Automated Sewing System Enabled by Machine Vision for Smart Garment Manufacturing

Subyeong Ku, HyunWoong Choi, Ho-Young Kim, and Yong-Lae Park

***Abstract*— This paper presents an automated sewing system designed for smart garment manufacturing, incorporating machine vision capabilities into a custom-built sewing machine. The vision system captures an image of the fabric pattern placed between two acrylic plates with a small opening, utilizing a deep learning model to detect and segment the opening, which represents the area of interest on the plate. Subsequently, a specialized algorithm detects a narrow seam line within the segmented image and generates a stitching path alongside the seam line, ensuring a consistent distance. The sewing machine then accurately stitches along the generated path automatically. The vision system utilized in this study achieves a spatial resolution of 68 μm per pixel. The custom-built sewing machine, controlled by an external computer, exhibits a spatial resolution of 10 μm, a translation speed of 60 mm/s, and an adjustable stitching interval ranging from 1 mm to 5 mm. The subsystems and components are interconnected using the Robot Operating System (ROS), enabling seamless communication and integration. The proposed system eliminates the need for human intervention, facilitating automated garment production. This innovative system is expected to play a critical role in realizing the vision of smart garment manufacturing.**

I. INTRODUCTION

Smart manufacturing represents technological innovations with the development of networks, robotics, sensors, artificial intelligence (AI), and the internet of things (IoT) [1], [2], enabling autonomous manufacturing based on the flow of data. For example, qualitative information on the products, such as consumer preferences and needs, is digitalized into data to design and produce customized products for individual consumers. Based on the digitized data of design, robots perform automated production as well as assistance to human operators, while measurement systems, such as sensors and computer vision, check the quality of the output or monitor each process. Beyond automation of individual processes or facilities, machines need to be systematically connected by data, enabling flexible manufacturing for smart factories.

Many industries, such as automobiles, pharmaceuticals, and semiconductors, have been adopting smart manufacturing environments [3]–[5]. Due to the high labor intensity, the garment industry has recently embraced the implementation of automated systems equipped with advanced production equipment and cutting-edge sensing technologies [6], [7]. The garment industry embarks on smart manufacturing by leveraging social commerce platforms that enable consumers to select clothing items of their choice and utilize augmented reality (AR) technology for virtual try-on experiences. Based on each consumer's fitting data with the AR technology in virtual space, sewing patterns, the basic fabric pieces that construct final clothes, are designed and prepared. By combination of these patterns, individualized clothes are produced.

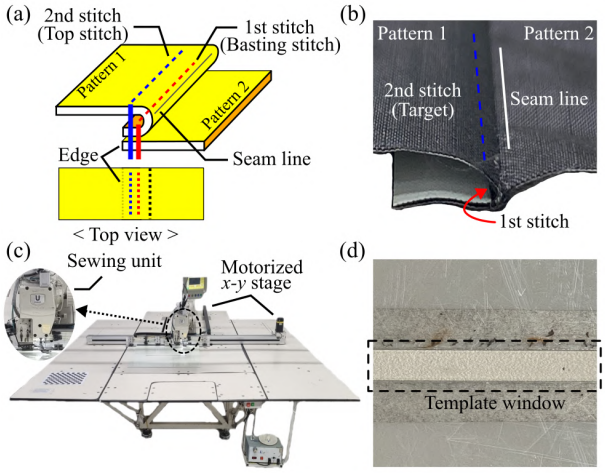

Fig. 1. (a) Description of top stitch process and (b) its actual image. (c) Main components of automated sewing machine. (d) Acrylic template used to hold the fabrics and window (i.e., opening) for sewing.

To realize smart manufacturing in garment industry, computer vision is not only critical to guiding and monitoring operations but also important in inspecting the quality of the intermediate parts as well as the final products. More specifically, computer vision is used to recognize the shapes and the sizes of patterns and to locate them in the desired places [8], to generate sewing paths [6], [9], to monitor the sewing and assembly processes [10], [11], and to check the quality of the work [12], [13]. Therefore, computer vision is a key element to systematically connect and automate multiple processes seamlessly.

Computer vision is also highly useful in generating sewing paths, one of the most critical tasks for automated production. However, due to technical difficulties, it has been limited to generating paths for simple overlapped patterns [6], patches, or logos [9] using simple edge detection algorithms. Most of the main processes for producing clothes are involved with stitching of two different patterns. For example, a simple T-

Manuscript received: March, 7, 2023; Revised May, 16, 2023; Accepted July, 8, 2023.

This paper was recommended for publication by Editor J. Yi upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by the SNU-Hojeon Garment Smart Factory Research Center funded by Hojeon Ltd. and in part by the National Research Foundation grants (RS-2023-00208052). *(S. Ku and H. Choi contributed equally to this work.) (Corresponding author: Y.-L. Park.)*

The authors are with the Department of Mechanical Engineering; the Institute of Advanced Machines and Design (IAMD); the Institute of Engineering Research, Seoul National University, Seoul 08826, South Korea (E-mails: {shetshield; alfred1224;hyk; ylpark}@snu.ac.kr)

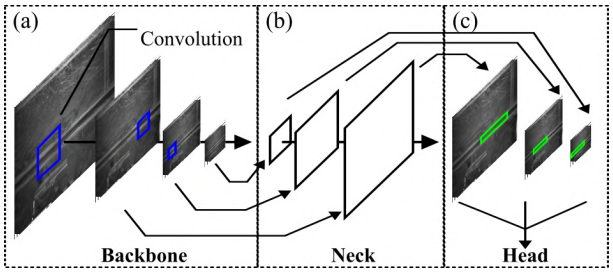

Fig. 2. Simplified architecture of YOLOv5. (a) Backbone for extracting features. (b) Neck for mixing features. (c) Head for inference of bounding boxes, masks, and class prediction.

