Eurasia Journal of Science and Technology
Print ISSN: 2663-1024
Online ISSN: 2663-1016
DOI: https://doi.org/10.61784/ejst3150

THE APPLICATION OF VISUAL-BASED ACUPOINT POSITIONING TECHNOLOGY IN ACUPOINT ROBOTS

MengDi Wang, DianHao Wu*, Yi Wang, Qiao Fang, ZhenBiao Chen, KunLin Xie
School of Mechanical Engineering and Automation, University of Science and Technology Liaoning, Anshan 114051, Liaoning, China.
**Corresponding Author: DianHao Wu*

Abstract: Acupuncture efficacy depends on the precision of acupoint localization. Based on prior research, this paper proposes a technical framework for automatic facial acupoint positioning and robotic acupuncture: facial information is captured using RGB-D vision, and an improved YOLOv8 model detects 27 index acupoints across 15 categories, enabling precise three-dimensional coordinate mapping for localization. A specialized end-effector integrating needle exchange, holding, and rotation functions converts lifting, thrusting, twisting, retention, and withdrawal into linear and rotational motions. An enhanced APF-RRT* algorithm is proposed to achieve obstacle-avoidance path planning during continuous multi-acupoint needling. Experiments conducted with UR5e, Kinect V2, and a simulated human head demonstrate that the system achieves average errors of 0.54 mm, 0.50 mm, and 0.66 mm in the three dimensions, achieving millimeter-level accuracy in acupoint localization.
Keywords: Acupuncture robot; Acupoint localization; YOLOv8; Trajectory planning; Machine vision

1 INTRODUTION

1.1 Research Background and Problem Statement

As an important part of the Traditional Chinese Medicine treatment system, the clinical effectiveness of acupuncture is related not only to syndrome differentiation and acupoint selection, but also to the accuracy of acupoint location and needling trajectory [1,2]. Traditional acupoint location commonly uses body-surface landmarks, proportional bone measurement, and finger-cun methods. Although these methods have long clinical foundations, they are subjective, difficult to repeat, and lack a unified quantitative engineering reference. In the facial region in particular, acupoints are densely distributed, surrounding blood vessels, nerves, and muscular tissues are complex, and individual facial contours and expressions vary. Relying only on manual experience is therefore insufficient to ensure stable and consistent localization.

After robotic technology is introduced into acupuncture, visual recognition, spatial calibration, trajectory planning, and end-effector motion control can be integrated into a repeatable operating procedure. Compared with manual operation, robots provide higher pose-control accuracy, recordable motion paths, and reproducible operating parameters, making them suitable for acupuncture standardization research and the development of assistive therapeutic devices. For facial acupoint needling, this study focuses on three key issues: whether the end-effector can complete needle changing, holding, and twirling; whether the vision model can stably recognize facial acupoints under limited computing resources; and whether the robotic arm can avoid already inserted needles and reach the target acupoint during continuous multi-needle operation.

1.2 Overview of Domestic and International Research

Early research on acupuncture robots mainly focused on mechanically reproducing needling manipulations, and later gradually shifted toward visual localization, force-controlled interaction, and continuous treatment workflows. In related work, robot-assisted laser acupuncture has been explored through computer vision and low-cost automated acupoint positioning systems [3]. Direct studies on acupuncture robots remain limited outside China; however, research on medical robots and vision-guided minimally invasive systems provides useful technical references for sub-millimeter localization, perception feedback, and safe motion control.

For acupoint localization, existing studies include augmented-reality visualization of facial acupoints, facial acupoint detection based on YOLOv8-pose, real-time acupoint localization using anatomical landmarks and pose estimation, hand-acupoint localization based on dual-attention networks, and back-acupoint localization based on prior information and deep learning [4-8]. In general, these methods can achieve relatively high localization accuracy under specific data and scenarios, but many still have high computational costs, limited suitability for edge deployment, and insufficient robustness to posture and expression changes. In trajectory planning, RRT, RRT*, Informed RRT*, and artificial potential field methods have been widely used for obstacle avoidance in robotic systems [9-11]. For acupuncture, the algorithm must not only generate a collision-free path but also consider inserted needles, patient surface morphology, and smooth robotic-arm motion, so a balance among search efficiency, path quality, and safety is required.

