

Penerapan Algoritma *Deep Learning* pada Robot Deteksi Botol

Agus Ismangil¹, Abel Gandhy², Asep Saepulrohman¹, Gustian Rama Putra¹, Muhamamd Bintang Drajar¹, Muhammad Taufiq¹, dan Arrazy Azha¹

¹Program Studi Ilmu Komputer, FMIPA, Universitas Pakuan

²Program Studi Manajemen, FEB, Universitas Pakuan

E-mail: {agusismangil, abelgandhy, asepsaepulrohman, gustianrama}@unpak.ac.id
{muhammadbintang14123, olewenusa, arazyazhar}@gmail.com

Abstrak

Dalam penelitian ini, algoritma *deep learning* digunakan untuk robot otonom yang mendeteksi botol. Pengembangan robot yang mampu berinteraksi dengan lingkungan bergantung pada kemampuan robot untuk mendeteksi objek. Dalam penelitian ini, algoritma *deep learning* digunakan untuk mengidentifikasi dan mendeteksi botol dalam berbagai kondisi pencahayaan dan sudut pengambilan gambar. Algoritma ini terutama menggunakan model berbasis jaringan saraf tiruan (*neural networks*), seperti *Convolutional Neural Networks* (CNN). Studi ini menunjukkan bahwa penggunaan model *deep learning* meningkatkan akurasi deteksi botol hingga 95%. Ini menunjukkan bahwa model ini dapat digunakan dengan baik dalam sistem robotika kontemporer.

Kata kunci: *Deep Learning*, Robot Deteksi, CNN, Botol, Jaringan Saraf Tiruan

Pendahuluan

Dalam beberapa dekade terakhir, robotika telah berkembang pesat, terutama dengan menambahkan kecerdasan buatan (AI) [1] ke dalam sistem robotik. Salah satu aplikasi AI yang semakin populer adalah algoritma *deep learning* [2], yang memungkinkan robot memahami dan berinteraksi dengan lingkungannya [3], [4], [5]. Salah satu fungsi penting dalam robotika adalah kemampuan deteksi objek, yang memungkinkan robot untuk mengenali, mengklasifikasikan, dan memproses berbagai objek di sekitarnya [6], [7].

Deteksi botol adalah salah satu fitur penting dalam pengelolaan sampah dan industri daur ulang. Untuk proses pemilahan dan daur ulang yang efektif, botol plastik dan kaca harus diidentifikasi secara cepat dan akurat [8]. Teknologi konvensional yang berbasis sensor seringkali memiliki kelemahan, terutama dalam lingkungan tanpa pencahayaan atau ketika objek tidak terlihat jelas. Oleh karena itu, penerapan algoritma *deep learning* [9], khususnya *Convolutional Neural Networks* (CNN), menjadi solusi yang menjanjikan untuk meningkatkan performa robot deteksi [10], [11].

Bagian dari *machine learning* yang dikenal sebagai *deep learning* menggunakan jaringan saraf tiruan (*neural networks*) berlapis-lapis untuk mengekstrak elemen penting dari data, seperti gam-

bar atau video [12], [13]. Keunggulan dalam pengolahan gambar dan deteksi objek telah ditunjukkan oleh CNN, salah satu arsitektur *deep learning* yang paling populer. Algoritma ini memiliki kemampuan untuk mempelajari aspek visual yang kompleks dari gambar, seperti bentuk, tekstur, dan warna. Kemudian, algoritma ini digunakan untuk mengklasifikasikan objek dengan sangat akurat [14].

Dalam penelitian yang dilakukan oleh LeCun et al., para peneliti meneliti bagaimana *Convolutional Neural Networks* (CNN) telah menjadi bagian penting dari proses yang dilakukan robot otonom, termasuk pengenalan objek, segmentasi gambar, dan navigasi visual. Dengan CNN, robot dapat mengenali dan memahami kondisi lingkungan nyata seperti perubahan pencahayaan dan *occlusion*. Ini adalah hal yang sebelumnya sulit dilakukan dengan teknik konvensional.

Studi tambahan yang dilakukan oleh Zhu et al. melihat penggunaan *Recurrent Neural Networks* (RNN) dan *Long Short-Term Memory* (LSTM) untuk memahami urutan data dari sensor waktunya, seperti sensor lidar atau kamera. Metode ini memungkinkan robot untuk memiliki pemahaman yang lebih baik tentang pergerakan dinamis dan pola lingkungan yang terus berubah.