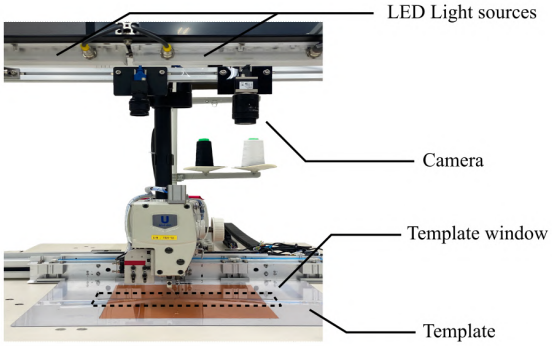

Fig. 3. Vision setup for taking images of template.

shirt consists of patterns of a collar, cuffs, back and front bodies, and left and right sleeves, which are first stitched for connection and then tightly secured by top stitching (Fig. 1-(a) and 1-(b)) [14], [15]. The first stitch (i.e., basting stitch) can be easily sewn along the profile marked on the two overlapped patterns. However, the second stitch (i.e., top stitch) must be sewn along the seam line, formed by turning over the overlapped upper pattern and spreading it apart from the lower one. A predetermined distance from the seam line should be maintained during the top stitch. In this case, the two patterns have the same color usually, making it difficult to distinguish the seam line using conventional computer vision methods. Moreover, environmental conditions, such as uncontrolled illumination, dust in the air, or any unwanted substances on the patterns, make it difficult to detect the seam line and to perform post-processing. Therefore, seam line detection and path generation for top stitch are the biggest challenges in computer vision for automation of garment manufacturing.

Although manual sewing is the most common processes in conventional garment factories, it is prone to yield variations in the quality depending on the levels of skill of the operators [16]. To automate the sewing process, a pattern former which is an automatic sewing machine with a motorized stage has been utilized (Fig. 1-(c)) [17]. A pattern former that consists of a sewing unit and an x-y stage automatically sews the pattern on the stage following the path provided by the operator. Pattern formers are in general suitable for stitching patterns with planar shapes and large areas. A template, made of two thin acrylic plates, is used to fix and hold the target fabric layers that are overlapped (Fig. 1-(d)). The template has a window that exposes the seam line and the area to be sewn. In spite of the automatic features, pattern formers currently available in the market have their own path generation software and interfaces provided by the manufacturers, making it difficult to use them for autonomous and seamless production. Even if a sewing path is generated by a computer, human interventions are unavoidable due to the lack of the capability of autonomous recognition of the patterns and generation of the sewing path.

In this study, we propose an automated sewing system integrated with machine vision and image processing algorithms to detect the seam line and to generate the corresponding top stitch path. In addition, we present an instance segmentation model that infers the template window from the taken image for preprocess of the algorithms. This area is detected by the proposed instance segmentation model, which is transfer learned [18] from a segmentation model called YOLOv5 [19]. Based on the inferred template window, the seam line was detected, and the top stitch path, with a designated distance from the seam line, was generated by the proposed algorithms. The top stitch path was successfully generated by the proposed smoothing algorithm, regardless of dirts on the pattern or changes in illumination. In addition, we built an programmable sewing machine that automatically performs the process of top stitching along the generated sewing path by the integrated vision system without any inputs or interventions from the human operator. With an additional inspection process after stitching, the quality of the stitched path can be evaluated and the resulting data can be generated for monitoring in the proposed system.

II. Algorithms

A. Instance segmentation based on YOLOv5

We employed the YOLOv5 model to detect the template window, which contains the seam line in an image. By using an instance segmentation, instead of object detection that only finds a bounding box around the template window, it is possible to infer an area occupied by the target on the image. Since the image was taken in a controlled environment, the accuracy of segmentation was expected to be high. Therefore, we used a one-stage detector rather than a two-stage detector to reduce the inference time. The architecture of the YOLOv5 instance segmentation model consists of three main components: a backbone, a neck, and a head, as shown in Fig. 2. In the backbone, features are extracted from the input image by convolution and subsampling (Fig. 2-(a)). In the neck, each output of the backbone becomes the input to the layer with the corresponding resolution, and the features are mixed through upsampling (Fig. 2-(b)). In the head, masks and classes are predicted at each image scale (Fig. 2-(c)), and the results are combined into a single image.

To fine-tune the YOLOv5 model, the 69 images, including the template window, were taken using a vision environment (Fig. 3). Six images were assigned to the validation dataset, the other six images were assigned to the test dataset, and the rest of the images were assigned to the train dataset.

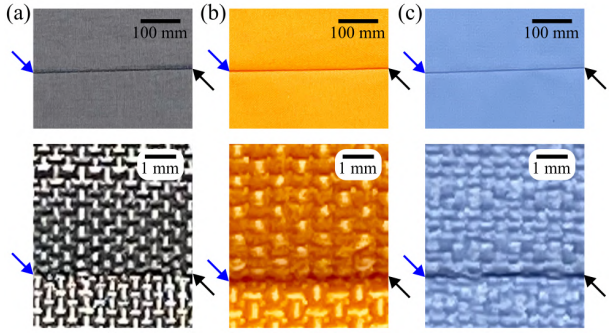

Fig. 4. Fabrics used in experiments. (a) Black, (b) orange, and (c) blue colored fabric, and their magnified images near the seam line. Blue and black arrows indicate the start and the end points of the seam line, respectively.