2 ACUPUNCTURE ROBOT SYSTEM AND END-EFFECTOR DESIGN

2.1 System Requirement Analysis

During facial acupuncture treatment, one prescription often contains multiple acupoints. When a robot executes such tasks, the first inserted needle becomes a new spatial obstacle. Subsequent needling trajectories must avoid the needle body so as to prevent contact, traction, or tissue injury. The system must therefore satisfy requirements for acupoint recognition, three-dimensional reconstruction, trajectory planning, end-effector posture adjustment, and safety interaction. The vision subsystem uses a Kinect V2 camera to acquire color and depth images; the upper computer performs acupoint recognition, point-cloud reconstruction, coordinate transformation, and trajectory generation; the robotic arm executes the planned path; and the end-effector performs needle clamping, needle changing, twirling, and needle insertion in cooperation with the robotic arm.

The end-effector design is based on the fundamental manipulations of acupuncture. Lifting-thrusting corresponds to linear reciprocating motion along the needle axis, while twirling corresponds to rotation of the needle around its own axis. Needle retention and withdrawal require the clamping mechanism to provide stable holding and controllable release. Because continuous multi-acupoint needling requires frequent replacement of sterile needles, the end mechanism must also provide needle storage and automatic feeding. To reduce volume and accumulated errors, the original study compared four-degree-of-freedom and three-degree-of-freedom schemes, and finally selected a compact structure in which clamping and twirling are integrated at the front end while lifting-thrusting is implemented through cooperation with the robotic arm.

2.2 End-Effector Structure

The end-effector is mainly composed of a needle storage box, pressure rod, telescopic rod, push rod, miniature linear-guide slide, drill chuck, synchronous-belt pulleys, stepper motors, and bottom mounting plate. The storage box contains guide grooves designed according to the size of acupuncture needles. The needle body is approximately 0.2 mm wide and the needle handle is approximately 1 mm wide. A push-rod inlet and a needle outlet are reserved on the two sides of the storage box. The pressure rod and spring telescopic rod provide continuous downward pressure so that the needles are arranged sequentially. To prevent bending during needle ejection, a guide tube is arranged at the outlet to feed the needle smoothly to the clamping mechanism.

The clamping and twirling mechanism is implemented using a drill chuck. When the front shell of the drill chuck and the outer wall move in opposite directions, the internal structure clamps or releases the acupuncture needle. When they move in the same direction, the whole drill chuck rotates, thereby realizing the twirling motion. Two stepper motors drive the corresponding pulleys through synchronous belts, making the clamping and rotation processes controllable. During the design stage, the synchronous-belt tension, deflection angle, and pulley center distance were calculated to ensure that the drill chuck could rotate forward and backward within the allowable range and meet the needs of practical acupuncture manipulations. Finite element simulation was then used to evaluate the stress and deformation of the needle-storage-box guide tube, support plate, and base plate, verifying that the key components meet the strength and stiffness requirements under the expected load, see Figure 1.

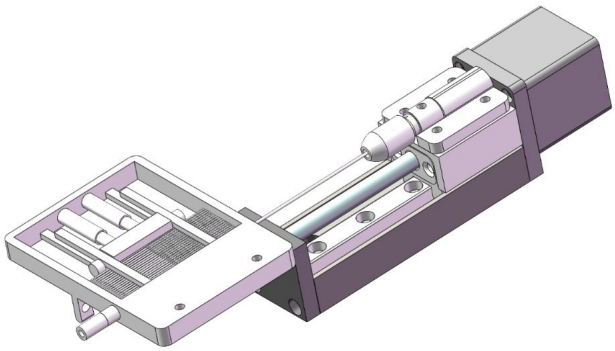

Figure 1 Final Design Scheme of the End-effector

3 FACIAL ACUPOINT LOCALIZATION METHOD BASED ON YOLOV8

3.1 Dataset Construction and Training Settings

Facial acupoint detection involves small targets with dense distribution, and different subjects vary in skeletal landmarks, soft-tissue thickness, and facial expression. For this reason, the original study constructed a dedicated facial acupoint dataset. The dataset covers 15 acupoint names and 27 indexed points, including Jingming, Cuanzhu, Yuyao, Sizhukong, Tongziliao, Yingxiang, Yintang, Shuigou, Dicang, Chengjiang, Daying, Juliao, Yangbai, Sibai, and Taiyang. The dataset contains 620 images in total, including 500 training samples and 120 test samples. The male and female