Tujuan penelitian ini adalah untuk membuat sistem robot deteksi botol berbasis *deep learning* yang menggunakan CNN. Dengan menggunakan

CNN, sistem ini diharapkan dapat membantu pemilahan dan daur ulang limbah dengan mendeteksi botol dalam berbagai kondisi pencahayaan, sudut pandang, dan variasi jenis botol. Selain itu, sistem ini diharapkan dapat beroperasi secara *real-time*. Selain itu, penelitian ini akan menyelidiki bagaimana penerapan model *deep learning* dapat diintegrasikan ke dalam platform robotika otonom, yang dapat bekerja secara mandiri tanpa bantuan manusia.

Penelitian ini menggunakan algoritma *deep learning* untuk membuat sistem robotik untuk mendeteksi botol. Tujuan utama penelitian adalah untuk membuat sistem yang dapat mengidentifikasi botol dari berbagai sudut dan dalam berbagai kondisi pencahayaan, memungkinkan robot untuk beroperasi secara optimal dalam lingkungan yang dinamis.

Metode Penelitian

Arsitektur Model *Deep Learning*

Model yang digunakan dalam penelitian ini adalah *Convolutional Neural Networks* (CNN), yang terkenal dengan kemampuan pengenalan pola dan klasifikasi gambar, adalah model yang digunakan dalam penelitian ini. CNN sangat baik dalam pengolahan citra, terutama mendeteksi sifat visual yang kompleks seperti bentuk dan tekstur botol. Karena dapat mempelajari fitur penting dari data citra secara otomatis, CNN juga dapat menangani masalah deteksi objek [15], [16]. Studi ini menggunakan arsitektur CNN yang terdiri dari beberapa lapisan, seperti:

Lapisan Konvolusi (*Convolutional Layer*)

Lapisan konvolusi bertugas untuk mengekstrak fitur dari input gambar. Operasi konvolusi dilakukan dengan cara menggeser filter (*kernel*) di seluruh gambar *input*. Untuk operasi konvolusi [17], [18], [19]:

$$Z[i, j] = \sum_m \sum_n X[i + m, j + n].K[m, n] + b \quad (1)$$

- $Z[i, j]$: *Output* dari konvolusi pada posisi i, j
- X : *Input* gambar atau fitur (misal matriks piksel)
- K : *Kernel*/filter (misalnya matriks 3x3 atau 5x5)
- b : Bias
- i, j : Posisi *output*
- m, n : Indeks pada *kernel*/filter

Dimensi *output* dari operasi konvolusi dapat dihitung dengan rumus [20]:

$$\begin{aligned} O_w &= \frac{W - F + 2P}{S} + 1 \\ O_h &= \frac{H - F + 2P}{S} + 1 \end{aligned} \quad (2)$$

- O_w, O_h : Dimensi *output* (lebar dan tinggi)
- W, H : Dimensi *input* (lebar dan tinggi)
- F : Ukuran filter (*kernel size*)
- P : *Padding* (jumlah piksel yang ditambahkan di tepi gambar)
- S : *Stride* (jumlah langkah filter bergeser pada gambar)

Lapisan Pooling (*Pooling Layer*)

Pooling berfungsi untuk mengurangi dimensi dari fitur yang dihasilkan pada lapisan konvolusi, umumnya menggunakan *Max Pooling* atau *Average Pooling*. Max Pooling [21], [22]:

$$Z[i, j] = (X[i, j], X[i + 1, j], X[i, j + 1], X[i + 1, j + 1]) \quad (3)$$

Rumus ini menunjukkan bahwa *Max Pooling* mengambil nilai maksimum dari matriks kecil (misalnya 2x2) di dalam fitur *input*. Dimensi *output* setelah *pooling*:

$$\begin{aligned} O_w &= \frac{W}{S_p} \\ O_h &= \frac{H}{S_p} \end{aligned} \quad (4)$$

- O_w, O_h : Dimensi *output* (lebar dan tinggi)
- W, H : Dimensi *input*
- S_p : Ukuran *stide* untuk *pooling* (misal 2x2 menghasilkan $S_p=2$)

Lapisan Aktivasi (*Activation Layer*)

Lapisan aktivasi menggunakan fungsi non-linear untuk menambahkan kemampuan model dalam menangani data yang kompleks. Salah satu fungsi aktivasi yang populer adalah ReLU (*Rectified Linear Unit*) [23]. Rumus ReLU:

$$f(x) = \max(0, x) \quad (5)$$

Fungsi ini mengubah nilai negatif menjadi 0 dan membiarkan nilai positif tetap seperti aslinya.