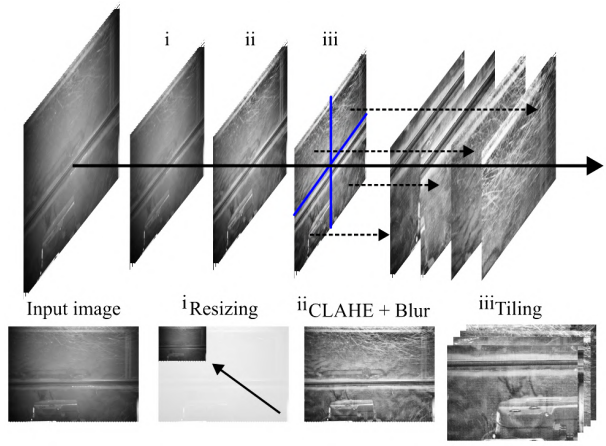

Fig. 5. Sequence of preprocessing. i) Resizing image. ii) Applying contrast limited adaptive histogram equalization and Gaussian blur. iii) Image tiling.

The physical setup consists of a monochromic camera (BFS-U3-123S6M-C, FLIR) and illuminations (EuroBrite^{TM} Bar Lights, Advanced Illumination). Three types of fabrics with three colors of black, orange, and blue were used to train, and each type has its own weaving pattern and seam line (Fig. 4-(a), 4-(b), and 4-(c)). Since the light reflected from the fabric is taken, the image changes depending on the color of the fabric, and thus we chose fabrics from dark to bright colors. In addition, images were taken while changing the exposure time from 10,000 to 100,000 μsec. to enable detection under various illumination conditions. The size of the image was 4096 $\times$ 3000 pixels, and our system had a resolution of 68 μm per pixel.

Before training the model, we performed a sequence of image preprocessing (Fig. 5). Since the input image size was large, which was not suitable for training the model, the image was resized to 1000 $\times$ 750 while maintaining the aspect ratio of the image (Fig. 5-i). Then, a contrast-limited adaptive histogram equalization (CLAHE) [20] was applied to the image. The CLAHE divides the image into a grid of tiles, places a limit on pixel intensity, redistributes the values over the limit, and equalizes the histogram of the grid. Since the installed light sources did not cover the entire area of the template, intensity gradation appeared along the length of the window in the image. Therefore, the CLAHE was

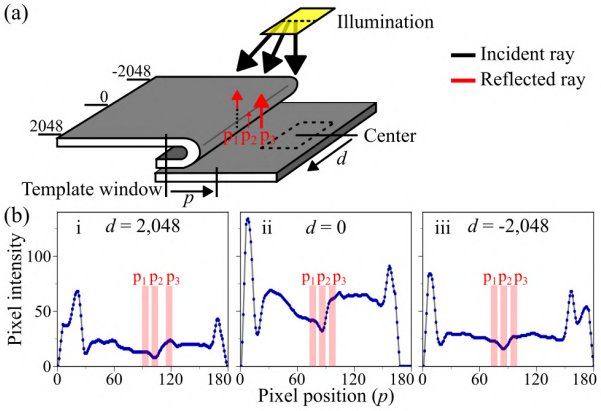

Fig. 6. Schematic of seam line detection principle and pixel intensity plots in different d and p locations.

applied to the image to reduce the difference in brightness between the center and the edge of the image, and then a Gaussian blur was applied to reduce the sharpness (Fig. 5-ii). In the image, the template window occupied 150 pixels high, which was very small compared to the total height of 3,000 pixels. Therefore, 4 $\times$ 4 tilings were applied to magnify small features (Fig. 5-iii). The image was divided into 16 tiles, and each area was resized to 1000 $\times$ 750 pixels and added to the dataset. We applied data augmentation to add variance to the image of the dataset, such as rotation to detect the template window with various orientations, mosaic for small feature detection, and cropping of the image to vary the size and the position of the template window. The YOLOv5 model was fine-tuned with the processed images using a GPU (A100-SXM4-40GB, NVIDIA). On the other hand, in the case of inference, only the CLAHE and the Gaussian blur were applied to the input image to predict the template window.

B. Seam line detection and path generation

The seam line is detected by the difference of the amount of light reflected by the fabric under the illuminations (Fig. 6-(a)). The light sources were placed asymmetrically to the camera to amplify the difference in reflected light. p_1 represents the curved part of the folded upper pattern, which becomes flattened as it goes away from the seam line. p_2 and p_3 represent the seam line and the flat part of the bottom pattern, respectively. Ideally, the incident light on the seam line is minimally reflected. Thus, the pixel intensity is the local minimum at this point (p_2) (Fig. 6-(b)-i, 6-(b)-ii, and 6-(b)-iii). Since the light source did not cover the entire area of the template, the average value of the pixel intensity differed according to the value of d, representing the pixel distance from the center of the light source. However, regardless of the value of d, the local minimum intensity in the template window was obtained from the pixel position p, near the seam line, except for both boundaries.

We developed an image processing algorithm for seam line detection based on the following assumptions: i) In the detected template window, each column of the image

Algorithm 1 Suggest possible seam line points

Input: width of detected template window w, adjacent pixel distance adj_d, i-th column of template window array C_i, number of seam line points g, rejected group of index R, temporary array A
Output: seam line candidates L

Initialization : $adj_d \leftarrow 3$, $g \leftarrow 5$
for $i = 0$ to w **do**
  $j = 0$, array A
  while A length $\neq g$ **do**
    index of j-th minimum intensity in C_i k_j, $j \leftarrow j+1$
    if $C_i[k_j - 1] = 0$ or $C_i[k_j + 1] = 0$ **then**
      append k_j to R
    else
      for $u = 1$ to adj_d **do**
        if $k_j + u \in R$ or $k_j - u \in R$ **then**
          append k_j to R
        else
          append k_j to A and R
        end if
      end for
    end if
  end while
  append A to L
end for
return L