sample distribution is shown in Table 1.(The Figure 2 is from the Second Affiliated Hospital of Harbin Medical University. All the volunteers have given their verbal consent)

Table 1 Composition of the Facial Acupoint Dataset

Gender	Training samples	Test samples	Total
Male	286	69	355
Female	214	51	265
Total	500	120	620

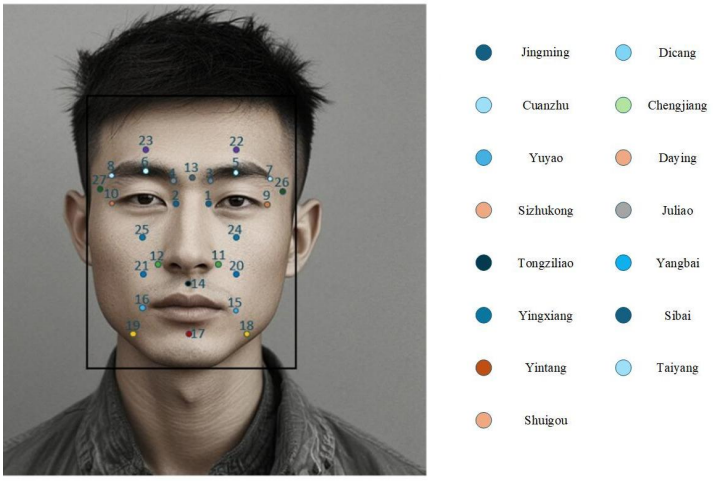

Figure 2 Facial Acupoint Points Labeled in the Dataset

The model training environment was Windows 10 with 16 GB of memory, an NVIDIA GTX 1050Ti GPU, and an AMD Ryzen 5 3600 six-core processor. The software environment used PyTorch 1.8.0 managed by Anaconda and CUDA 10.1. The number of training epochs was 300, the input image size was 640, the batch size was 8, the optimizer was SGD, the final learning rate was 0.01, and the optimizer momentum was 0.937. Mosaic data augmentation and Dropout regularization were used during training to increase sample diversity and suppress overfitting.

3.2 Structural Improvement of YOLOv8-SDI-C3Ghost

YOLOv8 was selected as the baseline network because of its fast detection speed, clear structure, and mature engineering deployment [12]. To address the limited memory and computing resources of edge devices, the original study introduces the C3Ghost lightweight module into YOLOv8. The basic idea of the Ghost module is to first obtain intrinsic features through a small number of ordinary convolutions and then generate more similar but not identical feature maps using low-cost linear operations, thereby reducing parameters and computation [13]. C3Ghost introduces the Ghost bottleneck into the C3 module, compressing the network scale while preserving feature representation capability.

Facial acupoints are small targets, and the localization result is sensitive to local texture and boundary details. The SDI module is used to replace the Concat operation in the Neck part. Through multi-level feature fusion, it injects high-level semantic information and low-level detail information at the same time, enhancing the model's ability to distinguish small-target positions. C3Ghost focuses on lightweight design, whereas SDI focuses on cross-layer feature refinement. Their combination forms the YOLOv8-SDI-C3Ghost model. This improvement does not simply increase network complexity; rather, it balances accuracy and deployment cost.

