

윤곽선의 선분 근사화를 활용한 양팔 화가 로봇의 개발

Development of a Dual-Arm Drawing Robot using Line Segment Approximation of Image Edges

김 정 규¹, 이 상 필², 정 혜 림³, 조 혜 경[†]

Jung-Kyu Kim¹, Sang-Pil Lee², Hye-Lim Jung³, Hye-Kyung Cho[†]

Abstract This paper introduces a dual-arm robot painter system which is capable of sketching a camera-captured image with short line segments. To express various curved edges in the image by combining line segments, we first apply edge detection algorithm to the entire image, split the edged image into small boxed pieces, and then apply Hough Transformation to each piece so that the edges inside the piece can be approximated with short line segments. To draw the picture within a reasonable time, we designed a simple dual-arm robot system and controlled both arms concurrently according to linear interpolation algorithm. From the experiments, we could verify that simple linear motions can describe various images effectively with a unique brush style.

Keywords: Edge Detection, Hough Transform, Painter Robot

1. 서 론

정보기술의 발전은 컴퓨터 안에서만 존재하던 소프트웨어를 다양한 사물 속으로 끌어 내고 있으며, 이로 인해 산업뿐만 아니라 예술이나 창작분야에서도 정보기술을 결합한 다양하고 창의적인 시도들이 이루어져 보는 이들에게 영감을 주고 있다. 이러한 시도가 개인의 개성표현을 넘어 전문적인 수준으로 발전함에 따라 큰 부가가치를 낼 수 있는 새로운 형태의 콘텐츠에 대한 관심이 높아지고 있으며, 예술과 기술을 융합하는 시도는 당분간 지속될 전망이다^[1]. 본 논문에서는 이러한 시도의 일환으로 영상처리기술과 로봇기술을 응용한 화가 로봇 시스템을 제안하고자

한다.

현재 여러 가지 방법으로 그림을 그리는 화가로봇들이 있다. 첫 번째로, Automatic Graffiti Robot라 불리는 로봇은 두 가닥의 줄에 스프레이가 매달려 그림을 그리는 시스템이다. 스프레이 양쪽에 달린 줄을 잡아 당기고 느슨하게 하면서 스프레이의 위치를 옮겨 그림을 그린다. 이 시스템을 이용하면 아파트 벽과 같은 건축물 외벽에 그림을 그릴 때, 사람이 위험하게 올라가서 그릴 필요가 없게 된다^[2]. 다음으로, 독일 콘스탄츠 대학 연구팀이 개발한 'e-David'가 있다. e-David는 붓을 이용해 물감으로 그림을 그리는 로봇이다. 이 로봇은 5개의 붓과 24가지 색을 선택할 수 있고, 심지어 어느 부분을 진하게 칠해야 하고 얇게 칠해야 하는지를 계산하여 그림을 그리기도 한다. 또한 e-David는 스스로 붓을 물로 깨끗하게 청소하는 기능도 갖고 있다^[4]. 마지막으로, Patrick Tresset에서 Paul이라 불리는 화가 로봇을 개발했다. Paul은 카메라로 얻은 사람 얼굴을 볼펜으로 그림 그린다. 이 로봇은 사람의 얼굴 형태부터 먼저 그린 다음 이목구비를 그리고 머리카락과 음영을 그리는 순으로 사람 얼굴을 완성해 간다^[5].

본 논문에서 제안한 화가 로봇 시스템은 선분을 근간으

Received : Jul. 22. 2014; Reviewed : Jul. 30. 2014; Accepted : Aug. 8. 2014

※ This work was supported by the Industrial Strategic Technology Development Program funded by the Ministry of Trade, Industry, and Energy (MOTIE, Korea)

¹ Dept. of Information & Communications Engineering, Hansung University (num003q@naver.com)

² Dept. of Information & Communications Engineering, Hansung University (splee0114@naver.com)

³ Dept. of Information & Communications Engineering, Hansung University (jhl3646@naver.com)