Algorithm 2 Determine seam line

Input: width of detected template window w, number of seam line points g, seam line candidates L, j-th seam line candidate SL_j, j-th group connectivity score s_j, j-th group minimum pixel difference min_j, k-th pixel index difference Δp_k, $(i-1)$-th pixel index in SL_j $p_{i-1}[j]$, j-th group seam line index p_j
Output: u-th seam line SL_u

Initialization : $g \leftarrow 5$, $SL_j[0] \leftarrow L[0][j]$
for $i = 1$ to w **do**
  for $j = 0$ to g **do**
    $min_j \leftarrow 9999$, $p_{i-1}[j] \leftarrow SL_j[i-1]$
    for $k = 0$ to g **do**
      Δp_k = abs$(L[i][k] - p_{i-1}[j])$
      if $\Delta p_k < min_j$ **then**
        $min_j \leftarrow \Delta p_k$, $p_j \leftarrow L[i][k]$
      end if
    end for
    append p_j to SL_j, $s_j \leftarrow s_j + min_j$
  end for
end for
$u \leftarrow argmin(s_1, s_2, ...s_g)$
return SL_u

Algorithm 3 Smoothing and generating top stitch path

Input: width of detected template window w, seam line SL, smoothing window size m, number of window n, smoothed seam line point $\hat{y}$, temporary array A, resolution of vision system r, j-th top stitch point ts_j, predetermine gap d_g
Output: top stitch path TS

Initialization : $m \leftarrow 300$
interval $n = integer(w/m)$
for $i = 0$ to n **do**
  $A = SL[i * m : (i+1) * m]$
  apply Savitzky-Golay filter to A, result $\hat{y}$
  for $j = 0$ to m **do**
    $ts_j = \hat{y}[j] + d_g/r$ and append p_j to TS
  end for
end for
return TS

array has a single seam line point. ii) The seam line points of adjacent columns are at similar locations. iii) The seam line point in each column has the minimum intensity. With these assumptions, the seam line detection was possible using only the pixel intensities without applying additional image processing techniques (e.g., thresholding, edge detection) and also possible regardless of the illumination conditions. An algorithm to find the points of interest (e.g., p_1, p_2, and p_3) with the local minimum intensity in the template window is first developed, as shown in Algorithm. 1.

This algorithm found g indexes of pixels with the minimum intensity in all columns included in the detected template window. Ideally, the index with the minimum intensity in the template window is near the seam line, but we found g ($g = 5$) candidates for the seam line points. At the boundaries of the template window, the pixel intensities were 0. Since these values were not of interest, they were excluded from the candidates of the seam line points, and all the indexes connected to these indexes were excluded, too. In addition, if the index that was found was within the adjacent distance, adj_d ($adj_d = 3$), this index was also excluded from the seam line point since it was considered connected. By adjusting only two parameters, we were possible to find candidates of the seam line points in the image. From the suggested seam line points L, the actual seam line points are found based on the connectivity between the adjacent columns (Algorithm. 2). Since the seam line generated by folding the upper pattern is a smooth curve, the actual seam line among the candidates has the minimum sum of the change in the pixel position.

The top stitch path was generated from the determined seam line SL_u. Before generating the path, the determined seam line was smoothed because the seam line did not have a smooth curve due to noise caused by dust or substances on the patterns even if the connectivity was minimized. Therefore, we divided the template window into n intervals, smoothed each interval, and merged them. Rather than applying a moving average that is signficantly affected by outliers, we applied a Savitzky-Golay smoothing filter ($\hat{y}$) [21], a finite impulse response system analysis method, instead of time-consuming local regression. After smoothing the seam line, the point p_j of the path TS was generated and merged, using the distance per pixel r of the vision system and the predetermined gap for the top stitch d_g. d_g was set to be 1.6 mm, the smallest gap required for precise clothing production (Algorithm. 3).

III. System

An automated sewing system is composed of a vision system and a custom-built sewing machine. As mentioned

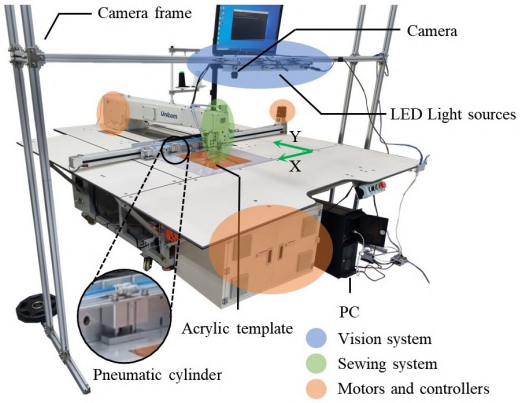

Fig. 7. Components of automated sewing system.

in Sec. II, the vision system includes the camera and the light sources to detect the seam line of the patterns. This section describes the details of the sewing machine, including the hardware specifications. In addition, the communication methods between the components of the system and operational sequences are introduced.

A. Automatic sewing machine

To automatically sew along the generated top stitch path, we developed an automatic sewing machine using a commercial pattern former (UAS-H700-D, UNICORN). The operational unit of the developed machine consists of three motors: two servo motors (SGM7J-04AFA21, SGM7J-08AFA21, YASKAWA) that are attached to the x-y stage to control the position of the template, and one servo motor (SGM7J-08AFA21, YASKAWA) that controls the position of the sewing needle and trims the thread. By connecting the motor controllers and an external PC, it is possible to control the position and the speed of the motors without using the factory-installed program. The sewing machine also has four pneumatic-guided cylinders which hold the template from vibrations and a presser foot that flattens the patterns during sewing (Fig. 7).