3.3 Visual Localization Experimental Results

Table 2 Ablation Experiment Results

SDI	C3-Ghost	P(%)	mAP50(%)	mAP50-95(%)	Params(MB)	GFLOPs	Size(MB)
×	×	**99.91**	**99.50**	**99.10**	**3.54**	**10.2**	**10.2**
√	×	**99.92**	**99.50**	**99.22**	**3.57**	**10.5**	**7.5**
×	√	**99.91**	**99.50**	**99.03**	**2.52**	**7.7**	**5.2**
√	√	**99.92**	**99.50**	**99.26**	**2.55**	**8.0**	**5.5**

The ablation results show that introducing SDI alone increases mAP50-95 from 99.10% to 99.22%, although the parameter size increases slightly, see Table 2. Introducing C3Ghost alone reduces the parameter size from 3.54 MB to

2.52 MB and reduces the model size to 5.2 MB, but the mAP50-95 is 99.03%. When SDI and C3Ghost are used together, the model achieves an mAP50-95 of 99.26%, with a parameter size of 2.55 MB, a computational cost of 8.0 GFLOPs, and a model size of 5.5 MB. Compared with the baseline YOLOv8n, this scheme achieves a good overall balance between lightweight deployment and localization accuracy, see Table 3 and Figure 3.

Table 3 Comparison of Different Improved YOLOv8 Models

Model	P(%)	mAP50-95(%)	Params(MB)	GFLOPs	Size(MB)
YOLOv8-C2f-ScConv	**99.92**	**99.13**	**3.34**	**9.9**	**7.0**
YOLOv8-fasternet-bifpn	**99.91**	**98.88**	**3.49**	**11.7**	**7.3**
YOLOv8-slimneck	**99.91**	**98.80**	**3.39**	**9.5**	**7.1**
YOLOv8-SDI-C3Ghost	**99.92**	**99.26**	**2.55**	**8.0**	**5.5**

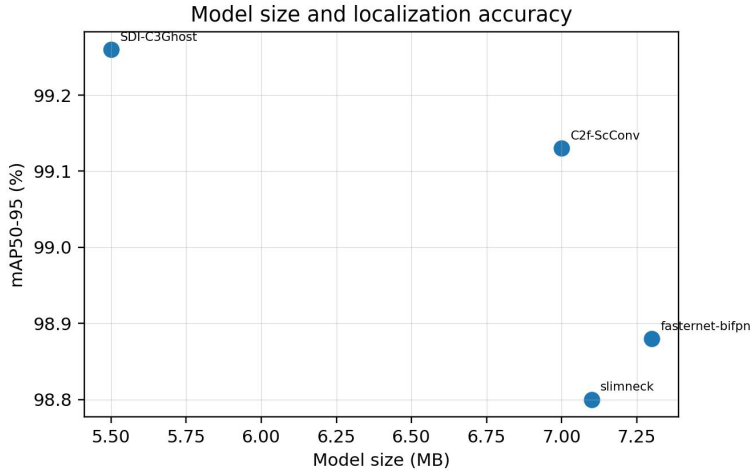

Figure 3 Relationship between Model Size and mAP50-95 for Different Models

Compared with YOLOv8-C2f-ScConv, YOLOv8-fasternet-bifpn, YOLOv8-slimneck, and other schemes, YOLOv8-SDI-C3Ghost has fewer parameters and a smaller model size while achieving the highest mAP50-95. The detection results show that the predicted points are basically consistent with the manually labeled points. In tests on an external dataset, the model still maintains good localization under complex lighting conditions. However, when facial expression changes are obvious, acupoints far from the facial contour occasionally show slight offsets, indicating that the model's robustness against expression interference and dynamic clinical scenarios still needs further improvement.

4 TRAJECTORY PLANNING METHOD FOR THE ACUPUNCTURE ROBOT

4.1 Fundamentals of Robot Kinematics

The acupuncture robot uses a UR5e six-degree-of-freedom robotic arm. The origin of the base coordinate system is defined at the center of the lower surface of the base, and the tool coordinate system is set at the acupuncture end. Kinematic analysis adopts the D-H parameter method to describe the spatial relationship between adjacent links. A homogeneous transformation matrix from the base to the end is established using link length, link twist, link offset, and joint angle. Forward kinematics is used to calculate the end pose from joint variables, while inverse kinematics is used to calculate the joint angles from a known target acupoint pose. The original study derived the inverse solution of the UR5e using an analytical method and analyzed possible singular positions of the shoulder, elbow, and wrist joints, see Table 4.

