

Smart Pattern Generation on Programmable Dielectrophoresis Array Chip for Single Particle Manipulation

Yu-Hsiang Wang, Wen-Yue Lin, Lin-Hung Lai, Chen-Yi Lee
Institute of Electronics, National Yang Ming Chiao Tung University, Hsinchu, Taiwan
E-mail: {david998869.ee12, cylee}@nycu.edu.tw

Abstract—Dielectrophoresis (DEP) is powerful for manipulating biological cells. However, single cell manipulation is usually time consuming and skill needed. This paper presents a system that integrates AI for real-time image recognition with a programmable dielectrophoresis (DEP) array chip for automated particle manipulation. The system comprises a DEP chip, an FPGA, a computer, a microscope, and a server. The YOLO v8 model is used to detect particle positions within microscope images and generate DEP manipulation patterns. The system utilizes a Breadth-First Search (BFS) algorithm for path planning, ensuring collision-free movement of particles within a grid structure. Experimental results demonstrated the system’s effectiveness in manipulating 20 μm polystyrene particles with a success rate of over 90%. This system offers a significant advancement in automated DEP-based manipulation, providing precise control at micro scales with high computational efficiency.

Index Terms—Dielectrophoresis, DEP, YOLO v8, Image Recognition, Particle Manipulate

I. INTRODUCTION

The precise control of micro and nano-sized particles has driven advancements in dielectrophoresis (DEP) technology in recent years. DEP is the phenomenon where a dielectric particle experiences a force when subjected to a nonuniform electric field [1]. The direction of the DEP force is determined by the relative polarizability of the particle compared to the surrounding medium [2]. The DEP force can be positive or negative. Positive DEP (pDEP) moves particles towards regions of higher electric field, whereas negative DEP (nDEP) moves them to regions with lower electric field. Nowadays, DEP is widely used in applications such as drug delivery [3], cell sorting [4], and cell characterization [5].

Single particle or cell manipulation using DEP has applications in fields such as cell behavior control and analysis [6], studying cell interactions [7], and modulating cell aggregation [8]–[10]. Automating these processes enhances scalability and reduces the dependency on manual labor, making them feasible for broader applications in both research and industry.

One major challenge in utilizing DEP is designing a system to efficiently control the programmable electrode arrays to generate dynamic electric fields for particle manipulation [11]. Conventional methods as shown in Fig 1(b), rely on human eye

This work was supported by National Science and Technology Council (NSTC) under Project 113-2321-B-A49-013 and in part by chip fabrication services from Taiwan Semiconductor Research Institute, Taiwan, R. O. C.

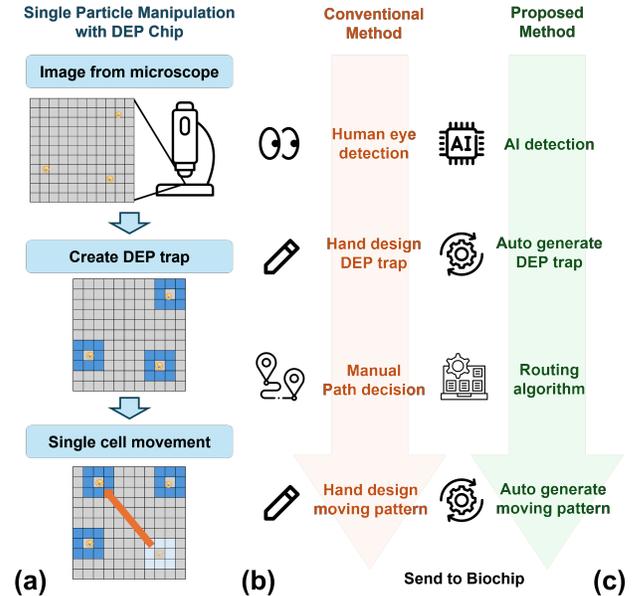


Fig. 1. (a) The experiment flow between our previous approach (b) and new method (c) including using AI image recognition to accelerate the particle control speed.

detection, are limited by prolonged detection times, making real-time updates difficult. Additionally, manually generating manipulation patterns requires significant expertise and time, which is impractical for rapid and automated control.

Recent advances in deep learning, particularly in object detection, provide promising solutions to these challenges. YOLO (You Only Look Once), has proven highly effective due to its balance of speed and accuracy [12]. YOLO v8, features improvements in architecture and training methods, enhancing real-time detection capabilities [13]. By integrating YOLO v8 with DEP systems, it is possible to automate electrode control pattern generation, improving efficiency and accuracy in manipulating micro and nano-scale particles.