The maximum travel distance of the stage is 1,230 mm in the x direction, parallel to the long side of the template window, and 720 mm in the y direction. The spatial resolution of the x-y stage is 10 μm in both directions. The maximum RPM of the motor of the sewing needle is 600. The average speed for linear translation is proportional to the RPM of the motor of the sewing needle and the sewing interval. The x-y stage has the maximum translation speed of 60 mm/s when the RPM is 600 and the sewing interval is 5 mm.

B. Communication

The entire system can be divided into three sub-systems: the sewing machine, the camera, and the lighting. To facilitate seamless communication between these sub-systems, we employed the Robot Operating System (ROS) with TCP/IP communication [22]. In the ROS, each sub-system is represented as a node, and communication is established through messages in ROS topics. The system consists of four nodes: the sewing machine node, the vision node, the light node, and the graphical user interface (GUI) node (Fig. 8). The sewing machine node controls the motors of the x-y stage and monitors the motor encoder values. The vision node captures the images through the camera and executes the proposed algorithm. The light node manages the state of the LED light sources, and the operator can monitor and control the top stitching process through the GUI node.

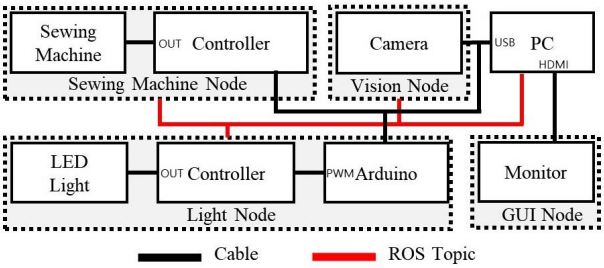

Fig. 8. Schematic diagram of communication in the automated sewing system.

IV. Experiments

We conducted experiments for evaluating the performances the inference of the trained segmentation model, the seam line detection and the top stitch path generation using the proposed algorithm, and the sewing machine and stitching quality.

A. Instance segmentation

For evaluating the trained model, the black (Fig. 9-(a)) and the blue (Fig. 9-(b)) fabrics were used because the amount of reflected light varied with colors. Then, the CLAHE and the Gaussian blur were applied to the taken images, which were provided and it was input to the trained model as the inputs (Figs. 9-(a)-(b)-middle). The time taken for the segmentation was about 0.3 sec. when using a GPU (Geforce 3080 TI, NVIDIA). The trained model detected the template window regardless of the optical noise reflected on the glossy acrylic template, the color and the unique weaving pattern of the fabric, and the brightness of the illuminations.

B. Seam line detection and top stitch path generation

We conducted an experiment to check whether the seam line was detected and the top stitch path was properly generated through the proposed algorithm, and compared the result with the outcome from the conventional edge detection algorithm. Using the segmented area as a mask, only the template window was isolated from the image and became the region of interest (ROI). By setting the ROI, the cropped image of the template window, seam line detection and post-processing were simplified. The result of application of the seam line detection algorithm and the generated top stitch path can be seen in Figs. 9-(a)-(b)-bottom. We compared the result with that of conventional edge detection algorithms. A Canny [23], [24], a Laplacian [25], and a Sobel [26] operators were applied to the ROI of the black pattern (Fig. 9-(c)). In the black pattern, it was unable to find the seam line by the Canny nor the Laplacian operator. In the case of the Sobel operator, as expressed with green, the red seam line found by the proposed algorithm overlapped the