Table 4 D-H Parameters of the Acupuncture Robot

Link No.	θi	αi	ai/mm	di/mm
1	θ1	π/2	0	**89.2**
2	θ2	0	-425	0
3	θ3	0	-392	0
4	θ4	0	0	**109.3**
5	θ5	π/2	0	**94.75**
6	θ6	0	300	**182.5**

4.2 Improved APF-RRT* Algorithm

The RRT algorithm quickly searches for feasible paths in high-dimensional space through random sampling and tree expansion. On this basis, RRT* adds parent-node reselection and rewiring mechanisms so that path cost gradually improves with iteration [9]. However, in continuous acupuncture needling, traditional RRT* still has problems such as strong sampling randomness, slow convergence, and redundant initial paths. The improved APF-RRT* algorithm proposed in the original study combines multiple strategies: bidirectional RRT* grows random trees from the start and target points to improve initial path-search efficiency; the goal-bias strategy directly selects the target point with a certain probability during sampling to enhance directionality; and Informed-RRT* limits the sampling space to an ellipsoidal region with the start and target as foci after an initial path is obtained, reducing invalid sampling [10].
The artificial potential field method establishes an attractive field for the target point and a repulsive field for obstacles, so that the expansion direction of new nodes is affected by the target, random point, and obstacles at the same time [11]. To avoid the local-minimum and goal-unreachable problems of traditional artificial potential fields, the algorithm introduces a target-distance term into the repulsive-force function and incorporates both target-point attraction and random-point attraction into the expansion-direction calculation. The dynamic adaptive step size adjusts the expansion length according to the current node's attraction, repulsion, and obstacle distance: a larger step size is used in areas far from obstacles to improve search speed, while a smaller step size is used near obstacles to ensure safety. Finally, path pruning removes redundant nodes, making robotic-arm motion smoother.

4.3 Simulation Comparison Results

Simulation experiments were conducted in both two-dimensional and three-dimensional environments. The 2D environment covered 600 mm x 700 mm, with the start point at (50, 50) and the target point at (550, 640). The 3D environment covered 600 mm x 700 mm x 500 mm, with the start point at (50, 50, 100) and the target point at (550, 640, 100). The obstacles had a height of 200 mm and a radius of 20 mm, and the maximum number of nodes was 2000. Each of the three algorithms was simulated 10 times, and the average results were taken, see Figure 4.

Table 5 Comparison of Trajectory Planning Simulation Results

Environment	Algorithm	Running time/s	Path cost/mm
2D environment	RRT*	**44.702**	**792.702**
2D environment	APF-RRT*	**29.798**	**775.796**
2D environment	Improved APF-RRT*	**9.538**	**750.477**
3D environment	RRT*	**84.594**	**933.077**
3D environment	APF-RRT*	**45.591**	**877.434**
3D environment	Improved APF-RRT*	**32.796**	**822.8568**

Figure 4 Comparison of Trajectory Planning Running Time

As shown in Table 5, in the 2D environment, the improved APF-RRT* has a running time of 9.538 s, which is lower than 44.702 s for RRT* and 29.798 s for APF-RRT*. Its path cost is 750.477 mm, which is also lower than those of the other two algorithms. In the 3D environment, the improved algorithm has a running time of 32.796 s and a path cost of 822.8568 mm, again outperforming RRT* and APF-RRT*. These results indicate that the improved strategy can reduce blind sampling, improve search efficiency, and obtain a shorter and more executable path in an obstacle environment.

