RoboCoDraw: Robotic Avatar Drawing with GAN-based Style Transfer and Time-efficient Path Optimization

Tianying Wang[1,*], Wei Qi Toh[1,*], Hao Zhang[1,*], Xiuchao Sui[1], Shaohua Li[1,2], Yong Liu[1,2], Wei Jing[1,3]

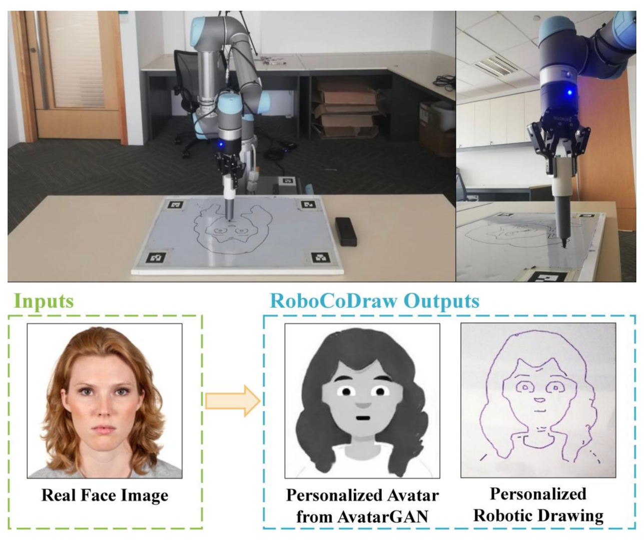
[1]Artificial Intelligence Initiative, A*STAR; [2]Institute of High Performance Computing, A*STAR; [3]Institute of Information Research, A*STAR; Corresponding Email: 21wjing@gmail.com

Agency for Science, Technology and Research SINGAPORE

Introduction

Robotic drawing as a human-robot interaction task– Robotic drawing is interactive, fascinating and entertaining to the public, with applications in a wide range of scenarios (e.g. early childhood education, psychological therapy, and social entertainment). Of these, drawing human faces is one of the most engaging tasks.

Robotic art creation– Although somewhat impressive, the generation of realistic drawings has limited level of amusement for users. To increase entertainment and engagement, we propose *RoboCoDraw*, a real-time robotic drawing system that converts input face images to cartoon avatars as a form of robotic art creation.



Main contributions:

- a two-stream CycleGAN model named AvatarGAN, that maps human faces to cartoon avatars while preserving facial features
- a modularized, interactive robotic drawing system that performs faithful style translation and time-efficient face drawing
- a path optimization formulation for the robotic drawing problem and an RKGA-based optimization algorithm with two-level local improvement heuristics

The RoboCoDraw System

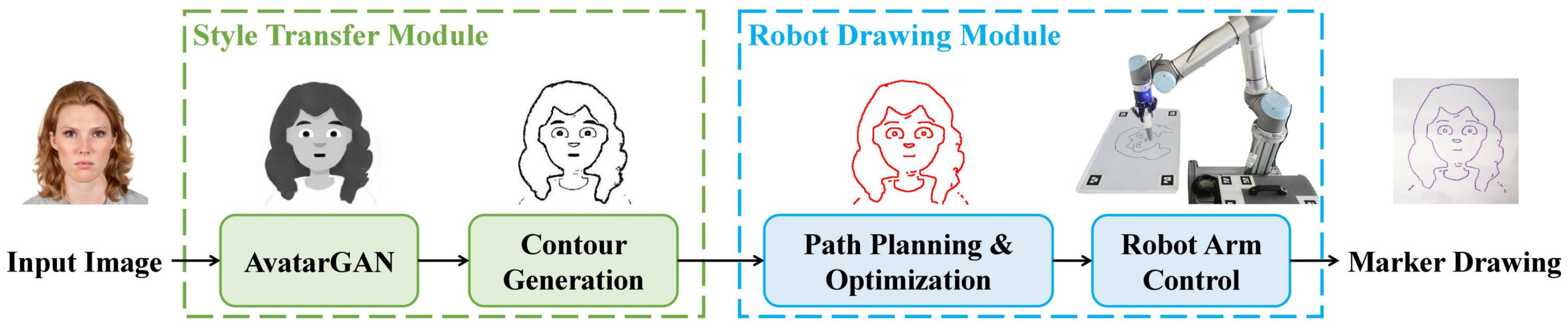


Fig. 1 Pipeline of the RoboCoDraw System, consisting of two main modules: Style Transfer Module and Robot Drawing Module.