Lapisan Fully Connected (*Fully Connected Layer*) Data set

Lapisan ini menghubungkan setiap neuron di satu lapisan dengan neuron di lapisan berikutnya. Setiap neuron menghitung nilai output sebagai kombinasi linear dari input [24]. Rumus dasar pada fully connected layer:

$$Z = W.X + b \tag{6}$$

- Z : *Output* dari neuron
- W : Bobot (*weight*) yang dipelajari selama pelatihan
- X : *Input*
- b : Bias

Softmax (*Output Layer*)

Lapisan *ouput* pada CNN umumnya menggunakan fungsi Softmax untuk klasifikasi multiklas [25]. Rumus Softmax:

$$S(Z_i) = \frac{e^{Z_i}}{\sum e^{Z_j}} \tag{7}$$

- $S(Z_i)$: Probabilitas kelas ke- i
- Z_i : *Output* dari neuron ke- i
- $\sum e^{Z_j}$: Penjumlahan dari semua eksponensial output untuk normalisasi arsitektur CNN.

Sebuah CNN untuk deteksi botol dengan arsitektur sebagai berikut:

1. *Input layer*: Gambar ukuran $224 \times 224 \times 3$ (RGB)
2. *Conv Layer*: Filter ukuran 3×3 , jumlah filter 32, *stride* 1, *padding* 1, *Output*: $224 \times 224 \times 32$
3. *Max Pooling Layer*: Ukuran *pooling* 2×2 , *stride* 2, *Output*: $112 \times 112 \times 32$
4. *Conv Layer*: Filter ukuran 3×3 , jumlah filter 64, *stride* 1, *padding* 1, *Output*: $112 \times 112 \times 64$
5. *Max Pooling Layer*: Ukuran *pooling* 2×2 , *stride* 2, *Output*: $56 \times 56 \times 64$
6. *Fully Connected Layer*: Neuron 512
7. *Output Layer*: Softmax untuk klasifikasi (misalnya 2 kelas: botol atau bukan botol) [26], [27].

Dataset yang digunakan terdiri dari ribuan gambar botol dalam berbagai situasi, seperti botol kosong, penuh, plastik, atau kaca, dan dengan berbagai tingkat pencahayaan. Untuk meningkatkan generalisasi model dan meningkatkan variasi data, teknik augmentasi gambar digunakan untuk mengolah set data ini [28]. Tujuan augmentasi data, seperti rotasi, *flip*, *cropping*, dan perubahan pencahayaan, adalah untuk meningkatkan variasi data dan mencegah overfitting, sehingga model dapat lebih umum dan sesuai dengan berbagai situasi di dunia nyata [29].

Setelah itu, gambar dari dataset diproses untuk memastikan bahwa data yang dimasukkan ke dalam model CNN memiliki format dan ukuran yang sama. Proses *preprocessing* terdiri dari:

- Normalisasi gambar: Nilai piksel gambar dinormalisasi agar berada di antara 0 dan 1.
- *Resizing*: Setiap gambar diubah ukurannya menjadi dimensi yang seragam, misalnya 224×224 piksel, untuk disesuaikan dengan input CNN.
- *Labeling*: Setiap gambar diberi label sesuai dengan klasifikasi mereka, yaitu botol atau bukan botol.

Pelatihan Model

Data yang sudah disiapkan digunakan untuk melatih model CNN. Algoritma Adam digunakan untuk melakukan optimalisasi, yang terkenal dengan kemampuan untuk mengonvergensi dengan cepat ke solusi optimal. Data pelatihan (80 persen) dan data pengujian (20 persen) dibagi untuk melakukan proses pelatihan.

Karena kemampuan untuk mempercepat konvergensi, algoritma optimasi Adam, yang populer dalam pelatihan model *deep learning*, digunakan untuk melakukan proses pelatihan. Untuk menjamin konvergensi model, parameter pelatihan seperti *Epoch*: 50 *Epoch* digunakan. *Size batch* 32 untuk menghemat memori selama pelatihan. Nilai awal untuk pengaturan kecepatan pembelajaran model adalah 0.001 [30].