This work presents a system integrating the YOLO v8 model for real-time object detection with a programmable DEP array chip. The system automates DEP control pattern generation, enabling precise and rapid particle manipulation without human intervention. The architecture includes an pro-

programmable DEP chip, a FPGA for communicate, a microscope for imaging, and a computer-server setup for processing and control.

Our approach as shown in Fig 1(c), addresses the limitations of manual pattern generation and slow position sensing by leveraging YOLO v8’s real-time detection capabilities, combined with automated path planning using a Breadth-First Search (BFS) algorithm. This ensures collision-free movement of particles within the DEP chip’s 128×128 electrode array, achieving high computational efficiency and precise control over particle trajectories.

II. SYSTEM ARCHITECTURE

A. System Overview

The system consists of five main components: the chip, the FPGA, the computer, the microscope, and the server, shown as Fig 2(a). The chip serves as the experimental platform, where control signals are generated by an FPGA according to commands issued by the computer module. These signals are used to create DEP manipulation patterns on the chip. The microscope captures images of the chip, while the server module processes these images using YOLO v8 for object detection. The server then returns the detection results to the computer module.

B. Programmable DEP Chip Array

The chip used in this experiment is a programmable DEP array chip with 128×128 electrodes. The chip generates non-uniform electric fields by applying two sine waves with different phases to the electrodes, producing a nDEP force to move the particles. All forces are generated along the edges of the electrodes with opposing phases. By surrounding one phase with the other, a hollow DEP trap well is formed, effectively confining the movement of the particles. Gradually moving the position of this DEP trap well allows precise control of particle movement to the desired location.

C. Hardware Equipment

The personal computer captures images through a microscope and transmits the image data to the server using a custom Python GUI. The detection results from the server are then processed on the personal computer to generate a control pattern, which is transmitted as UART format signals to the FPGA. All operations are completed through a single Python GUI, and the system only requires a standard personal computer to operate.

In our system, the FPGA serves as a communication bridge between the PC and the DEP chip. The PC transmits a series of commands using the UART protocol, which the FPGA receives and converts into scan chain signals required by the DEP chip for operation.

III. PATTERN GENERATION

A. Objection Detection

For particle detection, we use the pre-trained YOLO v8 model built on the Common Objects in Context (COCO)

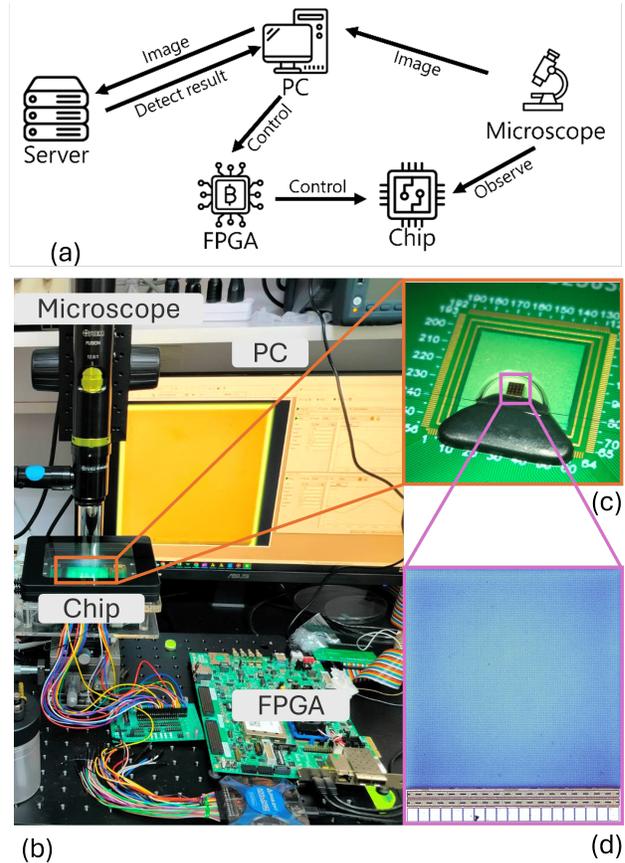


Fig. 2. (a) The system architecture over view. (b) Picture of the system. (c) Close look of the DEP chip (d) DEP Chip under microscope view.

dataset, utilized 65 microscope images of PS particles, applying rotations of 90° and 180° to obtain a total of 195 images. During training, the images were processed by cropping them to a unified size of 800×800 pixels, adding noise to 1.97% of the pixels, and adjusting saturation and brightness by $\pm 30\%$ and $\pm 25\%$, respectively, to enhance data augmentation. This resulted in 402 images, of which 302 were used as training data, 72 as validation data, and 28 as testing data. The model was trained with a batch size of 16 for a total of 1000 epochs, with early stopping applied. The final performance of particle detection achieved an mAP of 96.2%, precision of 93.2%, and recall of 91.3%.