Style Transfer Module

- ***AvatarGAN*** performs photo-to-avatar translation
- ***Contour Generation*** constructs coherent contours from generated avatars with Flow-based Difference-of-Gaussian (FDoG) filtering

AvatarGAN: Two-stream CycleGAN system

- Learns mapping G_{XY} and G_{YX} between real faces (X) and cartoon-style avatars (Y)
- Align distributions of generated avatars with target domains by minimizing adversarial loss
- Preserve consistency in facial features by minimizing cycle consistency loss

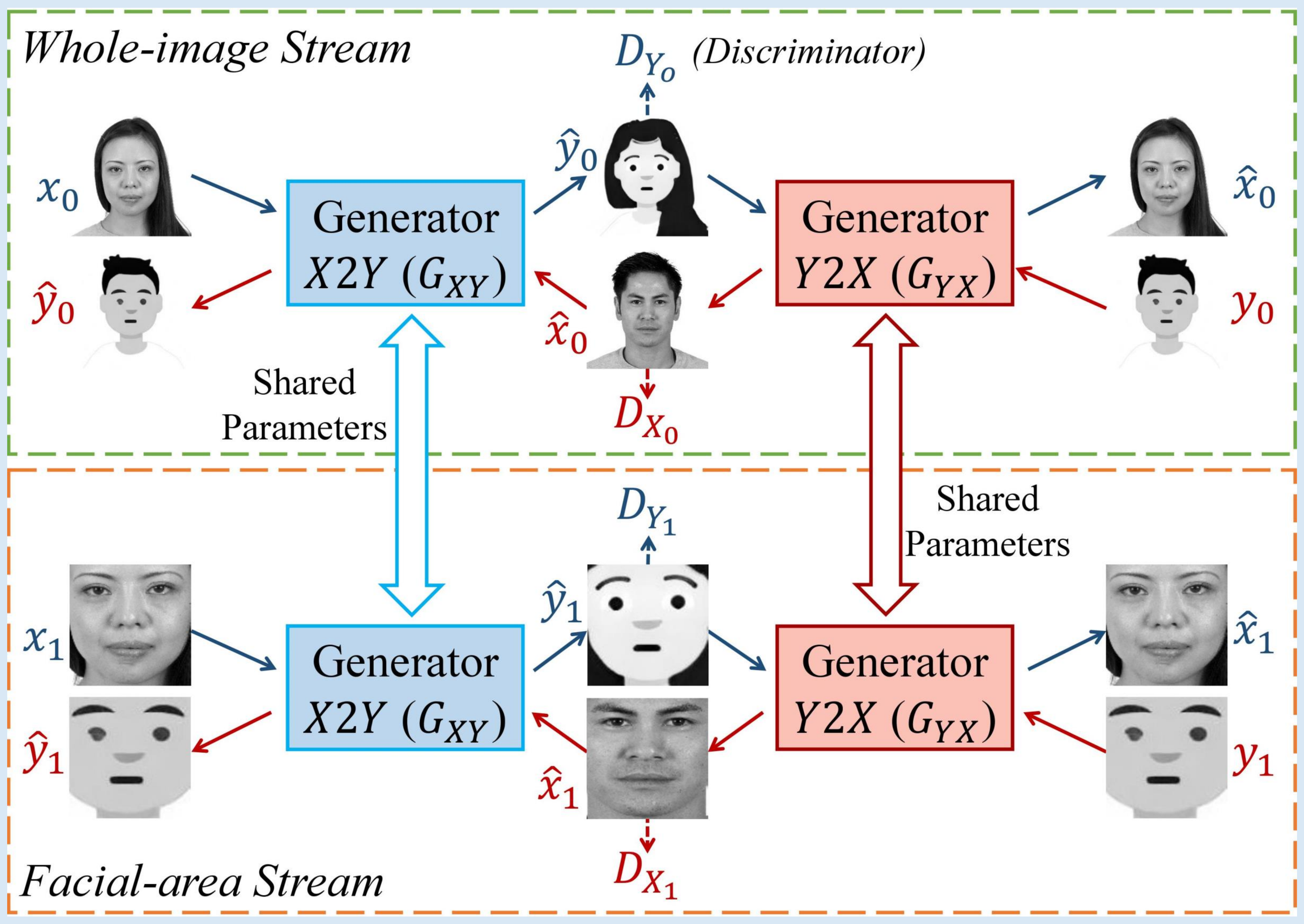


Fig. 2 The structure of *AvatarGAN*, which consists of two translation streams.

Robot Drawing Module

- ***Path Optimization*** by extracting robot drawing path, then formulating drawing task as a Generalized Travelling Salesman Problem (GTSP)
- ***Robot Arm Control*** for interfacing with robot and execution of robotic drawing

