International Journal of Engineering Research and Development
e- ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 22, Issue 3 (March 2026), PP 121-125

Person Recognition from Crowd Using Python and OpenCV

Abhir Maske, Harsh Mahendra, Dipanshu pathare, Darshan Kinekar, Pravesh Dhabekar
ETC, PBCOE

Abstract— *Spotting people in busy settings matters more now, thanks to uses like city monitoring, keeping crowds safe, helping machines understand humans, and watching video feeds. Figuring out who is who gets tough when bodies block each other, lighting shifts, angles change, visuals are fuzzy, or surroundings get messy. This summary digs into up-to-date methods that aim to name persons in packed scenes - pulling apart how these tools work on photos and footage, mainly through code built with Python and powered by OpenCV.*
Starting off, the article outlines key steps in identifying people: spotting a person first, pulling out distinguishing traits, then matching that identity. Instead of jumping straight into new tech, it looks back at older techniques - like Haar cascades, HOG, and LBP - before shifting toward today's tools powered by deep learning. CNNs take center stage here, joined by fast systems like YOLO and SSD, plus Re-ID models built to track individuals across cameras. On another note, Python paired with OpenCV shows up often, helping run these models quickly while stitching together full working setups. Scattered throughout are mentions of standard data collections, ways to measure success, and how various methods stack up when tested side by side.
Ending on a note about gaps, the study points out hurdles still standing - like handling busy environments, keeping up with live data precision, while also wrestling with personal information safety. With that in mind, this overview sets out to clarify where things stand now in research circles, offering something steady for scholars and learners alike to build from Finding faces through code that
runs with Python, tied to tools like OpenCV. Building ways to spot individuals by how they appear in images captured live or stored earlier.
Person Recognition Crowd Analysis Computer Vision OpenCV Python Face Recognition Human Detection Feature Extraction Deep Learning Person Re ID Video Surveillance Convolutional Neural Networks CNNs

I. Introduction

Most times, spotting one person in a busy scene matters a lot - think cameras watching streets or cities running on data. When places get packed, telling who is who becomes tough. Obstacles block parts of people. Bodies twist into odd angles. Light shifts mess up details. Pictures come out blurry or dark. Old ways of recognizing faces often fail under such conditions.

Nowadays machines see people better in busy scenes thanks to smarter visual algorithms. Built on Python and OpenCV, many programs handle photos, spot figures, while watching live footage at once. Looking back at known methods shows what works, where things still go wrong.

II. Literature Survey

1.Sun, D., Huang, J., Hu, L., Tang, J., & Ding, Z. (2022). **"**Multitask Multigranularity Aggregation With Global Guided Attention for Video Person Re-Identification**"**.

Abstract : Spotting one individual across several separate camera views is tough when angles differ or parts of the body are blocked. Noticing consistent patterns becomes harder if someone moves oddly or gets hidden temporarily. Our new system handles these issues by combining multi level details guided by overall appearance cues. Instead of treating every pixel equally, it learns where to look based on broader context clues first. Tests confirm strong performance, reaching 91.0% correct matches on MARS without relying on perfect conditions.

2.Jia, M., Cheng, X., Lu, S., & Zhang, J. (2022). "Learning Disentangled Representation Implicitly Via Transformer for Occluded Person Re-Identification".

Abstract: When parts of a person are blocked in images, matching them later gets messy because features line up poorly or pick up noise. Most current techniques try hard to align these features but often fail when things go slightly wrong. Instead of forcing alignment, our approach - called DRL-Net - sidesteps it completely by untangling what matters for identity from what comes from blockage. Built around transformers, the system uses broad contextual logic to make sense of incomplete appearances. It learns clean identity signals by pushing unwanted occlusion traits into separate channels via contrastive patterns and statistical independence rules. On standard tests where people are partially hidden, DRL-Net beats top-performing models by clear margins.

3. (Springer) Zhang, Y., Kang, Y., Shen, J. (2022). "Modality-synergy complement learning with cascaded aggregation for visible-infrared person re-identification".

Abstract: One big hurdle in matching people across visible and infrared images? The systems often struggle because the two types of data look so different. While many current approaches try to find shared traits between them - or build middle-ground representations - here's a new idea takes shape. Instead of treating each format separately, a method called MSCLNet comes into play. It works by linking information step by step through layers designed to blend signals smartly. This structure builds up details gradually, fitting pieces together like layers in time Light comes in different forms, one you see, another felt as heat. Step by step, the method sharpens details within small groups, then across similar types, finally setting apart distinct kinds. Tests run on two standard collections of images show this network does better than recent approaches, hitting 76.99% correct first guesses and 71.64% average precision when matching pairs on one setup.