After training, the model weights were packaged and deployed on a server, allowing users within the same network domain to upload images or stream live images using a webcam. The server processes these images and quickly returns detection results to the user via an HTTP POST request. Performance tests showed that the system could support up to 30 FPS video streams with an average response time of less than 50 ms, demonstrating excellent real-time detection capabilities.

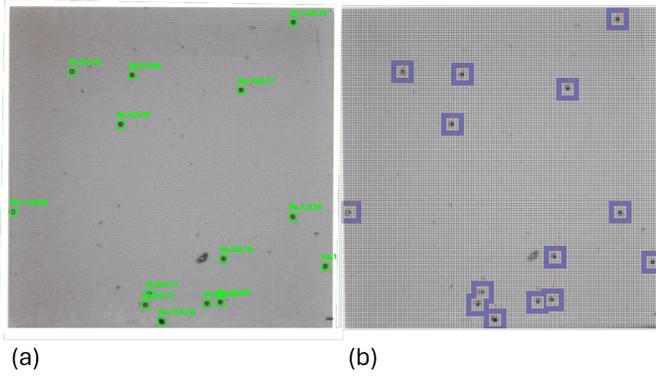


Fig. 3. (a) A detection result of particle image and (b) the pattern generate to automatically trap the particles.

B. Calibration

Image calibration affects the positional error between the detected particle positions and their actual locations on the chip. We adopt a manual calibration method to reduce this error. The user needs to aligns two crosshairs with the top-left and bottom-right corners of the chip displayed on the screen to complete the calibration. Through measurements, the calibration error does not exceed $5\mu\text{m}$, which is an acceptable margin. This error can be compensated by adjusting the size of the hollow section in the generated pattern.

C. Pattern and Routing Method

After acquiring the particle positions, the system first activates a circular DEP trap around each particle. This is done to prevent particle displacement away from their initially recognized positions. The size and internal hollow of each DEP trap can be adjusted according to the size of the particles being manipulated. Once the particles are secured, any selected particle can be moved to a target position as specified by the user.

For particle movement, Algorithm 1 is used, taking into account the DEP traps around each particle, which act as obstacles. Each moving particle must also have a surrounding DEP trap to push it forward. Thus, in the implementation, the DEP chip can be visualized as a 128×128 grid, where obstacles are represented by the DEP traps of both the moving and stationary particles, ensuring that they do not interfere with each other. The BFS algorithm is used to identify a feasible route, minimizing the pathfinding complexity. If there is no available path that prevents interference between particles, the user is notified that a valid path cannot be found. In such cases, the user can reduce the size of the DEP traps to create more space or first move particles obstructing the path.

IV. EXPERIMENT RESULTS

A. Particle Image Detection

As shown in Fig 3(a), YOLO v8 was employed to detect particles within the microscope images. In most of the cases we use confidence 0.5 and overlap 0.5 as the detection

Algorithm 1 Particle Movement Pathfinding Using BFS

Require: DEP Grid G of size 128×128 , Initial Position P_{init} , Target Position P_{target} , DEP Trap Configuration T

Ensure: Valid Path to Move Particle or Notification of No Path

```

1: Initialize BFS queue with  $P_{init}$ 
2: Set all DEP traps as obstacles in  $G$ 
3: while queue is not empty do
4:   Dequeue current position  $P_{curr}$ 
5:   if  $P_{curr} == P_{target}$  then
6:     return path from  $P_{init}$  to  $P_{target}$ 
7:   end if
8:   Mark  $P_{curr}$  as visited
9:   for each neighbor  $P_{neighbor}$  of  $P_{curr}$  do
10:    if  $P_{neighbor}$  is not an obstacle and  $P_{neighbor}$  is not
        visited then
11:      Enqueue  $P_{neighbor}$ 
12:      Mark  $P_{neighbor}$  as visited
13:    end if
14:  end for
15: end while
16: Notify user: "No valid path available"

```

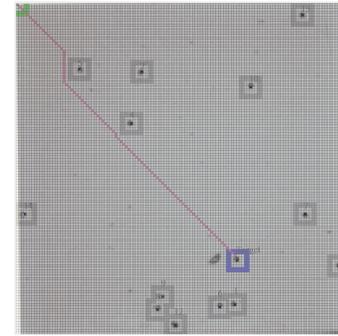


Fig. 4. A path finding result for moving the particle from center to top left corner than out of the chip.

parameter. Based on the detection results, the system automatically generated corresponding DEP traps around the identified particles as shown in Fig 3(b). It can be observed that the actual positions of the particles align very closely with the locations of the DEP traps, the close alignment between the particle positions and DEP traps ensures effective containment and manipulation.