[†] Corresponding author: Dept. of Information & Communications Engineering, Hansung University, 389, Samsun-Dong 2-Ga, Sungbuk-Gu, Seoul, Korea (hkcho@hansung.ac.kr)

로 그림을 그린다는 점에 있어 Patrick Tresset에서 개발한 Paul과 유사하나, 그림을 그리는 시간을 단축시키기 위해 두 팔을 사용하고 새로운 영상처리 알고리즘을 적용한다. 일반적으로 영상의 윤곽선은 곡선의 형태이나, 이를 수학적으로 모델링하고 곡선을 따라 로봇 팔을 제어하는 것은 상당히 복잡한 과정을 동반한다. 본 논문에서는 복잡한 계산을 피하고 선분들의 조합만으로 영상의 특징을 효과적으로 표현하기 위해, 다양한 영상처리 알고리즘과 좌표 변환 및 역기구학(inverse kinematics)을 이용하여 카메라를 통해 들어온 이미지를 로봇 팔이 종이에 그리도록 설계 · 구현한다.

2. 본 론

2.1 전체 시스템

본 논문에서는 사물이나 사람을 카메라로 촬영한 영상에서 선분을 추출하기 위해 캐니-윤곽선 추출 기법과 허프 변환을 이용하고, 좌표 변환과 역기구학을 이용해 그림을 그리는 양팔 화가 로봇을 구현한다.

전체 시스템의 구성은 Fig. 1과 같다. 카메라를 통해 촬영한 컬러이미지를 흑백화 한다. 흑백화 된 이미지에서 잡음을 제거하기 위해 가우시안 필터를 적용한 후, 캐니 윤곽선 추출 알고리즘을 이용하여 윤곽선을 추출한다. 추출된 윤곽선 이미지를 여러 개의 셀로 분할하고 각 셀 내에서 허프 변환을 적용한다. 허프 변환의 결과로 나온 직선들을 선분화 한다. 그 후 성분이 비슷한 선분들을 군집화시키고, 끝점이 이어지는 선분들은 연결하여 선분의 개수

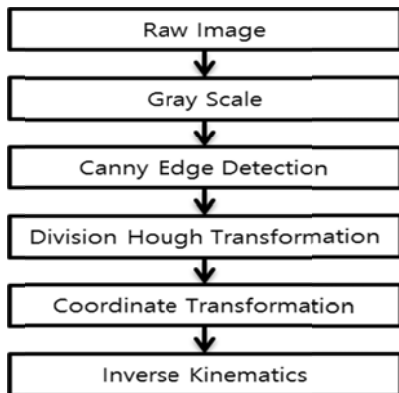


Fig. 1. Major work process

를 줄인다. 이렇게 얻은 선분들을 로봇 팔이 그릴 수 있도록 화면상의 좌표를 로봇이 그릴 도화지상의 좌표로 변환하고, 역기구학을 이용하여 모터의 각도를 계산한다. 그 다음 모터에게 명령을 보내어 로봇이 그림을 그릴 수 있도록 제어한다.

2.2 로봇 기구부 설계

Fig. 2는 로봇의 왼쪽 팔로 3개의 모터로 구성된다. 3개의 모터 중 2개의 모터를 사용하여 펜이 그릴 위치로 이동할 수 있도록 하며, 나머지 모터는 펜을 위아래로 움직여 종이에서 접촉할 수 있게 한다. Fig. 3은 그림이 그려질 종이상의 좌표를 나타낸 것이다.

그림을 그리기 위해서는 3자유도의 팔 제어가 필수적이므로 좌표 변환과 역기구학을 유도한다. 우선 좌표변환은 카메라를 통해 얻어지는 640 * 480 픽셀(pixel) 이미지 좌표를 종이 위 20cm * 15cm의 좌표로 변환하는 역할을 한다. 이것은 로봇 팔을 종이 위에서 정확하게 제어하기 위해 역기구학에서 필요한 로봇 기준 좌표를 구하는 과정이다. 다음으로, 역기구학은 로봇 좌표계 위치를 3자유도 모터의 각도로 계산하는 역할을 한다. 그 계산은 다음과 같다.

