

MONITORING LAHAN PARKIR MENGGUNAKAN PYTHON, CV2, CVZONE DAN NUMPY

Dewa Putu Sealtiel Armeza Diandra

Teknik Informatika, Universitas Pendidikan Nasional Denpasar
Jl. Bedugul No.39, Sidakarya, Denpasar Selatan, Kota Denpasar, Bali 80224
Armezadiandra@gmail.com

ABSTRAK

Sistem monitoring lahan parkir yang ada saat ini masih sangat bergantung pada komunikasi manual antar petugas, yang menyebabkan lambatnya aliran informasi mengenai ketersediaan slot parkir dan berpotensi menimbulkan kemacetan. Penelitian ini bertujuan untuk mengembangkan sistem pemantauan parkir berbasis Python dan teknologi *IoT* yang dapat mendeteksi ketersediaan tempat parkir secara *real-time*. Metode yang digunakan meliputi pemrograman dengan pustaka Python seperti OpenCV, CVZone, dan Numpy untuk memproses data visual dari kamera pengawas. Hasil dari pengujian menunjukkan bahwa sistem ini mampu mendeteksi slot parkir kosong dengan akurasi 98,55%, meskipun terdapat sedikit kesalahan deteksi akibat kendaraan yang tidak sepenuhnya masuk ke dalam slot. Dengan penerapan sistem ini, diharapkan dapat meningkatkan efisiensi pengelolaan lahan parkir dan mengurangi kemacetan akibat pencarian tempat parkir yang kosong. Penelitian ini memberikan kontribusi signifikan terhadap inovasi teknologi dalam manajemen parkir di area perkotaan.

Kata kunci : *Python, Smart Parking System, IoT*

1. PENDAHULUAN

Sistem monitoring parkir yang ada saat ini masih sangat bergantung pada komunikasi antar petugas parkir menggunakan alat komunikasi jarak jauh. Ketergantungan ini dapat menyebabkan lambatnya aliran informasi mengenai ketersediaan slot parkir, yang pada gilirannya mengakibatkan kemacetan akibat antrean kendaraan yang mencari tempat parkir kosong. Dengan adanya sistem pemantauan parkir yang lebih efisien, diharapkan masalah pencarian tempat parkir dapat diatasi secara efektif, sehingga pengguna kendaraan tidak perlu berkeliling mencari slot parkir dan mengurangi potensi terjadinya kemacetan [1].

Pengembangan sistem ini diharapkan dapat meningkatkan produktivitas petugas parkir. Melalui teknologi yang tepat, terutama pemanfaatan *Internet of Things (IoT)* dan bahasa pemrograman *Python*, petugas dapat dengan cepat mengetahui lokasi slot parkir yang kosong dan segera menginformasikannya kepada pengendara. *Python*, dengan kemampuannya dalam pengolahan data dan integrasi perangkat keras, memungkinkan pengembangan aplikasi yang dapat memberikan informasi ketersediaan slot secara *real-time* [2].

Dalam konteks ini, penerapan teknologi berbasis *IoT* menjadi sangat relevan. Teknologi ini memungkinkan pemantauan otomatis dan akurat terhadap kondisi area parkir serta memfasilitasi komunikasi yang lebih cepat antara petugas dan pengguna. Dengan menggunakan sensor cerdas dan algoritma yang ditulis dalam *Python*, sistem dapat mendeteksi dan melaporkan ketersediaan slot parkir secara efisien [2].

Dengan demikian, diharapkan permasalahan klasik dalam manajemen parkir dapat diminimalisasi, menciptakan pengalaman yang lebih baik bagi

pengguna kendaraan dan efisiensi operasional bagi pengelola area parkir. Melalui inovasi ini, sistem *monitoring* lahan parkir kosong berbasis *IoT* dan *Python* tidak hanya akan meningkatkan efektivitas operasional tetapi juga memberikan solusi berkelanjutan untuk tantangan urbanisasi saat ini. Pendahuluan ini sudah dirancang untuk mencerminkan fokus pada penggunaan *Python* dalam konteks sistem monitoring lahan parkir berbasis *IoT*, sambil mempertahankan alur logis dari masalah hingga solusi yang diusulkan [2].