To further evaluate the detection accuracy, we compared the YOLO v8-detected positions with manually annotated positions. The average AI detection error compared to manual was calculated to be within $5\mu\text{m}$, which is well within the acceptable limits for precise manipulation of nano-scale particles, since the trap well we set have at least $40\mu\text{m}$ hollow for the $20\mu\text{m}$ particle to be trapped. The success rate of accurate DEP trap placement based on the detected positions was over 95%.

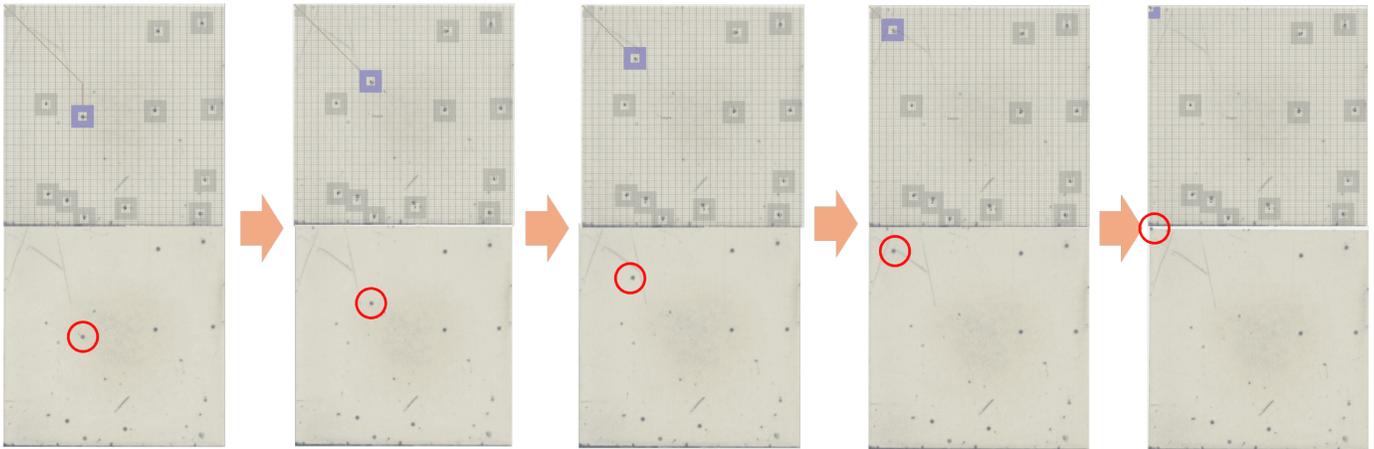


Fig. 5. A series of images from left to right shows moving PS particle from center to top left corner than remove from the chip , with the system view (top) showing the control pattern, and the microscope view (bottom) showing the PS particle.

B. Particle Selection and Path Planning

Upon successful particle detection, each particle is assigned a unique identifier by the system. Users are then able to select any particle and specify a target coordinate for movement. The coordinate system is defined such that the top-left corner is (0, 0) and the bottom-right corner is (127, 127). After selecting a target, the system uses a BFS algorithm to calculate and visualize the planned path from the initial to the target position, as shown in Fig 4.

To validate the path planning process, multiple scenarios were tested with varying numbers of particles and different initial and target positions. The system successfully generated collision-free paths in all tested cases. In situations where a valid path could not be found due to obstacles created by other particles' DEP traps, the system provided a notification, suggesting adjustments to the trap size or the removal of blocking particles. The BFS algorithm was found to be efficient, with an average pathfinding time of under 10 ms.

C. Particle Movement

In this experiment, polystyrene (PS) particles with a diameter of 20 μm were used. 1 MHz, 1.8 Vpp with a 180° phase difference sine waves were used as the DEP control wave. Fig 5 shows a particle being moved from its initial position to the specified target location.

The experimental results indicate that the particle successfully followed the planned path, and the DEP system effectively generated the necessary forces to move the particle as intended. The average time taken for each movement operation was approximately 30 seconds, depending on the distance and number of obstacles along the path. The stability of the DEP traps ensured that the particle did not deviate from its intended trajectory, even in the presence of minor external disturbances.

The effectiveness of particle movement was further evaluated by repeating the movement process with different starting and ending points 50 times. The success rate of moving the particle to the desired position was over 90%.