Selama pelatihan, model menguji akurasi dengan menggunakan fungsi kehilangan *cross-entropy*. Ini adalah cara untuk mengetahui seberapa baik prediksi model sesuai dengan label yang benar. Model dievaluasi menggunakan akurasi dan metrik lain seperti *Precision*, *Recall*, dan *F1-Score* untuk mengukur performa deteksi botol [31].

$$Akurasi = \frac{Prediksi\ Benar}{Total\ Data} \times 100\% \tag{8}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{9}$$

Waktu Komputasi

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (10)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (11)$$

Setelah proses pelatihan, model diuji untuk mengevaluasi kinerjanya dengan data uji. Parameter evaluasi utama yang digunakan adalah sebagai berikut:

- Akurasi, yang menunjukkan tingkat keberhasilan model dalam mendeteksi botol
- Ketepatan, *Recall*, dan *Skor F1*, yang menunjukkan performa deteksi yang lebih rinci, terutama dalam situasi ketidakseimbangan data.
- *Confusion Matrix*: Digunakan untuk melihat distribusi kesalahan klasifikasi. Ini melihat apakah model sering menganggap botol sebagai objek yg berbeda.

Implementasi pada Robot

Selanjutnya, model yang telah dilatih dimasukkan ke dalam sistem pengoperasian robot yang berbasis ROS [32]. Kamera yang dipasang pada robot mengambil gambar secara real-time. Gambar ini kemudian diproses oleh model *deep learning* untuk menentukan apakah objek yang terdeteksi adalah botol atau bukan.

Hasil dan Pembahasan

Dengan menggunakan algoritma *Convolutional Neural Networks* (CNN), penelitian ini mengembangkan sistem deteksi botol yang otonom. Setelah pelatihan, pengujian, dan penerapan pada robot, hasilnya adalah sebagai berikut:

Akurasi Model CNN

Model CNN yang dilatih dengan dataset botol menunjukkan kinerja yang sangat baik dalam mendeteksi botol. Hasil pengujian menunjukkan akurasi model mencapai 95% pada data uji, yang menunjukkan bahwa model dapat mengenali dan mengklasifikasikan botol dengan tingkat akurasi yang cukup tinggi.

Beberapa metrik tambahan yang diperoleh adalah: *Precision*: 93%, *Recall*: 94%, *F1-Score*: 93%. *Precision* yang tinggi dan *recall* yang tinggi menunjukkan bahwa model mampu mendeteksi sebagian besar botol yang ada dalam gambar.

Waktu komputasi yang dihasilkan oleh model untuk melakukan deteksi dalam gambar *real-time* adalah sekitar 0,2 detik per gambar, yang cukup cepat untuk digunakan dalam aplikasi robotika. Aplikasi seperti pemilahan botol dalam sistem manajemen sampah atau industri daur ulang dapat menggunakan kecepatan ini untuk melakukan tugasnya tanpa penundaan.

Beberapa metrik utama yang kami perhatikan saat mengevaluasi waktu komputasi untuk sistem deteksi botol menggunakan *deep learning* adalah *throughput* (jumlah *frame* yang diproses per detik), *latency* (waktu tunda), dan *real-time factor*. *Throughput* merupakan ukuran jumlah frame yang dapat diproses per detik. Untuk aplikasi *real-time*, ini merupakan indikator penting karena menunjukkan seberapa responsif sistem terhadap perubahan lingkungan.

$$Throughput = \frac{1}{TLF} \times 1000(F/s) \quad (12)$$

- *TLF*: Total Latency per Frame
- *F/s*: Frames per second

Dalam lingkungan dinamis dengan perubahan cepat, *throughput* di bawah 20 FPS mungkin tidak cukup karena robot tidak dapat merespons perubahan dalam lingkungan secara *real-time*. Oleh karena itu, sangat penting untuk memastikan bahwa sistem memiliki kemampuan untuk memproses 30 FPS atau lebih, terutama dalam situasi di mana deteksi dan komunikasi diperlukan dengan cepat. *Real-time factor* merupakan rasio antara waktu komputasi aktual dan waktu yang tersedia untuk menjaga *real-time performance*. RTF lebih dari 1 berarti sistem gagal memenuhi batasan waktu yang dibutuhkan untuk operasi *real-time*, sementara RTF di bawah 1 berarti sistem berfungsi dalam kapasitas *real-time*.

Pengujian pada Berbagai Kondisi

Robot yang telah menggunakan model CNN diuji dalam berbagai lingkungan, seperti:

- Kondisi pencahayaan rendah: Meskipun kinerjanya menurun dibandingkan dengan kondisi pencahayaan normal. Gambar 1 robot deteksi botol, dimana model tetap masih dapat mendeteksi botol dengan akurasi sekitar 87%.



