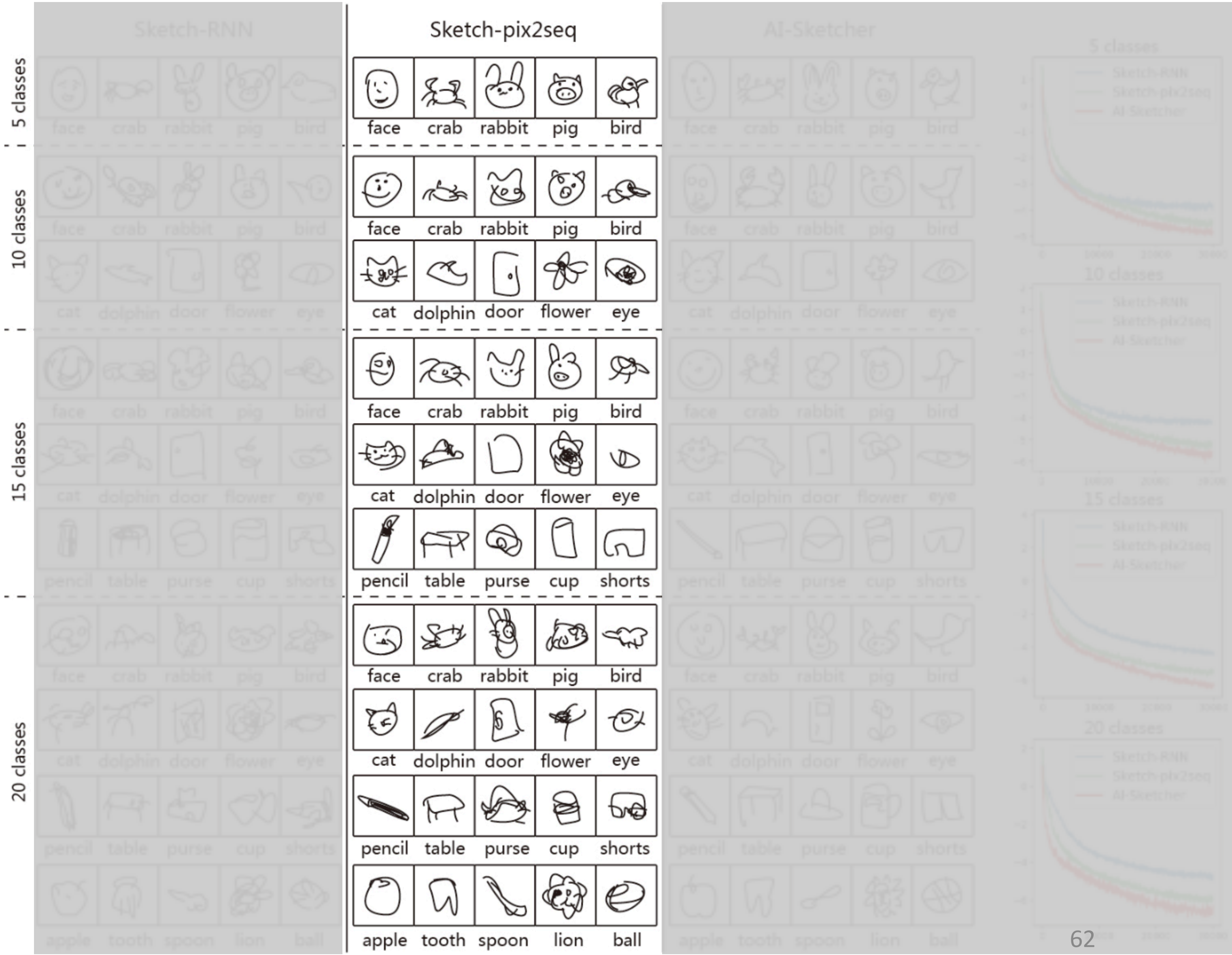
Experiments
based on QuickDraw Dataset



EVALUATION - 2nd Experiment

Generating sketches from multiple classes

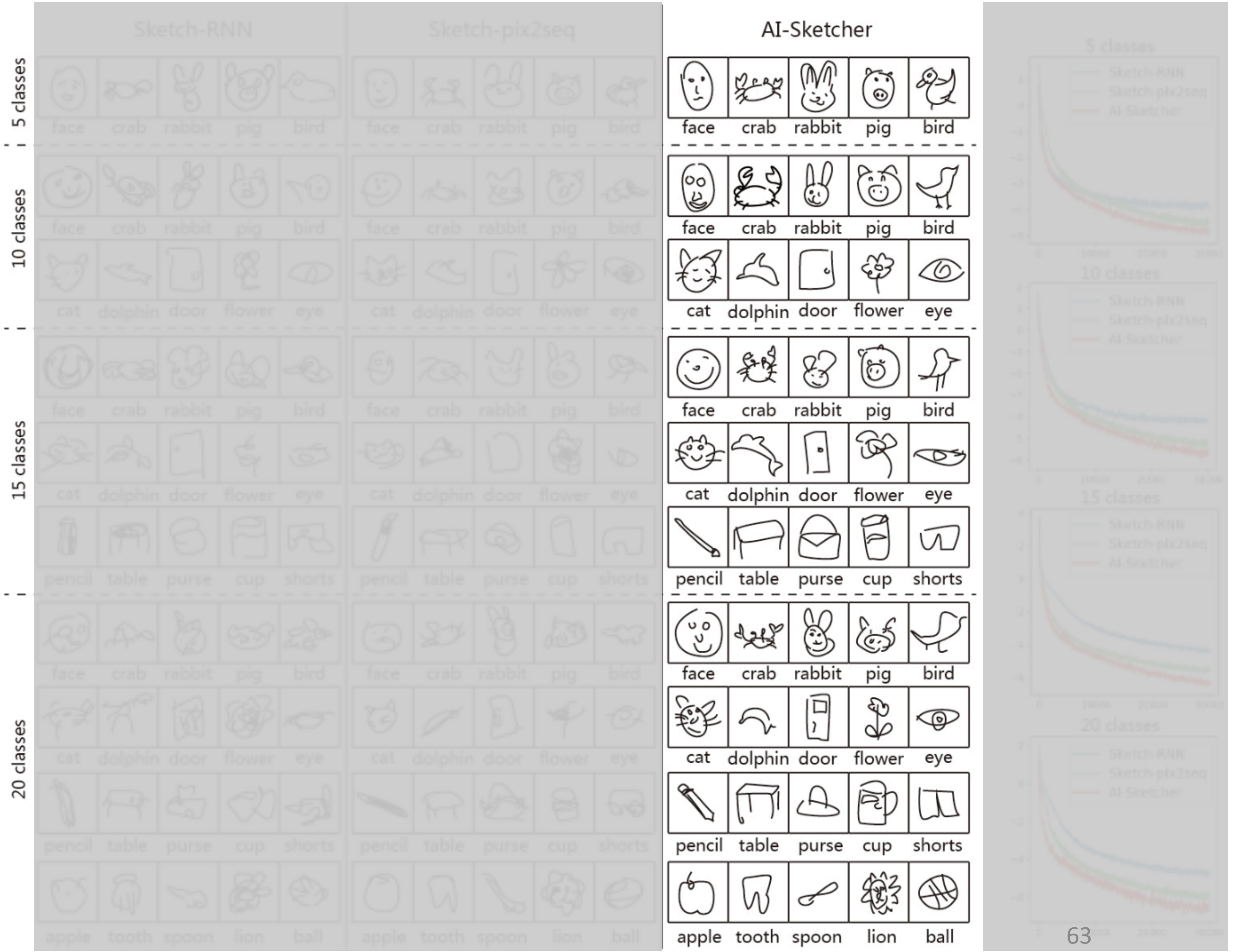
Experiments
based on QuickDraw Dataset



EVALUATION - 2nd Experiment