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Abdelrahman Osman Elfaki, Wassim Messoudi, Anas Bushnag, dan Shakour Aabuzneid, dalam penelitian mereka yang berjudul "*A Smart Real-Time Parking Control and Monitoring System*" pada tahun 2023, membahas pengembangan sistem parkir pintar yang dapat melakukan *monitoring* dan kontrol secara *real-time*. Penelitian ini memberikan solusi inovatif untuk meningkatkan efisiensi dalam pengelolaan area parkir dengan memanfaatkan teknologi terkini [3]. Marvelina Gracia Hernoko, Suryo Adi Wibowo, dan Nurlaily Vendyansyah. Mereka meneliti tentang "Penelitian *IoT (Internet of Things) Smart Parking System* Dan Pendeteksi Kebakaran Dengan Fitur *Monitoring*" pada tahun 2021, membahas penerapan sistem *monitoring* parkir yang dapat secara otomatis memberitau secara *real-time* ketersediaan tempat parkir serta penambahan sistem pendeteksi kebakaran di area parkir. Penelitian ini menunjukkan bagaimana teknologi *IoT* dapat meningkatkan efisiensi dan keamanan dalam pengelolaan area parkir [1].

2.2. Jaringan Komputer

Berdasarkan [4], Jaringan komputer adalah sistem yang menghubungkan dua atau lebih *node*, baik melalui koneksi *wired* (kabel) maupun *wireless* (tanpa kabel), dengan tujuan untuk memungkinkan komunikasi dan berbagi data antara perangkat-perangkat yang terhubung. Berdasarkan , jaringan ini tidak hanya mencakup berbagai perangkat seperti komputer, server, dan printer, tetapi juga perangkat lain yang dapat saling berinteraksi untuk bertukar informasi dan sumber daya. Dalam era digital saat ini, jaringan komputer memainkan peran penting dalam berbagai aspek kehidupan sehari-hari, mulai dari komunikasi pribadi hingga aplikasi bisnis yang kompleks. Dengan adanya jaringan komputer, pengguna dapat mengakses informasi secara cepat dan efisien, serta memanfaatkan berbagai layanan *online* yang tersedia.

2.3. Python

Python adalah bahasa pemrograman komputer yang sangat populer dan digunakan secara luas dalam berbagai bidang, termasuk pembangunan *software*, analisis data, dan pengembangan situs web. Bahasa ini pertama kali dirilis pada tahun 1991 oleh Guido van Rossum dan sejak itu telah berkembang menjadi salah satu bahasa pemrograman terpopuler keempat di dunia. Keunikan Python terletak dalam kemudahan penggunaannya yang berbeda dengan banyak bahasa pemrograman lainnya. Python menggunakan sintaksis yang mirip dengan bahasa Inggris, sehingga membuatnya mudah dipelajari bagi programmer baru. Struktur kode yang sederhana serta ekspresi fungsi yang fleksibel memungkinkan *developer* untuk menulis program dengan cepat dan efisien. Selain itu, komunitas Python sangat besar dan aktif, menyediakan banyak perpustakaan dan modul *open-source* yang dapat digunakan untuk meningkatkan kinerja aplikasi. [5]

2.4. Smart Parking

Parkir pintar atau smart parking adalah metode inovatif yang dirancang untuk memudahkan pengendara dalam mencari tempat parkir di area yang luas. Dengan sistem parkir pintar ini, petugas parkir dapat dengan cepat mengetahui lokasi *slot* parkir yang kosong dan mengarahkan pengemudi ke tempat yang tersedia. Hal ini tidak hanya mengurangi waktu yang dibutuhkan pengendara untuk menemukan tempat parkir, tetapi juga secara signifikan mengurangi kemacetan yang sering terjadi akibat kendaraan yang berputar-putar mencari slot kosong. Sistem ini memanfaatkan teknologi canggih seperti sensor dan algoritma pemrosesan citra untuk memberikan informasi *real-time* tentang ketersediaan tempat parkir. Dengan demikian, pengemudi dapat dengan mudah mendapatkan informasi akurat mengenai slot parkir yang tersedia, meningkatkan efisiensi penggunaan lahan parkir dan menciptakan pengalaman yang lebih baik bagi pengguna kendaraan.