Gambar 1: Robot deteksi botol

- Sudut pandang bervariasi: Robot dapat mendeteksi botol dengan akurasi 90%–95% tergantung pada kemiringan sudut.
- Penghalang sebagian: Dalam sekitar 80% kasus, jika botol sebagian tertutup oleh objek lainnya, model masih dapat menemukannya. Meskipun ada beberapa masalah ketika botol tidak dapat dilihat secara penuh, deteksi masih cukup akurat. Tetapi degradasi Kualitas Gambar merupakan masalah terbesar untuk menerapkan *deep learning* ini. Model *deep learning*, khususnya *Convolutional Neural Networks* (CNN), sangat bergantung pada kualitas gambar yang diumpankan ke sistem untuk melakukan tugas-tugas seperti pengenalan objek dan segmentasi gambar. Gambar yang dihasilkan dalam pencahayaan rendah mungkin tidak cukup informatif, menyulitkan model untuk mengenali objek atau lingkungan dengan akurat.
- Selain *Generalization Gap*: *Deep learning* sangat efektif ketika dilatih dalam kondisi pencahayaan yang konsisten; namun, generalisasi terhadap kondisi baru yang berbeda dari data pelatihan, seperti pencahayaan yang sangat berbeda.

Confusion Matrix

Sebagian besar kesalahan deteksi terjadi pada gambar di mana botol terhalang oleh objek lain atau gambar dengan pencahayaan yang buruk. Objek dengan bentuk yang mirip dengan botol, seperti kaleng atau gelas, juga memiliki kesalahan klasifikasi yang lebih kecil, menurut *confusion matrix* yang dihasilkan dari pengujian data uji.

Tabel 1: Prediksi Botol

	Prediksi Botol	Prediksi Bukan Botol
Botol Sebenarnya	94%	6%
Bukan Botol	4%	96%

Setelah model dimasukkan ke dalam robot, ia diuji dalam lingkungan kehidupan nyata. Robot dengan kamera dapat bergerak sendiri dan mendeteksi botol di sekitarnya. Hasil uji coba sistem deteksi adalah sebagai berikut: Kecepatan deteksi robot mampu mendeteksi botol dalam waktu kurang dari 1 detik setelah objek muncul dalam bidang pandangnya. Keandalan deteksi dalam uji coba dengan 100 objek botol yang tersebar secara acak, robot mendeteksi dan mengenali botol dalam 94 kasus (lihat Tabel 1). Adaptabilitas lingkungan robot berfungsi lebih baik di luar ruangan dengan pencahayaan sangat terang atau sangat gelap, tetapi mereka dapat berfungsi dengan baik di dalam ruangan dengan berbagai tingkat pencahayaan.

Secara keseluruhan dapat dikatakan bahwa model CNN berhasil memenuhi target kecepatan dan akurasi pada robot deteksi botol. Namun, beberapa masalah muncul, terutama dengan pencahayaan ekstrem dan objek yang sebagian tertutup.

Pengembangan lebih lanjut untuk penerapan *deep learning* termasuk diversifikasi dan peningkatan variasi data pelatihan untuk meningkatkan generalisasi model, langkah pengembangan penggunaan data sintesis dan simulasi, membuat dataset sintesis menggunakan simulasi realistis (seperti *unity* atau *gazebo*) dapat memperkaya data pelatihan dengan kondisi lingkungan yang jarang atau sulit di dunia nyata.

Pengembangan model *deep learning* yang dilatih dengan data sintesis teknik augmentasi data lanjutan, augmentasi berbasis *domain-shift* (misalnya, perubahan gaya visual antar kamera atau sensor) juga dapat membantu membuat model lebih kuat terhadap gangguan visual. Teknik seperti pemetaan tekstur acak, distorsi geometris, dan penambahan suara ke gambar juga dapat membantu.

Integrasi sistem multi-robot dan penggunaan berbagai jenis robot, langkah pengembangan sistem berbasis multi-robot mengembangkan sistem pembelajaran mendalam yang dapat digunakan pada kelompok robot yang bekerja sama untuk menyelesaikan tugas bersama. Dengan menggunakan pendekatan pembelajaran terdistribusi, setiap robot dalam sistem ini dapat dilengkapi dengan model yang berbagi informasi dan pengalaman.

Penggunaan robot dengan sensor beragam, penggunaan *deep learning* dengan robot yang menggunakan berbagai jenis sensor, seperti kamera inframerah, LiDAR, dan RGB, memungkinkan robot beradaptasi dengan lebih baik dalam situ-

asi yang tidak pasti. Misalnya, kamera inframerah dapat digunakan dalam lingkungan dengan pencahayaan rendah, sedangkan LiDAR dapat digunakan untuk navigasi di ruang yang penuh hambatan.