5 SYSTEM CALIBRATION AND NEEDLING EXPERIMENTS

5.1 Experimental System and Calibration

The experimental platform consists of a personal computer, a Kinect V2 camera, a UR5e six-degree-of-freedom robotic arm, an end-effector, and a simulated human head [14,15]. The Kinect V2 camera simultaneously acquires color and

depth images, while the upper computer runs acupoint recognition, three-dimensional reconstruction, and trajectory planning programs and sends motion commands to the robot controller through TCP/IP. To convert acupoints in two-dimensional images into executable three-dimensional coordinates in the robot coordinate system, the system first performs camera intrinsic calibration and then hand-eye calibration. The intrinsic calibration uses an 8 x 8 black-white checkerboard calibration plate with 63 effective inner corners and a square size of 25 mm x 25 mm. The hand-eye calibration uses an eye-to-hand configuration. The calibration plate is fixed at the robotic-arm end, and 18 different poses are collected to calculate the transformation between the Kinect V2 camera coordinate system and the robot base coordinate system [16,17].

The color image resolution is 1920 x 1080, and the standard depth image resolution of Kinect V2 is 512 x 424 [14]. Because the two images differ in imaging principle and physical position, RGB-D registration is required. In the registration process, depth pixels are first back-projected into the depth-camera coordinate system according to the intrinsic parameters, then transformed to the color-camera coordinate system through a rigid transformation, and finally projected onto the color image plane. The aligned depth image can be used to reconstruct the facial three-dimensional point cloud. For each acupoint pixel detected by the model, the system reads the corresponding depth value and back-projects it to obtain three-dimensional coordinates; if the depth at this position is missing, Gaussian-weighted interpolation is used to estimate depth within the neighborhood. The point is then transformed to the robot base coordinate system through the hand-eye calibration matrix to provide the target point for robotic needling.

5.2 Trajectory Planning and Needling Experiments

In the trajectory planning experiment, a black pen was mounted at the end of the UR5e robotic arm instead of the end-effector, and already inserted acupuncture needles on the simulated human head were used as obstacles. After the Kinect V2 collected facial images, the host computer reconstructed the facial and needle point clouds, and the improved APF-RRT* algorithm generated an obstacle-avoidance path and controlled the robotic arm to move along the path to the target point. During the experiment, the robotic arm avoided the inserted needles and reached the target position smoothly, and no obvious abrupt changes appeared in the joint angles, indicating that the planning results were executable, see Figure 5.

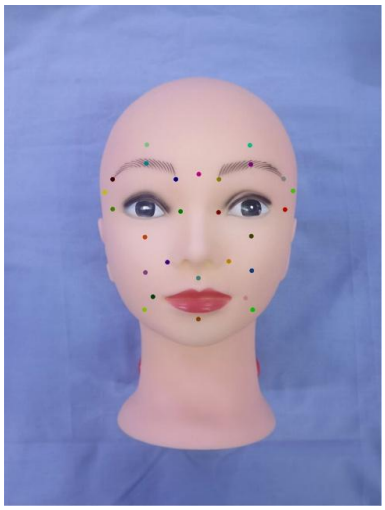

Figure 5 Acupoint Positions and Needling Results

In the comprehensive needling experiment, the system first used the Kinect V2 to acquire color and depth images of the simulated human head. After facial 3D point-cloud reconstruction and YOLOv8-SDI-C3Ghost acupoint prediction, the spatial coordinates of the target acupoints were obtained. Path planning was then performed by combining the facial point cloud, acupoint coordinates, and the pose of already inserted needles. Finally, the robotic arm drove the end-effector to complete needling. To evaluate localization accuracy, the experiment was repeated 10 times, and the deviations between the actual and theoretical needling points were measured. The statistical results of the original study show that the average errors in the X, Y, and Z directions were 0.54 mm, 0.50 mm, and 0.66 mm, respectively, with corresponding standard deviations of 0.25 mm, 0.41 mm, and 0.23 mm, see Table 6 and Figure 6.