역기구학 식은 ${}^0T = {}^0T_1T_1T$ 이고, 0T 는 주어진다.

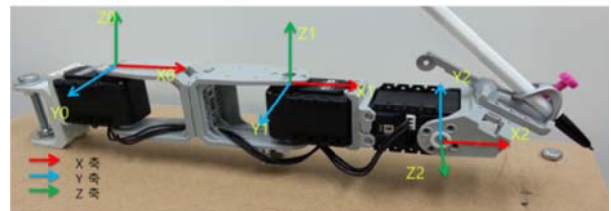


Fig. 2. Coordinate system definition of robot arm

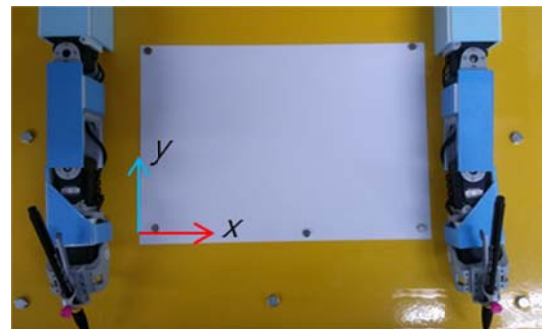


Fig. 3. Coordinate system of the drawing plane

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1c_1 + l_2c_{12} \\ s_{12} & c_{12} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

이 때, 역 기구학의 위치정보를 이용하면

$$x = l_1c_1 + l_2c_{12} \tag{1}$$

$$y = l_1s_1 + l_2s_{12} \tag{2}$$

가 되며, 식 (1), (2)의 양변을 각각 제곱하여 더하고, $\sin^2\theta_2 + \cos^2\theta_2 = 1$ 를 이용하면, θ_2 를 식 (3)과 같이 계산할 수 있다.

$$\begin{aligned} \theta_2 &= \tan^{-1}(\sin \theta_2 / \cos \theta_2) \\ &= \tan^{-1}(\pm \sqrt{1 - \cos^2 \theta_2} / \cos \theta_2) \end{aligned} \tag{3}$$

이 때, 양과 음의 부호에 따라 값이 두 개가 존재하게 되는데 본 논문에서는 양의 부호를 이용하여 자세를 결정한다.

다음으로 θ_1 을 구하면,

$$\begin{aligned} x &= l_1c_1 + l_2c_{12} \\ &= l_1c_1 + l_2(c_1c_2 - s_1s_2) \\ &= (l_1 + l_2c_2)c_1 - (l_2s_2)s_1 \end{aligned} \tag{4}$$

$$\begin{aligned} y &= l_1s_1 + l_2s_{12} \\ &= l_1s_1 + l_2(c_1s_2 + s_1c_2) \\ &= (l_1 + l_2c_2)s_1 - (l_2s_2)c_1 \end{aligned} \tag{5}$$

가 되며, 식 (4), (5)를 행렬로 정리하면

$$\begin{bmatrix} l_1 + l_2c_2 & l_2s_2 \\ l_2s_2 & l_1 + l_2c_2 \end{bmatrix} \begin{bmatrix} c_1 \\ s_1 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

가 되므로, 크레이머 공식에 의해

$$\begin{aligned} c_1 &= \{(l_1 + l_2s_2)x + l_2s_2y\} / \{(l_1 + l_2c_2)^2 + (l_2s_2)^2\}, \\ s_1 &= \{(l_1 + l_2s_2)y - l_2s_2x\} / \{(l_1 + l_2c_2)^2 + (l_2s_2)^2\} \end{aligned}$$

에서 최종적으로 θ_1 을 식 (6)과 같이 구할 수 있다⁶⁾.

$$\theta_1 = \tan^{-1}(s_1/c_1) \tag{6}$$

이렇게 구한 θ_1, θ_2 값을 모터에 각각 설정하여 목표위치까지 이동시킬 수 있다. 반대쪽 팔의 모터의 각도 값 역시 위와 동일한 과정을 거쳐 계산할 수 있고, 그 값들을 미러링(mirroring)하여 반대쪽 로봇 팔과 대칭적인 각도에서 모터가 동작하도록 모터들에게 명령어를 내린다.