Pengembangan model generik untuk berbagai jenis robot meningkatkan fleksibilitas model *deep learning* sehingga dapat diterapkan pada *platform* robotik yang berbeda, baik yang memiliki kemampuan manipulasi (robot lengan) maupun robot yang lebih *mobile* seperti *drone* atau robot beroda. Pengembangan tambahan dapat memperbaiki masalah ini dengan menambah data pelatihan dengan skenario pencahayaan ekstrem dan sebagian objek yang tersembunyi. Untuk meningkatkan akurasi dalam kondisi yang lebih sulit, teknik *ensemble model* digunakan untuk menggabungkan berbagai model yang dilatih dengan berbagai jenis data.

Penutup

Studi ini menciptakan dan menerapkan algoritma *deep learning* berbasis *Convolutional Neural Networks* (CNN) untuk mendeteksi botol dalam sistem robot otonom. Hasil menunjukkan beberapa hal penting berikut: Akurasi Deteksi yang Tinggi: Model CNN yang dilatih dalam penelitian ini dapat mendeteksi botol dengan akurasi 95%. Ini menunjukkan bahwa pendekatan *deep learning* sangat efektif dalam mengenali objek dengan bentuk dan fitur visual tertentu, seperti botol. Waktu Deteksi *Real-Time*: Sistem dapat mendeteksi botol dalam waktu rata-rata 0,2 detik per gambar, sehingga cocok untuk robot otonom yang perlu mendeteksi objek secara *real-time*, seperti dalam pemilahan sampah atau operasi industri otomatis.

Secara keseluruhan, penelitian ini menunjukkan bahwa menggunakan algoritma *deep learning*, khususnya CNN, dapat menghasilkan sistem robot deteksi botol yang efektif dan efisien. Teknologi ini memiliki potensi besar untuk digunakan dalam berbagai industri, seperti industri daur ulang, manajemen sampah cerdas, dan aplikasi robotika lainnya yang membutuhkan kemampuan deteksi objek yang cepat dan akurat.

Daftar Pustaka

- [1] M. Soori, B. Arezoo, and R. Dastres, "Artificial intelligence, machine learning and deep learning in advanced robotics, a review", KeAi Communications Co. doi: 10.1016/j.cogr.2023.04.001, 2023.
- [2] L.G. Divyanth, Divya Rathore, Piranav Senthilkumar, Prakhar Patidar, Xin Zhang, Manoj Karkee, Rajendra Machavaram, and Peeyush Soni, "Estimating depth from RGB images using deep learning for robotic applications in apple orchards", *Smart Agricultural Technology*, vol. 6, doi: 10.1016/j.atech.2023.100345, 2023.
- [3] Siyu Zhou, Yuan Gao, Renpeng Li, Huizhi Wang, Moxuan Zhang, Yuzhu Guo, Weigang Cui, Kayla Giovanna Brown, Chunlei Han, Lin Shi, Huanguang Liu, Jianguo Zhang, Yang Li, and Fangang Meng, "Neurosurgical robots in China: State of the art and future prospect", *iScience*, Vol: 26(11), Elsevier Inc, doi: 10.1016/j.isci.2023.107983, 2023.
- [4] Chao Zhuang, Tianyi Ma, Bokai Xuan, Cheng Chang, Baichuan An, Minghuan Yin, and Hao Sun, "Deep learning- based semantic segmentation of human features in bath scrubbing robots", *Biomimetic Intelligence and Robotics*, vol. 4, no. 1, doi: 10.1016/j.birob.2024.100143, 2024.
- [5] M. L. Dezaki, S. Hatami, A. Zolfagharian, and M. Bodaghi, "A pneumatic conveyor robot for color detection and sorting", *Cognitive Robotics*, vol. 2, pp. 60–72 , doi: 10.1016/j.cogr.2022.03.001, 2022.
- [6] U. Aulia, I. Hasanuddin, M. Dirhamsyah, and N. Nasaruddin, "A new CNN-BASED object detection system for autonomous mobile robots based on real-world vehicle datasets", *Heliyon*, vol. 10, no. 15, doi: 10.1016/j.heliyon.2024.e35247, 2024.
- [7] H. Sekkat, O. Moutik, B. El Kari, Y. Chaibi, T. A. Tchakoucht, and A. El Hilali Alaoui, "Beyond simulation: Unlocking the frontiers of humanoid robot capability and intelligence with Pepper's open-source digital twin", *Heliyon*, vol. 10, no. 14, doi: 10.1016/j.heliyon.2024.e34456, 2024.
- [8] Tomohito Sekine, Keita Ito, Yoshinori Shouji, Ryosuke Suga, Tatsuya Yasuda, Yi-Fei Wang, Yasunori Takeda, Daisuke Kumaki, Fabrice Domingues Dos Santos, Huang Tong, Atsushi Miyabo, and Shizuo Tokito, "Robotic e-skin for high performance stretchable acceleration sensor via combinations of novel soft and functional polymers", *Appl Mater Today*, vol. 33, doi: 10.1016/j.apmt.2023.101877, 2023.
- [9] R. Fernandez-Fernandez, J. G. Victores, and C. Balaguer, "Deep Robot Sketching: An application of Deep Q-Learning Networks for human-like sketching", *Cogn Syst Res*, vol. 81, pp. 57–63, doi: 10.1016/j.cogsys.2023.05.004, 2023.
- [10] John Shanley, Fan Pu, Jon D. Williams, Nathaniel L. Elsen, Sujatha M. Gopalakrishnan, Jeffrey Y Pan, and Andrew J. Radosevich, "Collaborative robotics to enable ultra-high-throughput IR-MALDEST", *SLAS Technol*, p. 100163, doi: 10.1016/j.slst.2024.100163, 2024.