Generating sketches from
multiple classes

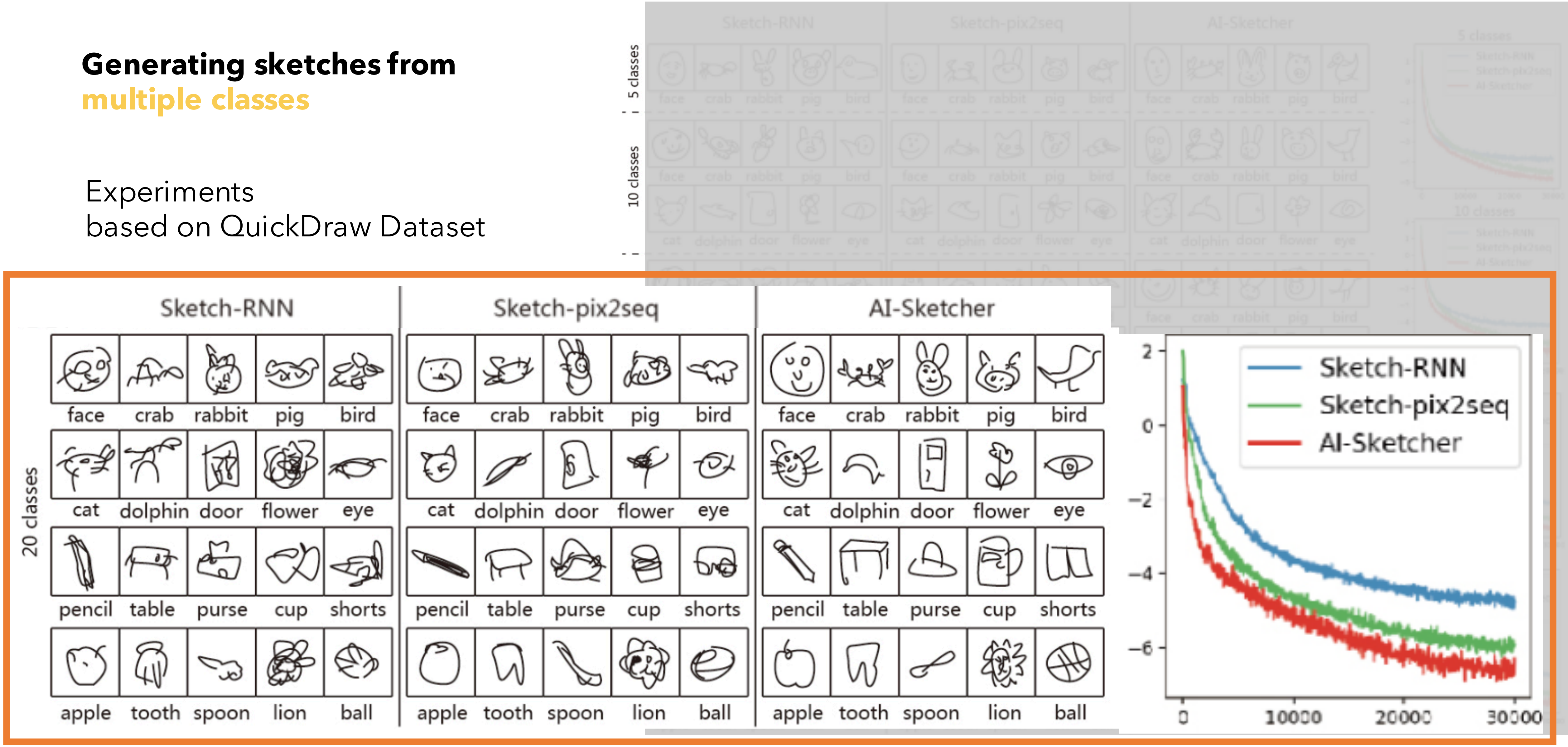
Experiments
based on QuickDraw Dataset



EVALUATION - 2nd Experiment

Generating sketches from multiple classes

Experiments
based on QuickDraw Dataset



EVALUATION - 3rd Experiment

Generation Diversity

In each set, the pairwised distances between sketches were calculated based on the **perceptual hash**.

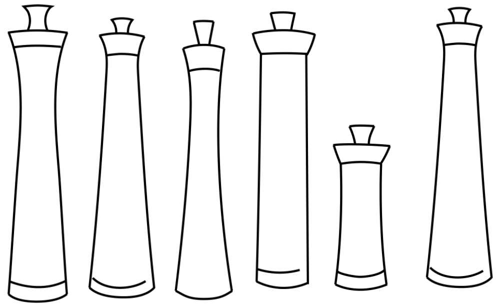
The unpaired t-test showed that AI-Sketcher and Sketch-RNN had no significant difference.