2.3 제어 소프트웨어 설계

소프트웨어의 아키텍처(architecture)는 Fig. 4와 같이 카메라 부, 컴퓨터 부 및 로봇 시스템 부로 구성되어 있다. 전체적인 흐름은, 카메라로 받아들인 영상 데이터를 PC에서 가공하여 윤곽선을 선분의 조합으로 변환하고, 이 선분 식을 이용하여 로봇 팔을 제어하는 것이다.

이 과정을 자세히 설명하면 다음과 같다. 먼저 카메라를 통해 대상의 이미지를 획득한 후, 그 이미지를 흑백화(gray scale)한다. 이후 캐니 윤곽선 추출 알고리즘(canny edge detection algorithm)을 적용하여 윤곽선을 추출한 후, 허프 변환(Hough transform)을 응용하여 윤곽선을 선분의 조합으로 근사화한다.

여기서 캐니 윤곽선 추출 알고리즘은 다음과 같은 과정으로 이루어진다. 가장 먼저 가우시안 마스크(Gaussian mask)를 이용하여 스무딩(smoothing)을 수행을 한 후, 소벨 마스크(Sobel mask)를 x방향, y방향에 대해 적용하여 x방향 미분 값과 y방향 미분 값을 구하여 기울기 크기와 방향각을 구한다. 다음으로 기울기 크기가 방향각 방향의 주변

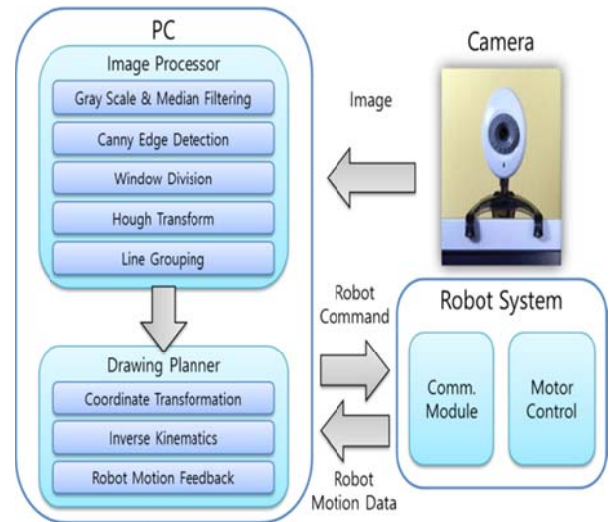


Fig. 4. Architecture of the overall system

값보다 작을 경우, 윤곽선 대상에서 제거한다. 마지막으로 상한과 하한의 문턱치(threshold)를 두어 상한을 초과하는 경우 무조건 윤곽선으로 삼고, 하한의 미만일 경우에는 윤곽선에서 제외시키며, 그 사이는 주변에 자신 이상인 것이 있을 때 윤곽선으로 삼는 과정을 거친다⁷⁸⁾.

허프 변환은 이미지를 구성하는 점들을 이용하여 주요 직선 성분을 찾는 과정이다. 직각 좌표계 상의 임의의 점 (x, y) 를 지나는 모든 직선들은 원점 $(0, 0)$ 에서부터의 직선거리와 방향각을 파라미터(parameter)로 재구성한 공간에서는 곡선으로 표현되는데, 이 공간에서 특정한 점을 지나는 곡선이 많다는 것은 원래 영상에 해당 파라미터로 표현되는 직선이 존재한다는 것을 의미하게 된다. 따라서 변환된 공간에서 임계치를 넘는 점들을 선별하면 직선 식을 구할 수 있다⁹⁾.