Table 6 Statistics of Acupoint Needling Errors

Index	X direction/mm	Y direction/mm	Z direction/mm
Mean error	**0.54**	**0.50**	**0.66**
Standard deviation	**0.25**	**0.41**	**0.23**

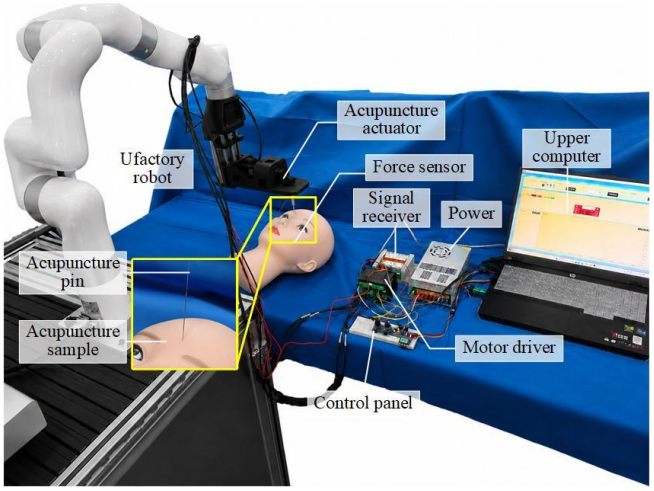

Figure 6 Schematic Diagram of the Acupuncture Robot System

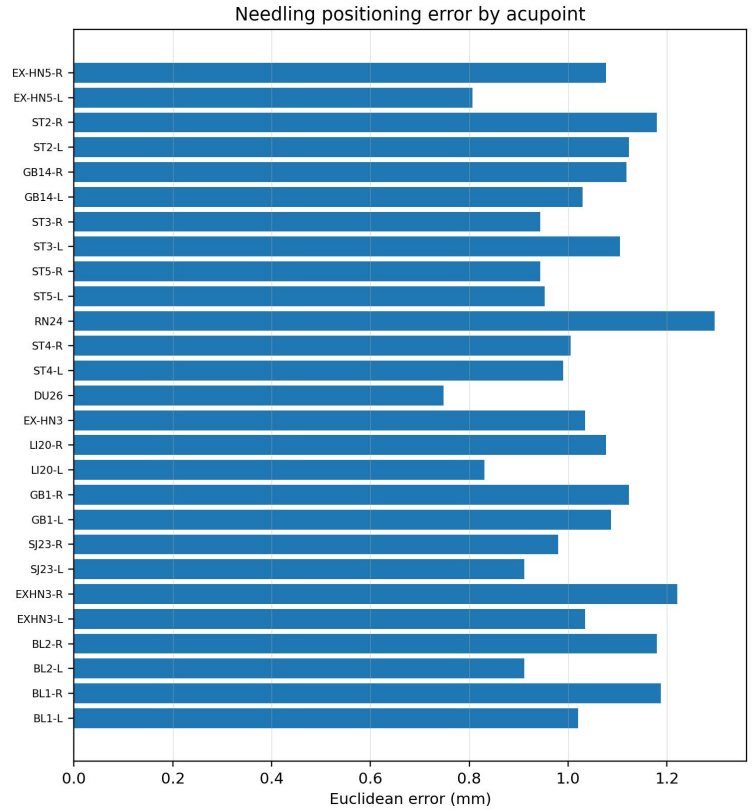

Figure 7 Spatial Error Distance of Needling for Each Acupoint

As shown in Figure 7, the spatial error of each acupoint is controlled within 1.3 mm. Combined with the mean error and standard deviation, the system shows good repeatability and stability in facial needling tasks on the simulated human head, and can meet the clinical accuracy requirement set in the original study. It should be noted that the current experimental object is a simulated human head and does not yet cover factors such as real human soft-tissue deformation, respiratory motion, patient micro-movement, and complex facial expressions. Future work should further improve safety by combining force perception and online correction.

6 CONCLUSION

This paper condenses the original study around vision-based acupoint localization and automatic needling for an acupuncture robot. Starting from clinical acupuncture operation, lifting-thrusting and twirling manipulations are converted into linear and rotational motions executable by a robot. An end-effector integrating needle changing, holding, and twirling functions is designed, and the structural feasibility is verified through synchronous-belt transmission calculation and finite element simulation.

For facial acupoint recognition, the paper uses YOLOv8 as the baseline and introduces C3Ghost and SDI modules to build a lightweight detection model. Experiments on the self-built dataset show that the improved model significantly

reduces the number of parameters and model size while maintaining high accuracy. Its mAP50-95 is 99.26%, and the model size is 5.5 MB, making it suitable for further deployment on edge devices.