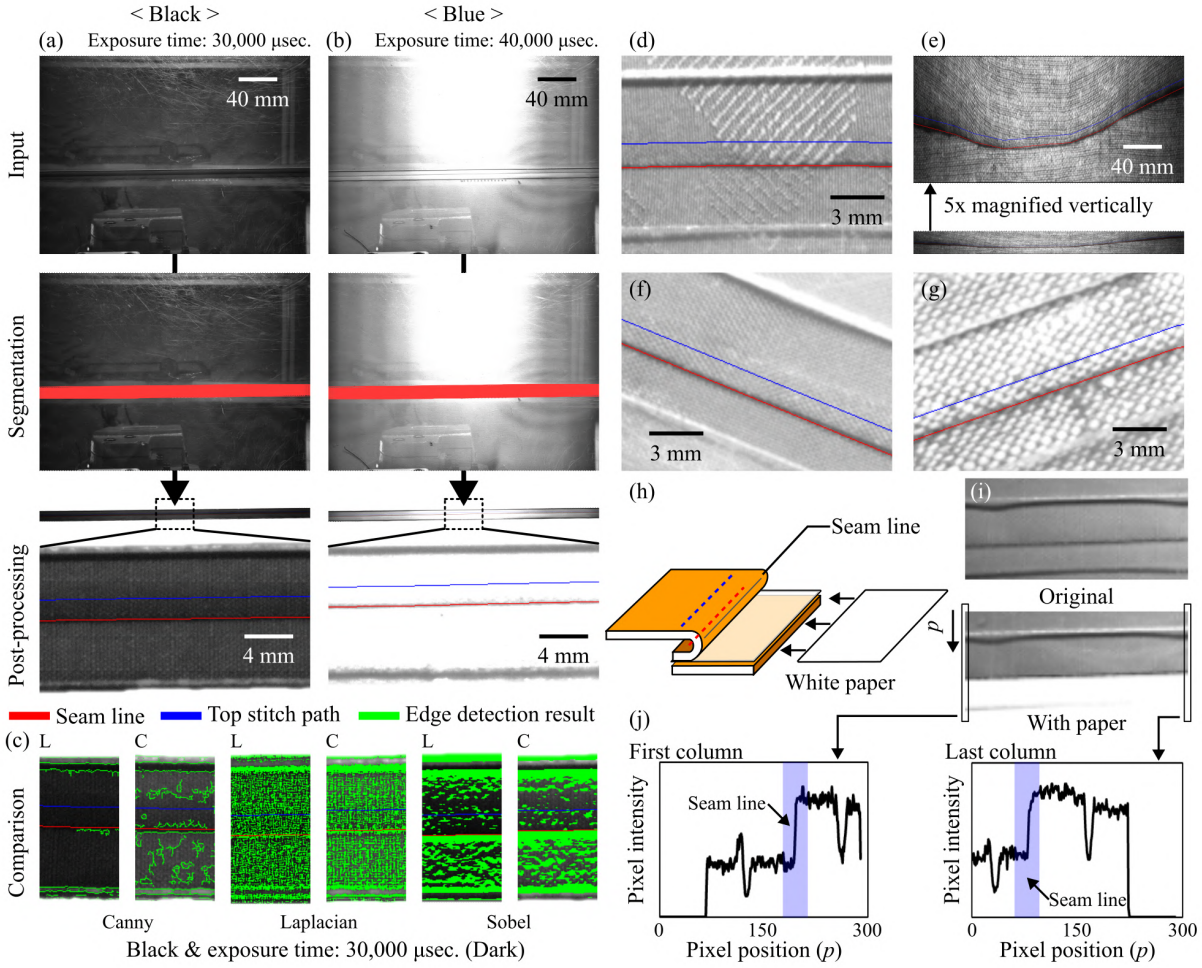

Fig. 9. Sequence and result of seam line detection and top stitch path generation. Black and blue fabrics with exposure time of (a) 30,000 μsec. and (b) 40,000 μsec., respectively. The sequence consists of input and preprocessing (top), segmentation of template window (middle), post-processing (i.e., masking the template window, smoothing, and top stitch path generation) (bottom). (c) Comparison with conventional edge detection algorithms. L and C represent the left end and the center of the image, respectively. Results of seam line detection in (d) the embossed fabric, (e) curved seam line in the fabric with horizontal stripes, (f) and (g) the diagonal seam line. (h) White paper insertion for quantifying performance. (i) Photos of with and without paper. (j) Intensity plots of the reflected light for the first and the last columns, respectively.

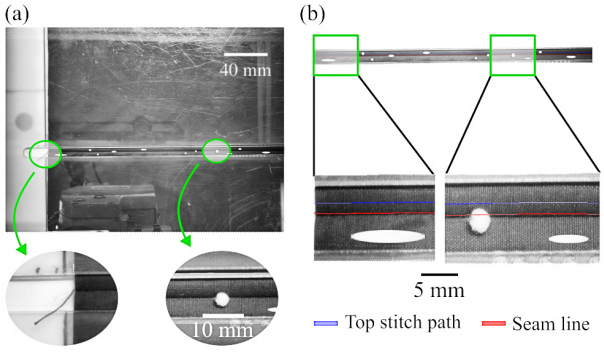

Fig. 10. Robustness test for the proposed algorithm. (a) A single strand of the edge thread is unwound (Left) and lots of dirt is placed on the template window and the seam line (Right). (b) Result of segmentation, seam line detection, and top stitch path generation.

green area. All the conventional algorithms were affected by the image brightness and required complex image processing before and after applying the algorithms to detect the seam line.

The proposed detection algorithm was possible to detect the seam lines on the embossed fabric (Fig. 9-(d)), the curved seam on the fabric with horizontal stripes (Fig. 9-(e)), and the diagonal seam (Fig. 9-(f) and 9-(g)).

To quantitatively evaluate the performance of the proposed detection algorithm, a piece of white paper was inserted into the gap between the patterns (Fig. 9-(h)). Since the white color is highly reflective, it has a high pixel intensity when photographed, making it easier to find the ground truth with contrast near the seam line (Fig. 9-(i)). Image processing was used to find the seam line boundary (Fig. 9-(j)). We found the ground truths for the black, the blue, and the orange patterns and compared the with the results from the proposed algorithm. The average position errors for the black, the blue, and the orange patterns were 0.05, 0.05, and 0.09 mm, respectively, with the standard deviations of 0.04, 0.04, and 0.09 mm, respectively.

We tested the robustness of the segmentation model and the proposed algorithm for seam line detection (Fig. 10-(a)). An unwound thread at the edge of the pattern makes the start point for sewing inconsistent, and the dirt on the seam line

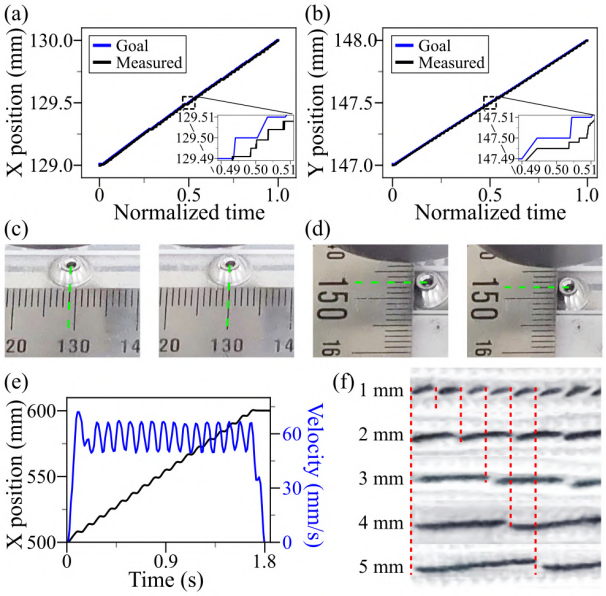

Fig. 11. Test results of the automatic sewing machine. The goal position and the measured position by the absolute encoder in (a) x-axis and (b) y-axis and the corresponding visual result in (c) x-axis and (d) y-axis. (e) Position and velocity measurement in x-axis. (f) Sewing results with stitching intervals from 1 mm to 5 mm.