Implementasi teknologi ini diharapkan dapat menjadi solusi efektif untuk tantangan manajemen parkir di area perkotaan yang semakin padat. [1]

2.5. CCTV

Kamera pengawas atau CCTV digunakan untuk merekam suara dan gambar bergerak. Hasil rekaman ini kemudian dikirim ke videorekorder. Setelah itu, isi rekaman dapat ditampilkan pada monitor untuk memantau situasi. Dengan adanya CCTV yang digunakan untuk memantau keadaan secara langsung dengan ini juga meningkatkan keamanan pengguna lahan parkir. [6]

2.6. Numpy

Numpy(*numerical python*) sebuah pustaka yang sangat penting dalam dunia pemrograman Python, khususnya dalam bidang manipulasi data dan komputasi numerik. Pustaka ini dirancang untuk memberikan dukungan yang kuat dalam melakukan operasi aljabar linear dan pengolahan matriks, yang merupakan elemen kunci dalam banyak aplikasi ilmiah dan teknik. Dengan NumPy, pengguna dapat dengan mudah melakukan operasi matematis yang kompleks pada array multidimensi, memungkinkan analisis data yang lebih efisien dan cepat. Salah satu keunggulan utama dari NumPy adalah kemampuannya untuk memproses data dalam jumlah besar dengan waktu pemrosesan yang relatif singkat, berkat optimasi yang dilakukan pada algoritma dasar dan penggunaan struktur data yang efisien. [7]

2.7. OpenCV

OpenCV atau *Open Source Computer Vision* adalah sebuah pustaka perangkat lunak yang dirancang untuk memecahkan berbagai masalah dalam bidang visi komputer. Pertama kali dirilis pada tahun 1999, OpenCV telah berkembang pesat dan menjadi salah satu alat utama yang digunakan oleh para peneliti dan pengembang di seluruh dunia untuk aplikasi terkait pengolahan citra dan video. Salah satu keunggulan utama OpenCV adalah kemampuannya untuk menangani berbagai algoritma pemrosesan gambar dan video secara efisien, memungkinkan pengguna untuk melakukan tugas-tugas kompleks seperti deteksi wajah, pelacakan objek, dan pengenalan pola dengan lebih mudah.. [8]

2.8. CVZone

CVZone adalah sebuah perpustakaan yang ada di python, CVZone sering digunakan untuk *Artificial Intelligence, Computer Vision, dan Deep Learning*. Dengan menggabungkan keunggulan dari dua pustaka terkenal, yaitu OpenCV dan Mediapipe, CVZone menawarkan antarmuka yang lebih sederhana dan intuitif, sehingga memungkinkan pengguna untuk fokus pada pengembangan fungsi kecerdasan buatan tanpa harus terjebak dalam kompleksitas teknis yang sering kali menyertai pemrograman visi komputer. Salah satu tujuan utama dari CVZone adalah untuk

menyederhanakan proses pengolahan gambar dan video, sehingga pengguna dapat dengan cepat menerapkan algoritma deteksi objek, pelacakan, dan pengenalan pola dengan lebih efisien. [9]

2.9. Pickle

Pickle adalah modul dalam Python yang berfungsi untuk menyimpan dan memanggil kembali objek dengan cara yang efisien. Pustaka ini memungkinkan pengguna untuk menyimpan berbagai jenis objek Python, termasuk struktur data kompleks seperti daftar, kamus, dan kelas, ke dalam berkas dengan format biner. Dengan menggunakan Pickle, pengguna dapat menyimpan keadaan aplikasi mereka sehingga dapat diakses kembali di lain waktu tanpa kehilangan informasi. Proses penyimpanan ini dikenal sebagai "*pickling*," sedangkan proses pemanggilan kembali objek yang telah disimpan disebut "*unpickling*." [10]