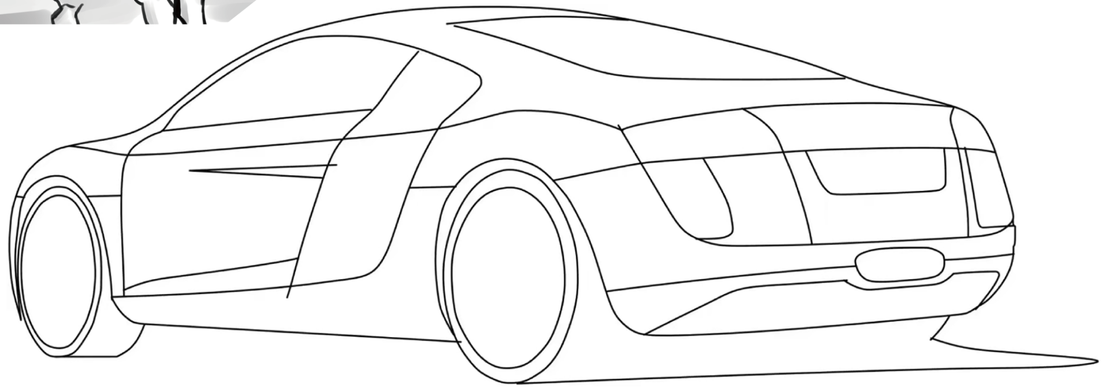
Input										
Model	●	●	●	●	●	●	●	●	●	●
Mean	30.66	30.97	29.76	29.87	28.98	28.82	29.23	29.45	29.66	30.00
SD	5.18	5.54	5.79	5.84	5.93	6.20	6.08	5.87	5.37	5.83
t(198)	-1.42		-0.53		0.65		-0.92		-1.53	
p	0.16 >.05		0.56 > .05		0.51 > .05		0.35 > .05		0.13 > .05	