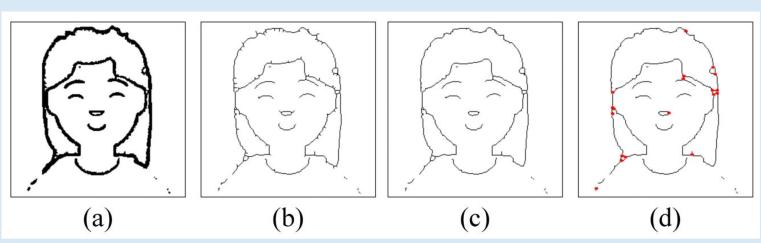


Fig. 3 Obtaining pixel-coordinates from reference image (a). Image is (b) thinned, (c) trimmed, (d) then split at junctions.

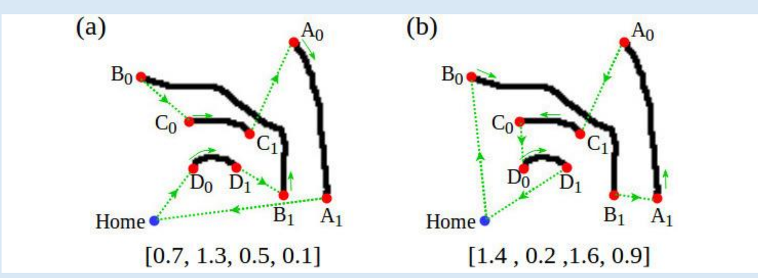


Fig. 4 Drawing path encoded as a list of random keys. Index of each key corresponds to a line segment (decimal part encodes sequence, integer part encodes direction of drawing)

Extraction of Drawing Path:

- Lines are thinned and split at junctions, then traced to obtain pixel-coordinates sequences

Path Optimization GTSP Formulation:

- Encoding and decoding with random keys
- Fitness measure relates to distance traversed in air between drawing of lines

Path optimization method

- RKGA with reproduction, crossover and mutation as genetic operators
- Two-level local improvement involving 2-opt, LK heuristics