본 논문에서는 Fig. 5와 같이 전체 영상을 작은 셀(cell)로 분할하여 허프 변환을 적용하는데, 전체 영상을 대상으로 그대로 허프 변환을 적용하면 연산의 속도가 매우 느릴 뿐만 아니라, 임계치를 설정하는 데에 따라 실제로는 바로 연결되지 않은 긴 직선 성분이 많이 나타나 짧은 곡선이나 섬세한 선들의 정보를 얻을 수 없기 때문이다. 이미지를 분할하면 분할된 셀(cell) 내의 점만으로 검사하기 때문에 연산량이 많이 줄어들며, 실제로 연결된 직선 성분들을 구할 수 있게 된다.

이렇게 직선을 구한 후에는 시작점과 끝점을 찾아 선분으로 변환하는 작업이 필요하다. 만일 직선을 그대로 그리면 마치 낙서를 한 듯한 직선들이 그려져 윤곽선의 형태를 알아볼 수 없게 되기 때문이다. Fig. 6은 직선과 선분이 어떻게 다른 영상을 만들어 내는지 보여준다. 처음의 점 영상과 그 점 영상을 허프 변환하여 직선으로 구성한 영상, 그리고 연구에서 제안한 알고리즘으로 구한 선분 영상

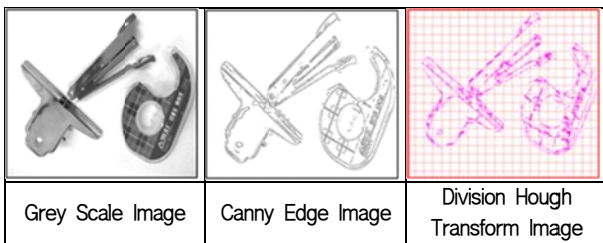


Fig. 5. Process of DHF(Division Hough transform)

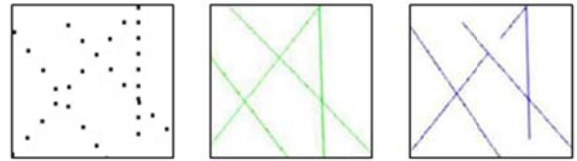


Fig. 6. Dot image, line image and line segment image

을 통해 윤곽선의 근사화 과정을 알 수 있다.

선분들의 추출 과정이 완료되면 다음으로 분할된 셀들의 선분들을 합치는 과정이 진행된다. 셀들의 선분들을 합치는 과정은 직선 근집화를 응용하였다. 직선의 근집화란 직선들끼리의 떨어진 거리, 차이 나는 각도를 계산하고, 평행한지 교점이 있는 지를 검사하며 교점이 없다면 최단 거리를 계산하여 직선들끼리의 연관성을 찾아내고 연관성이 있다면 최적의 직선으로 합치는 과정을 반복하는 것이다¹⁰⁾. 본 논문에서는 직선의 근집화를 응용하여 Fig. 7과 같이 선분을 근집화하는 과정을 진행한다.

우선 하나의 기준 선분을 결정하고 비교할 직선을 결정한다. 결정된 두 선분을 그림으로 표현하면 Fig. 8과 같다. 기준 선분과 비교 선분을 정하면, 두 선분 각도의 차이를 식 (7)과 같이 구할 수 있다.

$$\Delta\theta = \tan^{-1}\left(\frac{y_4 - y_3}{x_4 - x_3}\right) - \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right). \quad (7)$$

식 (7)이 임계값 보다 크면 다른 선분과 비교하며, 그렇지 않으면 기준 선분의 끝점과 비교 선분의 시작점과의 거리 식(8)을 계산한다.