2.10. CV2

CV2 adalah *upgrade* dari OpenCV, yang merupakan perpustakaan yang dirancang untuk memecahkan masalah-masalah pada visi komputer. Versi ini difungsikan sebagai generator yang tidak perlu menyimpan semua data di dalam memori secara bersamaan, melainkan menghasilkan nilai-nilai secara dinamis. Hal ini membuat penggunaan CV2 sangat cocok untuk mengolah data-data besar karena sistemnya membantu mengurangi beban kerja memori. Dengan fitur ini, CV2 dapat meningkatkan performa pemrosesan gambar dan video, membuatnya ideal untuk aplikasi-aplikasi yang memerlukan manipulasi citra skala besar tanpa mengganggu kinerja sistem. [7]

3. METODE PENELITIAN

Dalam penelitian ini membahas bagaimana mendeteksi sebuah lahan parkir kosong atau tidak dengan menggunakan python, CV2, CVZone dan Numpy untuk membantu memudahkan program.

3.1. Desain Studi

Dalam analisis yang dilakukan, terdapat beberapa aspek penting yang harus dipertimbangkan untuk memastikan keberhasilan sistem *monitoring* parkir yang dikembangkan. Pertama, objektif dari penelitian ini adalah untuk mengevaluasi seberapa akurat dan efektif teknologi yang digunakan dalam mendeteksi ketersediaan lahan parkir. Hal ini mencakup pengukuran kemampuan sistem dalam mengidentifikasi apakah suatu *slot* parkir kosong atau terisi, serta kecepatan dan ketepatan informasi yang diberikan kepada pengguna. Dengan menetapkan objektif yang jelas, penelitian ini bertujuan untuk memberikan kontribusi signifikan terhadap pengembangan teknologi pemantauan parkir yang lebih efisien. Kedua, *sample size* yang digunakan dalam penelitian ini adalah video yang diambil dari area perkotaan. Penggunaan sampel video dari

lingkungan nyata memungkinkan analisis yang lebih realistis mengenai kinerja sistem dalam kondisi lapangan yang bervariasi. Dengan menguji sistem pada berbagai situasi, seperti perbedaan jumlah kendaraan, jenis kendaraan, dan kondisi pencahayaan, penelitian ini dapat memberikan gambaran menyeluruh tentang efektivitas teknologi dalam mendeteksi ketersediaan tempat parkir. Melalui pendekatan ini, diharapkan hasil penelitian dapat memberikan wawasan berharga bagi pengembangan lebih lanjut dari sistem monitoring parkir pintar berbasis Python dan IoT, serta meningkatkan pemahaman tentang tantangan dan solusi dalam implementasi teknologi ini di dunia nyata.

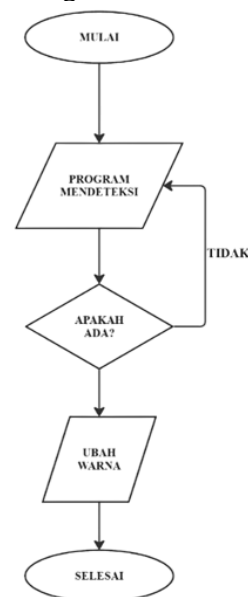
3.2. Rancangan Sistem



Gambar 1. Block Diagram Sistem

Pada gambar 1, Digambarkan block diagram sistem yang menjelaskan alur kerja dari sistem monitoring parkir pintar. Pertama, Sistem akan mengambil sumber data dari kamera yang terpasang di area parkir untuk memantau ketersediaan slot parkir. Setelah itu, kamera akan mengirimkan data video ke program yang telah dikembangkan untuk dianalisis. Program ini akan memproses sumber yang diterima dan melakukan deteksi terhadap slot parkir yang kosong atau terisi. Hasil analisis ini kemudian akan ditampilkan dalam bentuk output video, di mana setiap slot parkir akan ditandai dengan kotak berwarna sesuai dengan statusnya; hijau untuk slot yang kosong dan merah untuk slot yang terisi.