TABLE I
A COMPARE TABLE OF EACH OPERATION'S COST TIME

	Particle detection	DEP trap generation	Path planning	Moving pattern generation
Conventional method	1~3s	>10min	>1min	>10min
Proposed method	<0.05s	<0.01s	<0.1s	<1s

D. Result Comparison

Table I presents a comparison of the time taken by each operation in the conventional and proposed methods. The conventional approach required manual effort for particle detection, DEP trap generation, path planning, and moving pattern generation, leading to a significant amount of time per operation. In our previous experiment, the average time from detection to moving one particle is about half a hour. The integration of particle detection and an automated BFS algorithm for path planning drastically reduced the time required for each stage of the manipulation process, with less than 10s to finish the process.

V. CONCLUSION

This paper presents a system integrating YOLO v8 for real-time image recognition with a programmable DEP array chip for automated single particle manipulation. The experimental results demonstrate the system's effectiveness, with over 90% success in accurately manipulating 20 μm polystyrene particles. The use of YOLO v8 significantly enhances the speed of particle detection and the DEP trap pattern generation time, while the BFS algorithm ensures efficient and reliable path planning for particle movement and generation of the moving pattern. The system supports real-time detection at 30 FPS with a response time of under 50 ms. Future work will focus on improving the scalability of the system for multi-cell manipulation and optimizing DEP trap configurations for enhanced particle control.

REFERENCES

- [1] A. Mauro, "Dielectrophoresis: the behavior of neutral matter in nonuniform electric fields," 1980.
- [2] T. Ghomian and J. Hihath, "Review of dielectrophoretic manipulation of micro and nanomaterials: Fundamentals, recent developments, and challenges," *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 1, pp. 27–41, 2022.
- [3] G. Nabovati, E. Ghafar-Zadeh, A. Letourneau, and M. Sawan, "Smart cell culture monitoring and drug test platform using cmos capacitive sensor array," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 4, pp. 1094–1104, 2018.
- [4] Y.-S. Chan, T.-T. Hsu, Y.-P. Chen, H.-M. Wu, Y.-M. Wang, and C.-Y. Lee, "A traveling-wave dielectrophoresis bio-chip for cell manipulation in standard cmos process," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 1582–1586, 2021.
- [5] Q. Hu, Z. Wang, L. Shen, and G. Zhao, "Label-free and noninvasive single-cell characterization for the viscoelastic properties of cryopreserved human red blood cells using a dielectrophoresis-on-a-chip approach," *Analytical Chemistry*, vol. 94, no. 28, pp. 10245–10255, 2022.
- [6] H. Cha, H. Fallahi, Y. Dai, D. Yuan, H. An, N.-T. Nguyen, and J. Zhang, "Multiphysics microfluidics for cell manipulation and separation: a review," *Lab on a Chip*, vol. 22, no. 3, pp. 423–444, 2022.
- [7] K. Huang, B. Lu, J. Lai, and H. K. H. Chu, "Microchip system for patterning cells on different substrates via negative dielectrophoresis," *IEEE transactions on biomedical circuits and systems*, vol. 13, no. 5, pp. 1063–1074, 2019.
- [8] F. G. Torizal, K. Kimura, I. Horiguchi, and Y. Sakai, "Size-dependent hepatic differentiation of human induced pluripotent stem cells spheroid in suspension culture," *Regenerative Therapy*, vol. 12, pp. 66–73, 2019.
- [9] S. Islam, J. Parker, B. C. Dash, and H. C. Hsia, "Human ipsc-vascular smooth muscle cell spheroids demonstrate size-dependent alterations in cellular viability and secretory function," *Journal of Biomedical Materials Research Part A*, vol. 110, no. 11, pp. 1813–1823, 2022.
- [10] M. S. Bogacheva, R. Harjumäki, E. Flander, A. Taalas, M. A. Bystrakova, M. Yliperttula, X. Xiang, A. W. Leung, and Y.-R. Lou, "Differentiation of human pluripotent stem cells into definitive endoderm cells in various flexible three-dimensional cell culture systems: Possibilities and limitations," *Frontiers in Cell and Developmental Biology*, vol. 9, p. 726499, 2021.
- [11] W.-Y. Lin, L.-H. Lai, Y.-W. Lin, and C.-Y. Lee, "A programmable cmos dielectrophoresis array chip with 128×128 electrodes for cell manipulation," in *2024 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2024, pp. 1–5.
- [12] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of yolo algorithm developments," *Procedia computer science*, vol. 199, pp. 1066–1073, 2022.
- [13] M. Sohan, T. Sai Ram, R. Reddy, and C. Venkata, "A review on yolov8 and its advancements," in *International Conference on Data Intelligence and Cognitive Informatics*. Springer, 2024, pp. 529–545.