*** This project is open-sourced. Code is available at https://github.com/Psyche-mia/Avatar-GAN**

Experiments and Results

Avatar Generation with AvatarGAN

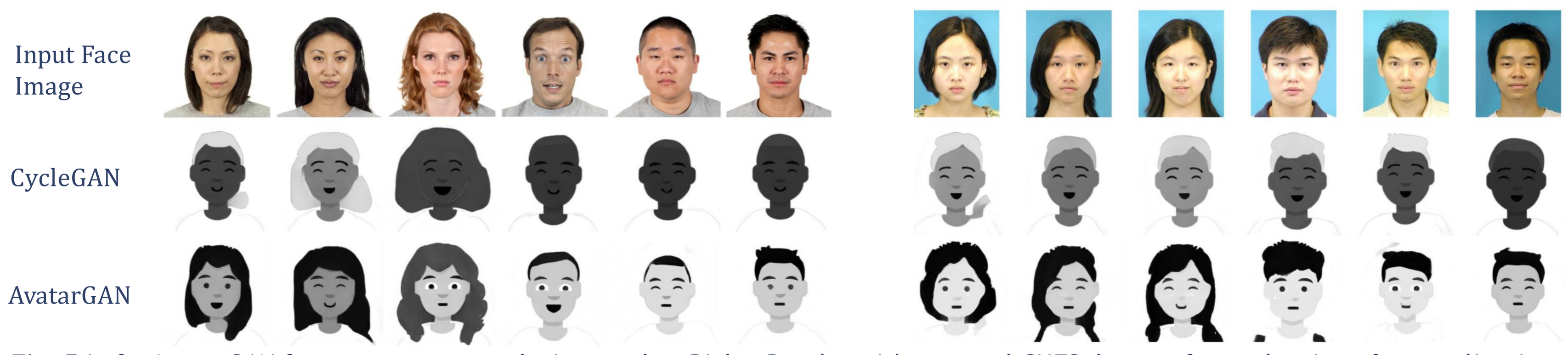


Fig. 5 Left: AvatarGAN face-to-avatar translation results; Right: Results with external CUFS dataset for evaluation of generalization

Evaluation of Generated Avatars:

- Similar facial features to input face images
- More diverse features observed in output
- Creative generation of new facial features not present in training dataset

Evaluation of Generalization:

- Can generalize to CUFS Dataset
- Significantly outperforms CycleGAN on facial-area translation (Table 1)

User Study:

- Matched 10 random faces to corresponding AvatarGAN output with mean accuracy of 98%

AvatarGAN Training Datasets:

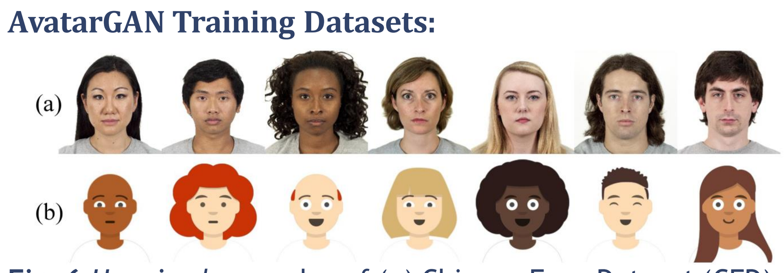


Fig. 6 *Unpaired* examples of (a) Chicago Face Dataset (CFD) images and (b) cartoon-style avatar images (generated from the *Avataaars* library) used to train AvatarGAN

Table 1 Mean cycle consistency loss of CycleGAN and AvatarGAN on the test datasets

Method	Mean cycle consistency loss $\mathcal{L}_{cycle}$			
	Face	Avatar	Real facial area	Avatar facial area
CycleGAN	**0.0355**	**0.0167**	0.2620	0.3298
AvatarGAN	0.0367	0.0252	**0.0329**	**0.0252**

Experiments on the Robot Drawing Module

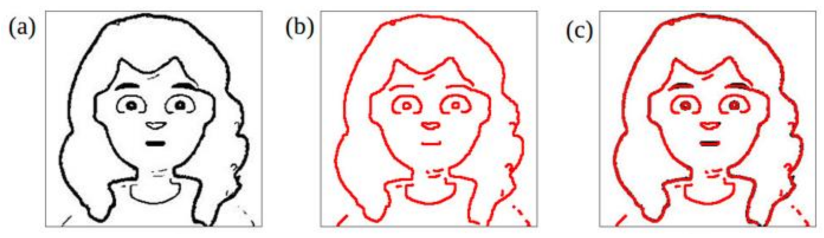


Fig. 7 Pixel-coordinates extraction result. (c) is the overlay of (a) the original avatar contour and (b) a drawing simulated using extracted pixel-coordinates

Table 2 Improvements in path fitness value of various optimization methods over the greedy benchmark (%)