For trajectory planning, the paper proposes an improved APF-RRT* algorithm to solve the problem that already inserted needles become obstacles during continuous multi-acupoint needling. Both 2D and 3D simulations show that the algorithm outperforms RRT* and APF-RRT* in planning time and path cost. In system experiments, camera calibration, hand-eye calibration, 3D reconstruction, trajectory planning, and needling verification are completed using the Kinect V2, UR5e robotic arm, and simulated human head. The localization errors in ten repeated experiments remain stable within the millimeter range.

Future research can be carried out in three directions. First, miniature force/torque sensors can be integrated into the end-effector, and compliant control can be introduced to reduce the influence of tissue deformation and patient micro-movement on needling accuracy. Second, the acupoint recognition model can be further optimized to improve its robustness under expression changes, occlusion, and complex lighting conditions. Third, verification in real clinical scenarios can be expanded to form a complete acupuncture robot operating procedure that includes safety boundaries, physician confirmation, and abnormal termination mechanisms.

COMPETINGINTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This study was partially supported by the Undergraduate Innovation Training Program of University of Science and Technology Liaoning and the Doctoral Research Startup Project of Liaoning Provincial Natural Science Foundation (Project No.: 2025-BS-0376).

REFERENCES

[1] World Health Organization. Acupuncture: review and analysis of reports on controlled clinical trials. Geneva: World Health Organization, 2002: 1-87.
[2] Vickers AJ, Vertosick EA, Lewith G, et al. Acupuncture for chronic pain: update of an individual patient data meta-analysis. J. Pain, 2018, 19(5): 455-474.
[3] Lan KC, Lee CY, Lee GS, et al. An initial study on automated acupoint positioning for laser acupuncture. Evid.-Based Complement. Altern. Med., 2022: 8997051.
[4] Zhang M, Schulze J P, Zhang D. FaceAtlasAR: atlas of facial acupuncture points in augmented reality. Comput. Sci. Inf. Technol., 2021, 11(7): 1-11.
[5] Yuan Z, Shao P, Li J, et al. YOLOv8-ACU: improved YOLOv8-pose for facial acupoint detection. Front. Neurorobot., 2024, 18: 1355857.
[6] Malekroodi HS, Seo SD, Choi J, et al. Real-time location of acupuncture points based on anatomical landmarks and pose estimation models. Front. Neurorobot., 2024, 18: 1484038.
[7] Wang H, Liu L, Wang Y, et al. Hand acupuncture point localization method based on a dual-attention mechanism and cascade network model. Biomed. Opt. Express, 2023, 14(11): 5965-5978.
[8] Liu YB, Qin JH, Zeng GF. Back acupoint location method based on prior information and deep learning. Int. J. Numer. Method Biomed. Eng., 2023, 39(12): e3776.
[9] Karaman S, Frazzoli E. Sampling-based algorithms for optimal motion planning. Int. J. Robot. Res., 2011, 30(7): 846-894.
[10] Gammell JD, Srinivasa SS, Barfoot TD. Informed RRT*: optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2014: 2997-3004.
[11] Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. Int. J. Robot. Res., 1986, 5(1): 90-98.
[12] Terven J, Cordova-Esparza DM, Romero-Gonzalez JA. A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Mach. Learn. Knowl. Extr., 2023, 5(4): 1680-1716.
[13] Han K, Wang Y, Tian Q, et al. GhostNet: more features from cheap operations. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020: 1580-1589.
[14] Pagliari D, Pinto L. Calibration of Kinect for Xbox One and comparison between the two generations of Microsoft sensors. Sensors, 2015, 15(11): 27569-27589.
[15] Universal Robots A/S. UR5e e-Series datasheet. Odense: Universal Robots A/S, 2024.
[16] Zhang Z. A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell., 2000, 22(11): 1330-1334.
[17] Tsai RY, Lenz RK. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. IEEE Trans. Robot. Autom., 1989, 5(3): 345-358.