4.Xu, B., He, L., Liang, J., & Sun, Z. (2023). "Learning Feature Recovery Transformer for Occluded Person Re Identification".

Abstract: When parts of a person are blocked from view, recognizing them again becomes tough because key details go missing or get messy. Our approach uses a system called Feature Recovery Transformer, built to fix these gaps by linking what can still be seen across images. Instead of just comparing everything, it zeroes in on areas shared between photos to keep matches clean. From there, the model fills in hidden features by borrowing traits from similar examples stored earlier. It pulls data from the closest k matching entries to rebuild a full profile. Tests prove it works much better than current models when dealing with both partial and whole-body views. On Occluded-Duke, its accuracy jumps ahead in ways others haven't matched.

5.Xu, T., Chai, T., et al. (2022). "Video Person Re Identification Using Attribute-Enhanced Feature Presentation".

Abstract: This method boosts video person matching through richer attribute details. Instead of just identity clues like clothes or gender, it also uses unrelated traits such as pose and movement. A new component sharpens attention on key body areas while fading out messy backgrounds. Another piece adjusts learning by treating different camera angles and motions as less important. Tests show the system beats current top models on two major benchmarks - MARS and DukeMTMC-VideoReID

III. Flowchart Of System Architecture

Starting off, the flowchart shows how a system spots objects or people using Python and OpenCV along with an RGB-D camera and ArUco tags. Instead of jumping straight in, it pulls together needed tools like the marker finder, object recognizer, plus sets up the depth-sensing camera. Connection check comes next - camera has to be linked before anything moves forward. When no link exists, everything pauses, just waiting till either the device connects or someone hits stop. Only after confirming the hook-up do the color and depth images begin streaming in from the hardware. Color data feeds into visuals work; depth info gives distances for whatever gets spotted. Then markers pop up in view - their corners get pinpointed so they can guide real-world scaling and layout setup.