- [11] C. Qin, A. Song, H. Li, L. Zhu, X. Zhang, and J. Wang, "Overcoming the cognition-reality gap in robot-to- human handovers with anisotropic variable force guidance", *Comput Struct Biotechnol J*, vol. 24, pp. 185– 195, doi: 10.1016/j.csbj.2024.02.020, 2024.
- [12] Jisoo Kim, Azat Slyamov, Erik Lauridsen, Mie Birkbak, Tiago Ramos, Federica Marone, Jens W. Andreasen, Marco Stampanoni, and Matias Kagias, "Macroscopic mapping of microscale fibers in freeform injection molded fiber-reinforced composites using X-ray scattering tensor tomography", *Compos B Eng*, vol. 233 , doi: 10.1016/j.compositesb.2022.109634, Mar. 2022.
- [13] R. Fernandez-Fernandez, J. G. Victores, and C. Balaguer, "Deep Robot Sketching: An application of Deep Q-Learning Networks for human-like sketching", *Cogn Syst Res*, vol. 81, pp. 57–63 , doi: 10.1016/j.cogsys.2023.05.004, Sep. 2023.
- [14] E. Okafor, M. Oyediji, and M. Alfarraj, "Deep reinforcement learning with light-weight vision model for sequential robotic object sorting", *Journal of King Saud University - Computer and Information Sciences*, vol. 36, no. 1 , doi: 10.1016/j.jksuci.2023.101896, Jan. 2024.
- [15] M. R. Wilkinson, B. Castro- Dominguez, C. C. Wilson, and U. Martinez-Hernandez, "Low-cost, autonomous microscopy using deep learning and robotics: A crystal morphology case study", *Eng Appl Artif Intell*, vol. 126 , doi: 10.1016/j.engappai.2023.106985, Nov. 2023.
- [16] E. U. Thoden van Velzen, Y. Workala, W. Tenissen, and I. Smeding, "Volatile organic contaminants in HDPE milk bottles along the mechanical recycling value chain, revealing origins and contamination pathways", *J Clean Prod*, vol. 459 , doi: 10.1016/j.jclepro.2024.142571, Jun. 2024.
- [17] V. B. Gjørnum, I. Strümke, J. Løver, T. Miller, and A. M. Lekkas, "Model tree methods for explaining deep reinforcement learning agents in real-time robotic applications", *Neurocomputing*, vol. 515, pp. 133– 144 , doi: 10.1016/j.neucom.2022.10.014, Jan. 2023.
- [18] P. Malvido Fresnillo, S. Vasudevan, W. M. Mohammed, J. L. Martinez Lastra, and J. A. Pérez García, "A method for understanding and digitizing manipulation activities using programming by demonstration in robotic applications", *Rob Auton Syst*, vol. 170 , doi: 10.1016/j.robot.2023.104556, Dec. 2023.
- [19] E. Kok and C. Chen, "Occluded apples orientation estimator based on deep learning model for robotic harvesting", *Comput Electron Agric*, vol. 219 , doi: 10.1016/j.compag.2024.108781, Apr. 2024.
- [20] S. Wang, J. Zhang, P. Wang, J. Law, R. Calinescu, and L. Mihaylova, "A deep learning-enhanced Digital Twin framework for improving safety and reliability in human-robot collaborative manufacturing", *Robot Comput Integr Manuf*, vol. 85 , doi: 10.1016/j.rcim.2023.102608, Feb. 2024.
- [21] S. C. Steinhäusser and B. Lugin, "Integrating sound effects and background music in Robotic storytelling – A series of online studies across different story genres", *Computers in Human Behavior: Artificial Humans*, vol. 2, no. 2, p. 100085 , doi: 10.1016/j.chbah.2024.100085, Aug. 2024.
- [22] Sung Gon Park, Jeonghyun Park, Hong Rock Choi, Jun Ho Lee, Sung Tae Cho, Young Goo Lee, Hanjong Ahn, and Sahyun Pak, "Deep Learning Model for Real-time Semantic Segmentation During Intraoperative Robotic Prostatectomy", *Eur Urol Open Sci*, vol. 62, pp. 47–53 , doi: 10.1016/j.euro.2024.02.005, Apr. 2024.
- [23] Yuankai Wu, Rayene Messaoud, Xiao Chen, Arne-Christoph Hildebrandt, Marco Baldini, Constantin Patsch, Hamid Sadeghian, Sami Haddadin, and Eckehard Steinbach, "Vision-driven Collaborative Mobile Robotic Human Assistant System for Daily Living Activities", in *IFAC- PapersOnLine*, Elsevier B.V., pp. 4400–4405. doi: 10.1016/j.ifacol.2023.10.1824, Jul. 2023.
- [24] V. Sejdiu, A. Pajaziti, G. Rexha, X. Bajrami, E. Rrustemi, and J. Kola, "Detection, Recognition, and Grasping of Objects through Artificial Intelligence Using a Robotic Hand", in *IFAC- PapersOnLine*, Elsevier B.V. , pp. 443–446. doi: 10.1016/j.ifacol.2022.12.077, Oct. 2022.
- [25] A. Z. Swartz, V. Novoa y Arruga Novoa, J. S. Hassoun, M. A. Crispens, and L. S. Prescott, "Robotic-assisted gynecologic surgery associated tympanic membrane perforation: A report of two cases and review of the literature", *Gynecol Oncol Rep*, vol. 46 , doi: 10.1016/j.gore.2023.101151, Apr. 2023.
- [26] J. Dikobe, F. A. Melato, C. J. L. Adlem, and K. Netshiongolwe, "Determination of chromium species in water using diphenylcarbazide with a sequential spectrophotometric discrete robotic analyser", *Heliyon*, vol. 10, no. 14 , doi: 10.1016/j.heliyon.2024.e34670, Jul. 2024.
- [27] Y. Zhao, X. An, and N. Sun, "Virtual simulation experiment of the design and manufacture of a beer bottle- defect detection sys-