3.3. Rancangan Program



Gambar 2. Flowchart Perancangan Program

Pada gambar 2 yang merupakan *flowchart* perancangan program, Didalam *flowchart* tersebut digambarkan secara singkat bagaimana proses atau alur dari program yang ada. Pertama program yang dibuat akan mendeteksi vidio yang dikirimkan, Dimana program akan menentukan apa ada kendaraan didalam petak yang dibuat atau tidak, Jika ada kendaraan dan program mendeteksi perubahan pixel maka program akan mengubah warna petak parkir menjadi terisi atau menjadi warna yang ditentukan beserta mengubah berapa ketersediaan lahan parkir, Jika program tidak mendeteksi adanya perubahan maka program akan berjalan sampai mendeteksi adanya perubahan pixel yang terjadi dipetak parkir.

3.4. Proses Deteksi Objek

Didalam program terdapat perpustakaan atau *library*, Serta program ini dijalankan menggunakan *visual studio code* yang membantu mempermudah program untuk berjalan seperti :

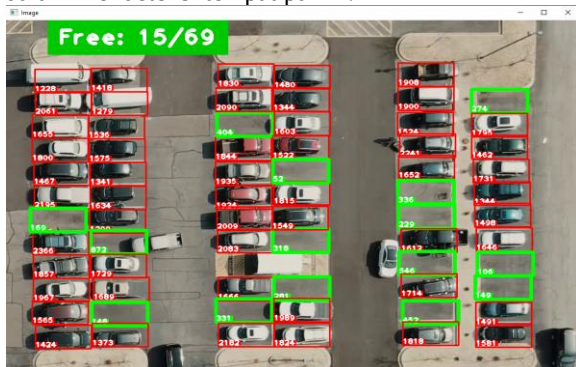
- Numpy yang berfungsi sebagai *library* untuk melakukan proses manipulasi data sehingga proses menjadi lebih cepat.
- Pickel berfungsi untuk menyimpan data kedalam bentuk format .pkl yang nanti dapat dipanggil kembali.
- CVZone berfungsi sebagai pengolah gambar sehingga gambar dan implementasi fungsi menjadi lebih mudah.
- CV2 adalah *library* yang berfungsi untuk mengolah objek seperti kotak.

Program ini menghitung jumlah piksel dalam setiap slot; jika jumlah piksel non-zero kurang dari 900, maka slot dianggap kosong dan kotak berwarna hijau. Sebaliknya, jika ada kendaraan yang terdeteksi (jumlah piksel lebih dari 900), kotak akan berwarna merah. Dengan cara ini, sistem dapat memberikan visualisasi yang jelas mengenai status setiap slot parkir secara real-time.

4. HASIL DAN PEMBAHASAN

4.1. Pengujian Program

Pengujian program dilakukan dengan cara mengambil vidio dari sebuah tempat parkir, Dengan ini program dapat dilihat tingkat keberhasilannya dalam mendeteksi tempat parkir.



Gambar 3. Tampilan Program

Gambar 3 merupakan hasil dari output program dimana program berhasil mendeteksi tempat parkir yang kosong dan terdapat kendaraan beserta jumlah tempat parkir yang tersedia dari total tempat parkir, Terdapat pemberitahuan berapa banyak tempat parkir yang tersisa.

```
def checkParkingSpace(imgProc):
    spaceCounter = 0
    for pos in posList:
        x, y = pos
        imgCrop = imgProc[y:y+height, x:x+width]
        count = cv2.countNonZero(imgCrop)
        if count < 900:
            color = (0, 255, 0)
            thickness = 1
            spaceCounter += 1
        else:
            color = (0, 0, 255)
            thickness = 2
        cv2.rectangle(img, pos, (pos[0] + width, pos[1] + height), color, thickness)
        cvzone.putTextRect(img, str(count), (x, y + height - 3), scale=1,
            thickness=2, offset=0, color=color)
    cvzone.putTextRect(img, f'Free: {spaceCounter}/{len(posList)}', (100, 50), scale=3,
        thickness=5, offset=20, color=(0, 200, 0))
```