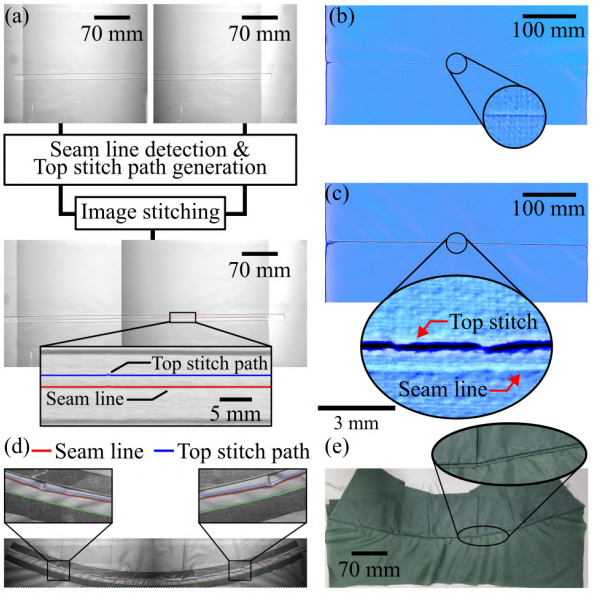

Fig. 12. Automation procedure for stitching. (a) Left and right images of template (top) and stitched image with detected seam line and generated top stitch path (bottom). (b) Before and (c) after sewing. Magnified image shows the consistent distance (1.58 mm) between the top stitch and the seam lines. (d) Detected curved seam line and (e) the result of the top stitching.

makes the top stitch path jagged. The unwound thread was rejected by the trained segmentation model, and the generated top stitch path was smooth by the proposed algorithm, even with the presence of dirt on the seam line (Fig. 10-(b)). The experimental results show that the developed sewing machine play a key role of the pattern former, while precisely operating with the sewing path generated by the vision system.

C. Top stitching with automatic sewing machine

We conducted an experiment to evaluate the performances of the custom-built sewing machine in terms of the spatial resolution, the maximum translation velocity, and the stitching interval. The sewing machine was controlled to move 1 mm in the x and y directions by moving 100 steps in 10 μm increments with simultaneous measurement of an absolute encoder (Fig. 11-(a) and 11-(b)). The inset plots indicate that when the measured position of the x-y stage reached the target position, the next goal position was input to the x-y stage. This sequence was repeated 100 times to move 1 mm in x and y directions. As a result of analyzing the images taken by the vision system before and after the operation, the system moved 1 mm in each axis and showed the spatial resolution of approximately 10 μm (Fig. 11-(c) and 11-(d)).

Next, we tested the maximum translation speed under the condition of stitching. We moved the stage 100 mm in the x direction under the maximum RPM (600) of the sewing needle, and the result is shown in Fig. 11-(e). The black and blue curves show the position in the x-axis and the speed, respectively. The stage moved 100 mm for 1.67 sec. with the average speed of 60 mm/s.

Since the optimal stitching interval is different for each fabric, this length needs to be changed by the operator. We sewed the pattern by changing the stitching interval from 1 mm to 5 mm, and the results were captured with the vision system (Fig. 11-(f)). The actual stitching interval showed a similar length to the input value.

V. Application

We set up an automated sewing system by integrating the vision system with the sewing machine and connecting components through the ROS. The integrated system generated the top stitch path based on the captured template image by itself and sewed along the path without any interventions of the human operator. We demonstrated the automated stitching using the integrated system. First, each part of the pattern was taken (Fig. 12-(a)), and the top stitch path was generated using the proposed algorithm. The process of image merging was required since the patterns used to make actual clothes would be longer than a single capture by the vision. The top stitch path was generated at a distance of 1.58 mm (1/16 inch) from the seam line in the merged image, and the custom-built machine sewed along the path. 1.58 mm is the smallest gap between the seam line and the top stitch line used in the typical garment production field [27], [28]. The images of the patterns for the straight seam line before and after the top stitch are shown in Figs. 12-(b) and 12-(c), respectively. The curved seam line was also detected by the proposed algorithm (Fig. 12-(d)) and top stitch was possible along the generated curved path (Fig. 12-(e)).

VI. Discussion

In most cases, the proposed algorithm was able to detect the seam line by adjusting two parameters (adj_d and g). The proposed algorithm finds the points with the intensities of local minimums and estimates the seam line by connecting them. Therefore, if the pattern is matte and dark with a low

reflectivity, it is difficult to distinguish. In addition, if the pattern has large mesh patterns with high air permeability, it is possible to have multiple local minimums along the stripes, requiring additional image processing.

Our proposed system and algorithm were designed to automate legacy production equipment as a starting point for smart garment manufacturing. To realize an advanced system for smart manufacturing, individual production machines need to be physically connected. Therefore, we are currently working on a system that integrates a mobile platform, a collaborative robot, and a robotic gripper, to autonomously handle and transport various types of fabric patterns for multi-step sewing processes to fully automate the production process.