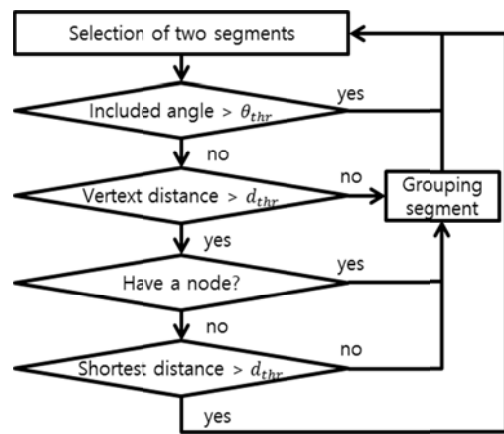


Fig. 7. Algorithm for grouping line segments

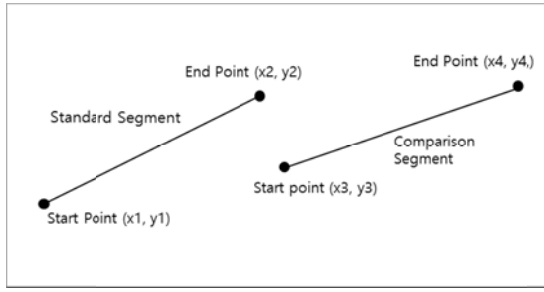


Fig. 8. Example of segments

$$d = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} \tag{8}$$

식 (8)의 거리가 가깝다면 기준 선분의 시작점 (x_1, y_1) 과 비교 선분의 끝 점 (x_4, y_4) 을 잇는 새로운 선분을 만든다. 만일 두 선분 사이의 거리가 멀다면 두 선분의 교점을 검사한다. 교점의 좌표는 다음과 같이 구한다. 선분을 포함한 직선 식은 각각

$$\begin{aligned} r_1 &= (y_2 - y_1), & r_2 &= (y_4 - y_3), \\ s_1 &= -(x_2 - x_1), & s_2 &= -(x_4 - x_3), \\ t_1 &= x_2y_1 - x_1y_2, & t_2 &= x_4y_3 - x_3y_4 \end{aligned}$$

이므로 $r_1x + s_1y + t_1 = 0$, $r_2x + s_2y + t_2 = 0$ 일 때, 교점의 좌표는 식 (9)와 같이 된다.

$$\left(\frac{s_1t_2 - s_2t_1}{s_2r_1 - s_1r_2}, \frac{t_1r_2 - t_2r_1}{s_2r_1 - s_1r_2} \right) \tag{9}$$

하지만 이 교점의 좌표가 직선 식에서 구한 식이기 때문에 선분 상에서 교점이 있는 없는 지를 아직 알 수 없다. 교점의 좌표를 (x_c, y_c) 라 한다면 x_c 는 x_1, x_2 의 사이와 x_3, x_4 의 사이에 존재하고 y_c 는 y_1, y_2 의 사이와 y_3, y_4 의 사이에 존재해야 두 선분끼리 교점이 존재한다. 이렇게 교점을 검출이 되면 두 선분의 각각의 시작점, 끝점 중 가장 긴 선분을 결정하여 새로운 선분을 만든다. 두 선분이 교점이 없다면 평행이거나 약간만 연장을 하면 교점을 가진다. 그러므로 두 선분 사이의 최소 거리를 계산하여 임계값 보다 작다면 두 선분을 군집화한다. 그 과정은 교점이 발생된 선분들을 연결하는 것과 동일하다. 이렇게 기준 선분과 비교 선분을 바꿔가면서 군집되는 선분이 없을 때까지 Fig. 7의 과정을 반복을 한다. 군집되는 선분이 없으면

비로소 군집 합병 과정이 끝나게 된다.

Fig. 9는 캐니 윤곽선 이미지와 군집화 과정을 반복으로 얻게 된 선분들의 이미지를 보인다. 이처럼 곡선을 선분들의 조합으로 표현하는 것이 가능하게 되며, 이를 그림으로 옮기면 마치 연필로 스케치한 듯한 효과가 생성된다고 할 수 있다.

이러한 과정을 통해 선별된 선분들을 시작점과 끝점을 활용하여 그리기 위해서는 직선 보간 (linear interpolation)이 필요하다. 직선 보간은 시작점과 끝점 사이에서 중간 점들을 구하는데, 다음과 같은 식 (10)으로 표현된다.

$$\vec{x}_t = \vec{x}_s + k(\vec{x}_d - \vec{x}_s) \tag{10}$$

\vec{x}_t 는 중간 점 벡터, \vec{x}_s 는 시작점 벡터, \vec{x}_d 는 끝점의 벡터로 k 를 0부터 1까지 증가시킨다. k 의 값은 선분의 길이에 따라 보간의 섬세함이 비례해야 하기 때문에 길수록 서서히 증가 해야 한다. k 는 식 (11)와 같다.