	G1	G2	G3	G4	G5	Avg.
Greedy w/ 2-opt	16.4	20.2	5.5	8.8	9.8	12.1
Greedy w/ 2-opt, LK	16.9	22.5	14.7	11.0	10.3	15.1
RKGA w/ 2-opt	18.0	22.5	19.8	12.2	11.3	16.8
RKGA w/ 2-opt, LK	**19.1**	**22.9**	**20.4**	**12.4**	**11.9**	**17.3**

- Proposed optimization algorithm (RKGA w/ 2-opt, LK) has significant average improvement in path fitness of 17.34% against the greedy search benchmark

Integrated Tests of the RoboCoDraw System

- Conducted 20 integrated trials with our *RoboCoDraw* system implemented on a UR5 robot
- Average time used for the physical drawing was 43.2 s; other computational processes (image preprocessing, AvatarGAN, contour generation, etc.) used 9.9 s

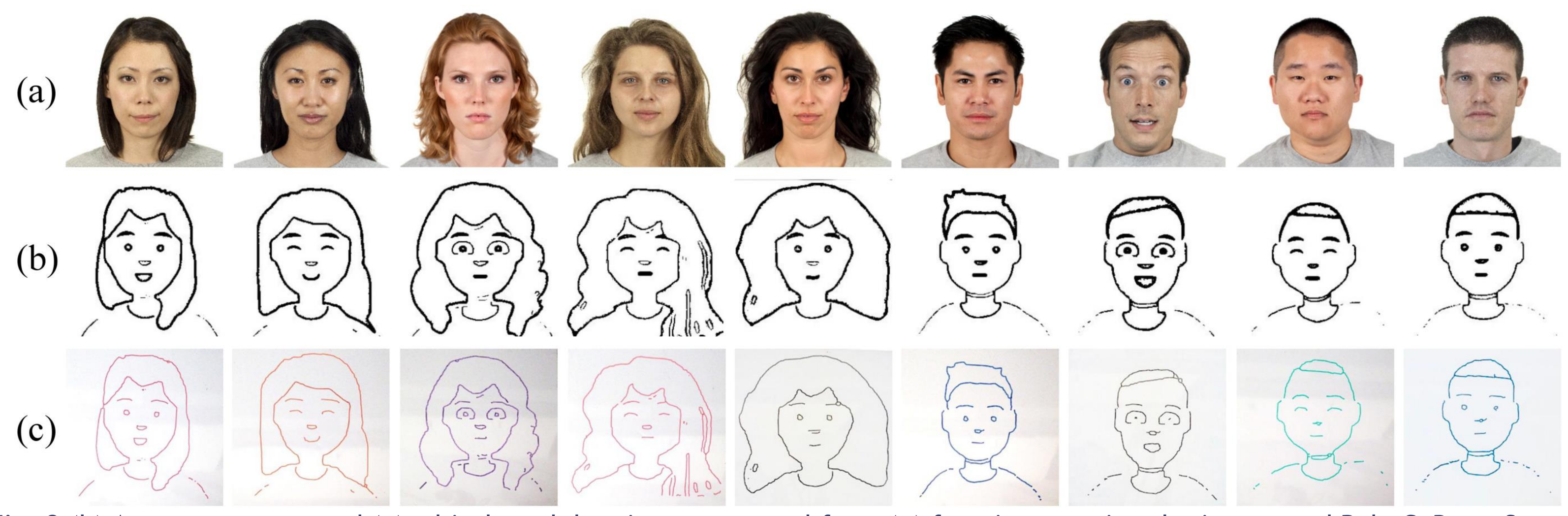


Fig. 8 (b) Avatar contours and (c) whiteboard drawings generated from (a) face image using the integrated RoboCoDraw System

Conclusions

- The proposed two-module *RoboCoDraw* system facilitates the **efficient creation and drawing of personalized avatar sketches** on the robotic arm, given real human face images
  - For style transfer, AvatarGAN generates **more diversified** cartoon avatars with **much better likeness**
  - For path optimization, our method has **17.34% improvement in fitness** compared with the baseline
- Great potential in public amusement and human-robot interactive entertainment applications