VII. Conclusion

We developed an automated sewing system by integrating a custom-built sewing machine with a machine vision system, which does not need any interventions or assistance from the human operator. All the components were systematically connected through the ROS. In the vision part, a trained deep learning model and the proposed algorithms were sequentially executed. A template window was detected and segmented by the trained deep learning model in the captured image, and the top stitch path was generated based on the seam line by the proposed algorithms. The custom-built sewing machine was controlled by an external PC and showed a spatial resolution of 10 μm, the maximum translation speed of 60 mm/s, and the adjustable stitching interval from 1 mm to 5 mm. Through this integrated system, automated pattern sewing and simultaneous monitoring were possible. By repeating the image processing of the result after top stitching, the quality of the stitched path was evaluated and the resulting data were generated. The automation of the sewing machine, the quality assessment of the output through the vision system, and the generated data for use in the next process will enable seamless production by significantly increasing the connectivity of multiple processes in garment manufacturing. Therefore, we expect our system to play an essential role in achieving smart manufacturing in the future garment industry.

References

[1] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2017.

[2] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Bus. Inf. Syst. Eng.*, vol. 6, no. 4, pp. 239–242, 2014.

[3] C. Cronin, A. Conway, and J. Walsh, "Flexible manufacturing systems using iiot in the automotive sector," *Procedia Manuf.*, vol. 38, pp. 1652–1659, 2019.

[4] N. S. Arden, A. C. Fisher, K. Tyner, L. X. Yu, S. L. Lee, and M. Kopcha, "Industry 4.0 for pharmaceutical manufacturing: preparing for the smart factories of the future," *Int. J.Pharm.*, vol. 602, p. 120554, 2021.

[5] J. Moyne and J. Iskandar, "Big data analytics for smart manufacturing: Case studies in semiconductor manufacturing," *Processes*, vol. 5, no. 3, p. 39, 2017.

[6] S. Lee, S. H. Rho, S. Lee, J. Lee, S. W. Lee, D. Lim, and W. Jeong, "Implementation of an automated manufacturing process for smart clothing: The case study of a smart sports bra," *Processes*, vol. 9, no. 2, p. 289, 2021.

[7] S. Ku, J. Myeong, H.-Y. Kim, and Y.-L. Park, "Delicate fabric handling using a soft robotic gripper with embedded microneedles," *IEEE Rob. Autom. Lett.*, vol. 5, no. 3, pp. 4852–4858, 2020.

[8] E. Torgerson and F. W. Paul, "Vision-guided robotic fabric manipulation for apparel manufacturing," *IEEE Control Syst. Mag.*, vol. 8, no. 1, pp. 14–20, 1988.

[9] Brother. (2022) Brother Electronic Controlled Programmable Sewing Machine with Vision Camera System JJB, brother industries, ltd, nagoya, aichi, japan, 7905387. [Online]. Available: https://www.brother-usa.com/products/7905387

[10] W.-K. Jung, D.-R. Kim, H. Lee, T.-H. Lee, I. Yang, B. D. Youn, D. Zontar, M. Brockmann, C. Brecher, and S.-H. Ahn, "Appropriate smart factory for smes: concept, application and perspective," *Int. J. Precis. Eng. Manuf.*, vol. 22, pp. 201–215, 2021.

[11] J.-Y. Lee, D.-H. Lee, J.-H. Park, and J.-H. Park, "Study on sensing and monitoring of sewing machine for textile stream smart manufacturing innovation," in *IEEE Int. Conf. Mechatron. Mach. Vision Pract. (M2VIP)*, 2017.

[12] Y. Li, W. Zhao, and J. Pan, "Deformable patterned fabric defect detection with fisher criterion-based deep learning," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 1256–1264, 2016.

[13] H. Kim, W.-K. Jung, Y.-C. Park, J.-W. Lee, and S.-H. Ahn, "Broken stitch detection method for sewing operation using CNN feature map and image-processing techniques," *Expert Syst. Appl.*, vol. 188, p. 116014, 2022.

[14] A. Shahriar, "The optimization of knitted T-shirt for rapid production process," *Int. J. Text. Sci.*, vol. 8, no. 1, pp. 16–25, 2019.

[15] R. E. Glock and G. I. Kunz, *Apparel manufacturing: Sewn product analysis*. Pearson/Prentice Hall, Upper Saddle River, NJ, 2005.

[16] G. Li, C. M. Haslegrave, and E. Corlett, "Factors affecting posture for machine sewing tasks: The need for changes in sewing machine design," *Appl. Ergon.*, vol. 26, no. 1, pp. 35–46, 1995.

[17] T. Gries and V. Lutz, "Application of robotics in garment manufacturing," in *Automation in Garment Manufacturing*, R. Nayak and R. Padhye, Eds. Woodhead Publishing, 2018, pp. 179–197.

[18] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. IGI global, 2010, pp. 242–264.

[19] G. Jocher et al., "ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference," Feb. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6222936

[20] K. Zuiderveld, "Contrast limited adaptive histogram equalization," *Graphics Gems IV*, pp. 474–485, 1994.

[21] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.

[22] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, vol. 3, 2009.

[23] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 6, pp. 679–698, 1986.

[24] Y. Han, A. Varadarajan, T. Kim, G. Zheng, K. Kitani, A. Kelliher, T. Rikakis, and Y.-L. Park, "Smart skin: Vision-based soft pressure sensing system for in-home hand rehabilitation," *Soft Robot.*, vol. 9, no. 3, pp. 473–485, 2022.

[25] X. Wang, "Laplacian operator-based edge detectors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 886–890, 2007.

[26] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the sobel operator," *IEEE J. Solid-State Circuits*, vol. 23, no. 2, pp. 358–367, 1988.

[27] D. Aasen, R. S. Mong, and P. Fendley, "Topological defects on the lattice: I. the ising model," *J. Phys. A: Math. Theor.*, vol. 49, no. 35, p. 354001, 2016.

[28] J. H. Hammond and A. Raman, *Sport Obermeyer Ltd*. Harvard Business School Boston, 1994.