$$k = \frac{n}{|\vec{x}_d - \vec{x}_s|} \tag{11}$$

여기서 n 은 $0 \sim |\vec{x}_d - \vec{x}_s|$ 사이 값에서 일정하게 증가한다. 직선이 길면 n 의 값이 $0 \sim |\vec{x}_d - \vec{x}_s|$ 까지 증가하는 데 걸리는 횟수가 많이 필요하고, 직선의 길이가 짧다면 n 의 증가 횟수가 적게 필요하다. 이렇게 k 의 값은 0부터 1까지 증가하게 되며, 시작점에서 시작하여 중간 점들을 하나씩 샘플링을 하면서 끝점까지 직선 보간이 이루어지게 된다. 직선 보간에서 중간 점의 샘플링이 이뤄질 때마다 그 중간 점들을 역기구학을 이용하여 모터들의 각도 값을 계산

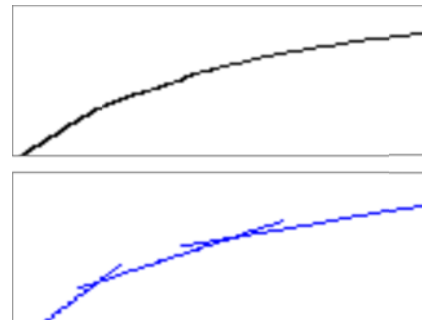


Fig. 9. Detected canny edges and grouped segments

하고 명령을 주어 그 점들을 이어 선분의 그림을 그릴 수 있게 한다.

그림을 그리는 속도는 중간 점들의 샘플링 주파수와 샘플링 픽셀에 결정된다. 샘플링 주파수는 5Hz이며, 매 샘플링 마다 일정하게 5픽셀 거리를 움직인다. 640 * 480 픽셀을 20cm * 15cm로 출력하면 초당 0.78cm의 선분을 그린다. 여기서 5픽셀을 정한 이유를 설명하자면, 로봇 시스템을 구현에 이용된 모터의 최소 제어 각도는 0.29°이다. 최소 표현 가능 동작은 0.125cm이다. 5픽셀은 0.156cm이므로 최소 움직임 보다 약간 크므로 5픽셀로 정했다. 아래 Fig. 10에 보인 초상화의 경우, 허프 변환으로 발생하는 선분의 수는 대략 200~300개이며 선분의 평균 길이는 20~30픽셀로 그림을 그리는 시간은 10~20분 정도 소요된다.