- tem.,” *Virtual Reality and Intelligent Hardware*, vol. 2, no. 4, pp. 354–367 , doi: 10.1016/j.vrih.2020.07.002, Aug. 2020.
- [28] Z. Cai, Z. Feng, L. Zhou, X. Yang, and T. Xu, “Deep-reinforcement- learning-based robot motion strategies for grabbing objects from human hands,” *Virtual Reality and Intelligent Hardware*, vol. 5, no. 5, pp. 407–421 , doi: 10.1016/j.vrih.2022.12.001, Oct. 2023.
- [29] Y. Dmytriyev, M. Carnevale, and H. Giberti, “Enhancing flexibility and safety: collaborative robotics for material handling in end-of-line industrial operations”, in *Procedia Computer Science*, Elsevier B.V. , pp. 2588–2597. doi: 10.1016/j.procs.2024.02.077, 2024.
- [30] R. Aarthi and G. Rishma, “A Vision Based Approach to Localize Waste Objects and Geometric Features Exaction for Robotic Manipulation”, in *Procedia Computer Science*, Elsevier B.V. , pp. 1342–1352. doi: 10.1016/j.procs.2023.01.113, 2022.
- [31] P. Zheng, C. Li, J. Fan, and L. Wang, “A vision-language-guided and deep reinforcement learning-enabled approach for unstructured human- robot collaborative manufacturing task fulfilment”, *CIRP Annals*, vol. 73, no. 1, pp. 341–344, Jan. 2024, doi: 10.1016/j.cirp.2024.04.003, Jan. 2024.
- [32] A. Da Rold, M. Furiato, A. M. A. Zaki, M. Carnevale, and H. Giberti, “Deep learning-based robotic sorter for flexible production”, in *Procedia Computer Science*, Elsevier B.V. , pp. 1579–1588. doi: 10.1016/j.procs.2022.12.358, 2022.