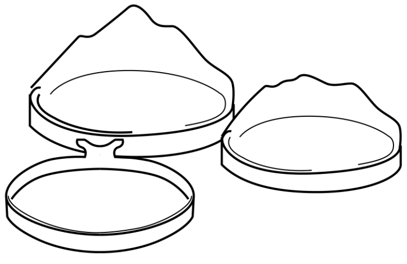
● AI-Sketcher ● Sketch-RNN

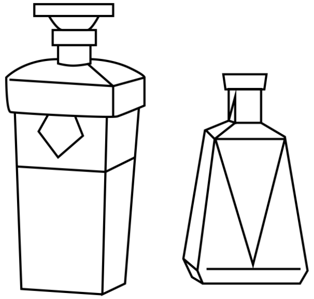
Experiments based on QuickDraw Dataset

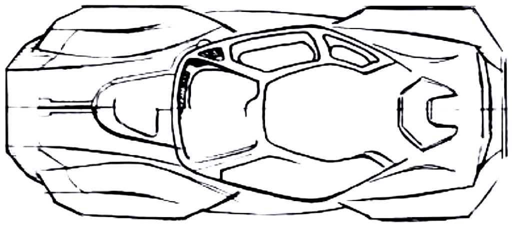
POTENTIAL APPLICATION

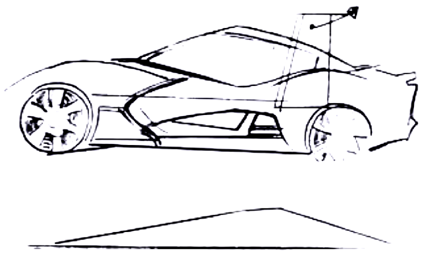




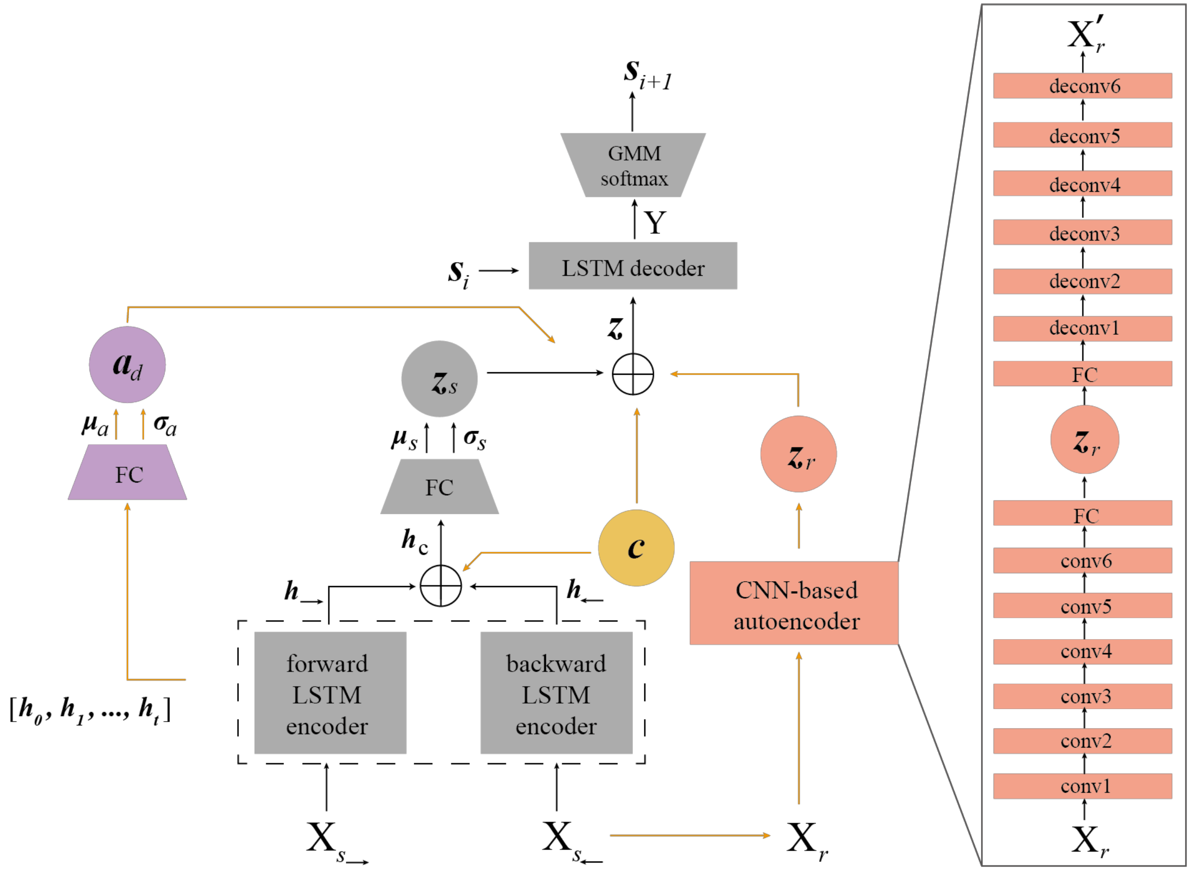








CONCLUSION



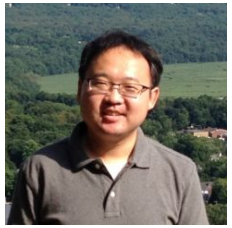
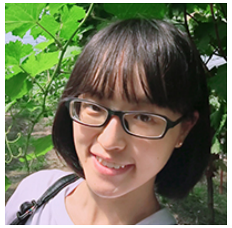
We introduced **AI-Sketcher,** a hybrid deep learning model to automatically generate high quality sketch drawings .

Our model improves drawing quality by

- employing **a CNN-based autoencoder** to capture the positional information.
- introducing **an influence layer** to more precisely guide the generation of each stroke.
- provideding **a conditional vector** to support multi-class sketch generation.

Thank You

Nan Cao, **Xin Yan,** Yang Shi, Chaoran Chen

Tongji University

Intelligent Big Data Visualization Lab

FaceX

ABOUT US ACKNOWLEDGEMENT

A Dataset Containing 5,240,088 Hand-Drawing Sketches

The dataset contains over 5 million labeled facial sketches categorized by genders (male, female), viewing angles (frontal, mid-profile left view), emotions (neutral, happy, sad, angry, fearful, surprised, disgusted), and artistic styles (realistic, cartoon, abstract styles).

DOWNLOAD CONTRIBUTE

SVG format: 73 NPZ format: 64

https://facex.idvxlab.com/