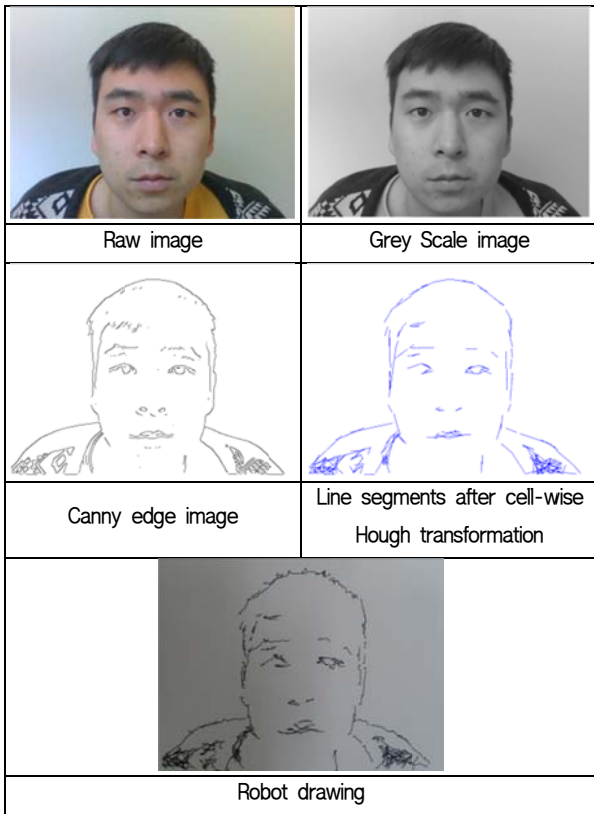


Fig. 10. Images generated during the process

3. 결 론

본 연구에서는 카메라를 통해 얻은 영상을 디지털 영상

처리 기법을 적용하여 단순하게 변형하고 그 영상을 종이에 그려주는 로봇 시스템을 고안하였다. 카메라 영상의 단순화를 위해 캐니 윤곽선 추출 알고리즘과 허프 변환을 사용하였고, 단순화된 영상 정보에 따라 로봇 팔을 제어하기 위하여 역기구학을 유도하였다. 최종적으로 구현된 로봇 화가 시스템은 고유한 스타일을 이용하여 원본 영상을 성공적으로 스케치하였다.

펜으로 그린 그림이 마치 화가처럼 고유한 스타일을 갖는 이유는 다양한 원인의 오차가 작용한 결과이다. 5픽셀 기본 단위 움직임으로 인한 정밀도 오차, 모터 제어 오차, 균일하지 않은 종이 마찰 및 기구적인 펜의 흔들림으로 인해 실제 펜으로 그린 그림에서는 각 선분의 시작과 끝에 고유한 터치가 남았다. 향후 전문적으로 설계된 기구를 활용하면 이미지를 더 정확하게 복원할 수 있을 것으로 예상되며, 나아가 초상화의 경우, 표정인식이나 감정인식^[11,12]을 결합하면 지능적이고 차별화된 문화 콘텐츠로 개선할 수 있을 것으로 기대된다.

References

- [1] SangSang BalJeonSo, "[IT/Application] IT Fusion Contents, Meet the new strategy", <http://koreancontent.kr/1572>, July, 2013.
- [2] MK News, "Moon Ye-Wie · Korea Culture Technology Institute conducts fusion between Art and Technology", September, 2013.
- [3] MediaTrend, "Wonderful painting robot drawing pictures", August, 2006.
- [4] Robot new, "Robot, debuts a painter", <http://blog.naver.com/kips1214/40193224831>, July, 2013.
- [5] wired.co.uk, "Patrick Tresset's robots draw faces and doodle when bored", June 2011.
- [6] S. B. Moon et al., "Intelligent Robotics", Scitech, 2010.
- [7] M. Y. Jung et al., "C# Digital Image Processing", MireaCom, 2005
- [8] S. H. Jung, M. H. Lee, "Image Processing Comprehension and Utilization using C", YeongHan publisher, 2003.
- [9] D. J. Kang, J. E. Ha, "Digital Image Processing with Visual C++", InfinityBooks, 2010
- [10] J. Y. Kim et al, "The Iterative Straight Line Clustering using Blocked Hough Transform", Korea Institute of

Information Scientists and Engineers spring workshop presentation file Vol. 23. No. 1, pp2-3, 1997

- [11] D. Cho et al., "Facial Feature Tracking Using Adaptive Particle Filter and Active Appearance Model", Journal of Korea Robotics Society, Vol. 8, No. 2, 2013.
- [12] B. Ahn et al., "Robust Real-time Tracking of Facial Features with Application to Emotion Recognition", Journal of Korea Robotics Society, Vol. 8, No. 4, 2013.



정혜림

2011~현재 한성대학교 정보통신공학과 학사과정
관심분야 : Image Processing



김정규

2011~현재 한성대학교 정보통신공학과 학사과정
관심분야 : Image Processing, Humanoid Robots



조혜경

1987 서울대학교 제어계측공학과 (공학사)
1989 서울대학교 제어계측공학과 (공학석사)
1994 서울대학교 제어계측공학과 (공학박사)

1996~현재 한성대학교 정보통신공학과 교수
관심분야 : Robots in education, Human-robot interaction



이상필

2008~현재 한성대학교 정보통신공학과 학사과정
관심분야 : Image Processing