Gambar 4. Source Code Yang Mengganti Warna Jika Terdapat Kendaraan

Pada gambar 4 adalah sumber kode yang mengganti warna jika terdapat kendaraan pada tempat parkir, Awalnya tempat parkir berwarna hijau sebagai penanda bahwa tempat parkir kosong namun jika ada penambahan pixel lebih dari 900 maka warna akan diganti menjadi merah yang menandakan tempat parkir terisi, Label sisa tempat parkir juga berubah berdasarkan ketersediaan tempat parkir.

```
width, height = 187, 48
try:
    with open('CarParkPos', 'rb') as f:
        posList = pickle.load(f)
except:
    posList = []

def mouseClick(events, x, y, flags, params):
    if events == cv2.EVENT_LBUTTONDOWN:
        posList.append((x, y))
    if events == cv2.EVENT_RBUTTONDOWN:
        for i, pos in enumerate(posList):
            x1, y1 = pos
            if x1 < x < x1 + width and y1 < y < y1 + height:
                posList.pop(i)

with open('CarParkPos', 'wb') as f:
    pickle.dump(posList, f)

while True:
    img = cv2.imread('carParkImg.png')
    for pos in posList:
        cv2.rectangle(img, pos, (pos[0] + width, pos[1] + height), (255, 0, 255), 2)
    cv2.imshow("Image", img)
    cv2.setMouseCallback("Image", mouseClick)
    cv2.waitKey(1)
```

Gambar 5. Source Code Yang Menandai Tempat Parkir

Pada gambar 5, Program dimulai dengan mencoba membuka file bernama 'CarParkPos' yang berisi daftar koordinat area parkir. Jika file tidak ada, program akan membuat daftar kosong. Program kemudian mendefinisikan fungsi *mouseClick* yang dipicu oleh klik mouse pada jendela gambar. Fungsi ini menambahkan koordinat klik ke daftar area parkir jika pengguna mengklik tombol kiri mouse dan menghapus koordinat dari daftar jika pengguna mengklik tombol kanan mouse. Akhirnya, program menampilkan gambar dan menunggu input dari pengguna, seperti klik mouse. Setiap kali pengguna mengklik, fungsi *mouseClick* akan dipicu dan daftar area parkir akan diperbarui. Program terus berulang hingga pengguna menutup jendela gambar.

Tabel 1. Pengujian keakuratan

	Kondisi Sebenarnya		Hasil deteksi Program		Error
	Terisi	Kosong	Terisi	Kosong	
Pengujian 1	54	15	52	17	1,45%
Pengujian 2	54	15	52	17	1,45%
Pengujian 3	54	15	52	17	1,45%
Pengujian 4	54	15	52	17	1,45%
Pengujian 5	54	15	52	17	1,45%
Pengujian 6	54	15	52	17	1,45%
Pengujian 7	54	15	52	17	1,45%
Pengujian 8	54	15	52	17	1,45%
Pengujian 9	54	15	52	17	1,45%
Pengujian 10	54	15	52	17	1,45%
Rata-rata					1,45%

Pengujian dilakukan dengan merekam video di area parkir dan menganalisis kemampuan sistem dalam mendeteksi ketersediaan tempat parkir. Hasil output program menunjukkan bahwa sistem berhasil mendeteksi slot parkir kosong dan terisi dengan baik. Namun, terdapat kesalahan deteksi sebesar 1,45% yang disebabkan oleh kendaraan yang tidak sepenuhnya masuk ke dalam slot, sehingga jumlah piksel tidak cukup untuk mengubah warna kotak menjadi merah. Ini menunjukkan bahwa meskipun sistem telah dirancang dengan baik, masih ada ruang untuk perbaikan dalam hal akurasi deteksi.

5. KESIMPULAN DAN SARAN

Setelah dilakukan pengujian program, diperoleh hasil bahwa sistem mampu menganalisis dengan baik posisi tempat parkir yang terisi dan yang kosong, yang ditandai dengan perubahan warna kotak; hijau untuk slot yang kosong dan merah untuk slot yang terisi. Meskipun demikian, hasil deteksi kendaraan menunjukkan adanya kesalahan sebesar 1,45%, yang disebabkan oleh kurangnya jumlah piksel yang terisi untuk mengubah warna kotak menjadi merah, sehingga tidak memberikan indikasi yang tepat bahwa slot parkir tersebut penuh. Dalam uji coba ini, hanya kendaraan mobil yang diuji, sementara kendaraan kecil seperti sepeda motor atau sepeda belum teruji. Oleh karena itu, pengembangan program selanjutnya akan difokuskan pada kemampuan sistem untuk mendeteksi kendaraan kecil seperti sepeda motor. Dengan adanya fitur tambahan ini, diharapkan para penjaga parkir dapat lebih efisien dalam mengatur alur parkir dan mengurangi kemacetan yang disebabkan oleh pengendara yang berputar-putar mencari tempat parkir kosong. Peningkatan akurasi deteksi dan perluasan cakupan jenis kendaraan akan sangat membantu dalam meningkatkan pengalaman pengguna serta efisiensi operasional di area parkir.

DAFTAR PUSTAKA

- [1] Marvelina Gracia Hernoko, Suryo Adi Wibowo, dan Nurlaily Vendyansyah, Penerapan IoT (Internet of Things) smart parking system dan pendeteksi kebakaran dengan fitur monitoring, vol 5, pp. 261-262, 2021. Available:

<https://ejournal.itn.ac.id/index.php/jati/article/view/3281>.

- [2] George Reynaldi Kote, Hesti Probodnananti, Johannes Daulat Tamba, Marshanda Krisnawi Saputri, Stelly Alison Kwa, Hadisantono, Parama Kartika Dewa, Penerapan internet of things pada smart parking system untuk kebutuhan pengembangan smart city, pp. 49-50, 2023. Available: <https://ojs.uajy.ac.id/index.php/JTIMR/article/view/7204>.
- [3] Abdelrahman Osman Elfaki, Wassim Messoudi, Anas Bushnag, dan Shakour Aabuzneid, A Smart Real-Time Parking Control and Monitoring System, vol 5, pp. 1-2, 2023. Available: <https://www.mdpi.com/1424-8220/23/24/9741>.
- [4] M. Noer Afdhol. P. Y. , Ami Aggraini Samudra, Rahayu Trisetyowati Untari, Perancangan jaringan komputer menggunakan metode failover, vol 7, pp. 1474-1475, 2023. Available : <https://ejournal.itn.ac.id/index.php/jati/article/view/7313>.
- [5] M. Riziq sirfatullah Alfarizi, Muhamad Zidan Alfarish, Muhamad Taufiqurrahman, Ginan Ardiansah, Muhamad Elgar, Penggunaan python sebagai bahasa pemrograman untuk machine learnin dan deep learning, vol 2, pp. 3-4, 2023. Available: <https://ojs.unida.ac.id/karimahtauhid/article/view/7518>.
- [6] Emanuel Safirman Bata, Pemanfaatan CCTV analog camera pada sistem monitoring keamanan ruangan berbasis mobile web, vol 11, pp. 29-36, 2023. Available: <https://ejurnal.undana.ac.id/index.php/jicon/article/download/10033/4887/>.
- [7] Fauziah, Yoan Elviralita, Widya Wisanty, Color Palette menggunakan Python cv2 dan NumPy, pp. 1-2, 2021. Available: <https://jurnal.poliupg.ac.id/index.php/sntei/article/view/4427>.
- [8] I Nyoman Tri Anindia Putra1, Ketut Sepdyana Kartini2, Yoga Kristian Suyitno, I Made Sugiarta, Ni Kadek Era Puspita, Penerapan Library Tensorflow, Cvzone, dan Numpy pada Sistem Deteksi Bahasa Isyarat Secara Real Time, vol 2, pp. 416-417, 2023. Available: <https://ejournal.sidyanusa.org/index.php/jkdn/article/view/335>.
- [9] Dody Khairianto, Rahmad Firdaus, Penerapan hand gesture recognition media kontrol presentasi aplikasi powerpoint, vol 8, pp. 1852-1860, 2024. Available: <https://ejournal.itn.ac.id/index.php/jati/article/view/9167>.
- [10] Sherina Viola Widayari, Muhammad Ihsan Muttaqin, Tariska Putri Ananda, Arnisa Stefanie, Implementasi internet of things pada sistem monitoring kematangan buah pepaya california dengan metode deep learning, vol 7, pp. 1946-1947, 2023. Available: <https://ejournal.itn.ac.id/index.php/jati/article/view/6953>