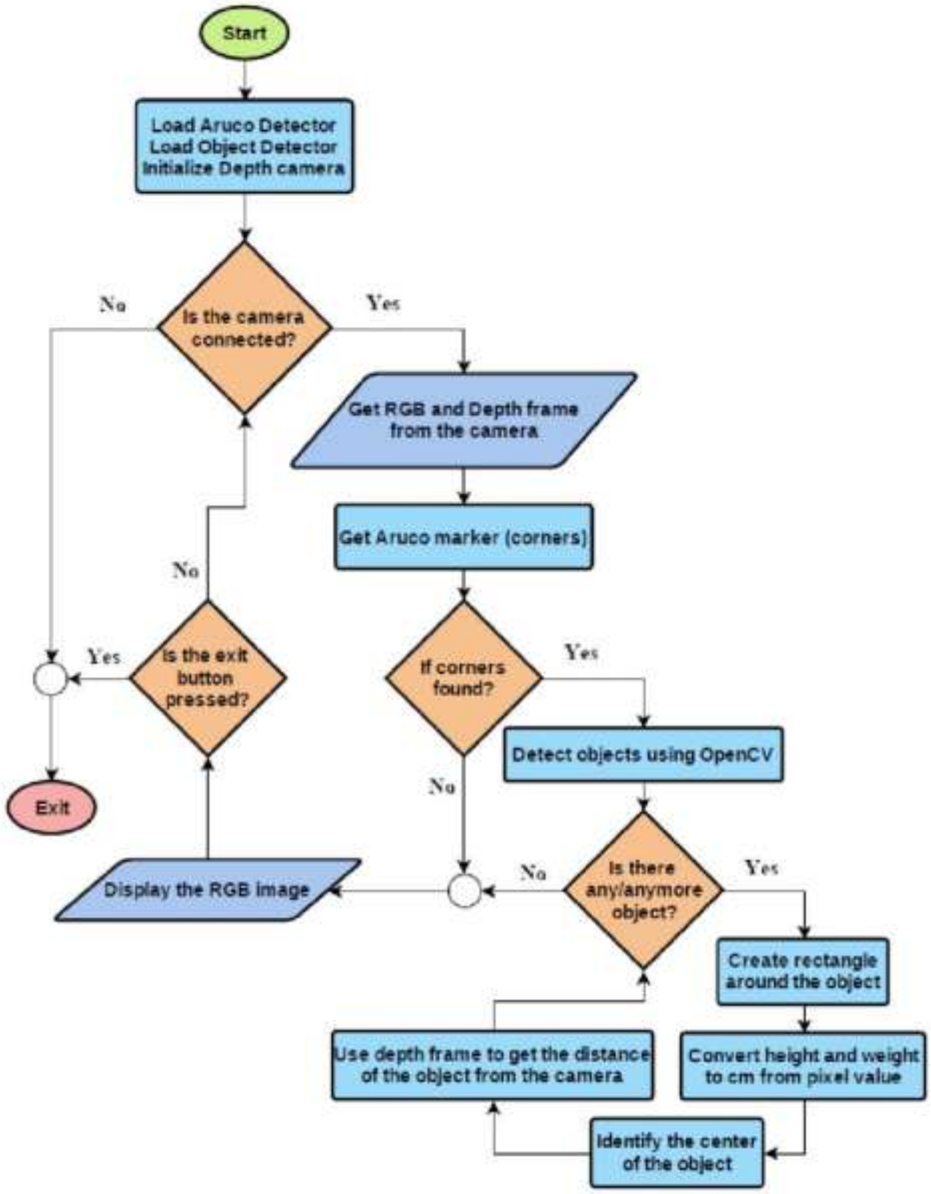

Fig.3 SYSTEM ARCHITECTURE

IV. Working Of The System

From capturing footage to spotting faces, sorting out who is where happens step by step. One frame at a time, detection begins before anything else shows up. After that, features get pulled from each face seen moving through the scene. Matching those traits to known people turns pixels into names eventually. Code written in Python drives it all, linking tools that handle vision tasks. OpenCV supports the heavy lifting behind how images become data worth checking.

1.Starting off, the system grabs visuals through a camera - either live feed or stored recordings. Instead of static shots, it pulls continuous frames using OpenCV for immediate processing. A stream flows in, broken into individual pictures the software can examine one after another. Each moment captured becomes data ready for analysis without delay. From beginning to end, timing stays tight so nothing gets missed during intake.

2. Starting with every recorded image, adjustments happen before analysis begins. A smaller size comes next, cutting down processing load. Noise gets removed through filtering that smooths variations without losing detail. Turning colors into shades of gray follows, simplifying data for better clarity. Brightness levels shift slightly to highlight hidden edges. Each change prepares the image for clearer recognition later.

3. People show up in busy scenes. These get spotted using older tricks, including Haar cascades or HOG paired with SVMs. Newer setups lean on deep learning tools - think YOLO or SSD instead. OpenCV brings them together so more than one person can be seen at once. Even when packed tightly, detection still works.

4. A person shows up first. Then their unique traits get pulled out to stand for who they are. Think face details, how the body looks, what clothes they wear, or surface textures. These signals come alive through smart systems built on Convolutional Neural Networks. Such networks find complex patterns without tripping over changes in angle or light. Tough conditions do less harm now.
5. Features pulled from an image get checked against those saved in a database to figure out who someone is. When there's a close enough fit, that person gets tagged with a name. No match means they stay labeled as unidentified.

Moving through crowds, systems rely on Re-ID methods to link one person seen in separate video clips or angles. These approaches help follow people even when views change or blockages occur.
6. When people move through a busy space, keeping track of who is who becomes tricky. Instead of losing someone between camera shots, systems often rely on patterns like center-point paths or smart prediction tools. Even if one person briefly disappears behind another, these techniques piece together their route. Movement gets messy, yet each step can still be linked back to the right individual.
7. Boxes appear around detected people, outlining their position. Following each box, names or numbers show up - sometimes with how sure the system is - pinned right onto the moving footage. These details stick to the screen through tools built into OpenCV, quietly tagging along as things shift frame by frame.
8. Keeps working through new video clips without pause, only ending when someone hits stop or a quit button. Real time checks on busy areas happen because of this steady flow. Running nonstop makes live oversight possible during active periods. Stops immediately if told, otherwise it goes on watching.

V. Applications And Performance Considerations

From homes to hospitals, voice-driven robotic helpers show up in everyday tasks, yet their success hinges on how well they catch words amid chatter. Though quick replies matter, delays creep in when connections waver or hardware lags behind demand. Even if understanding speech seems smooth indoors, outdoor din often trips them up. While energy use shapes long-term function, breakdowns still surface under heavy loads. Reliability isn't guaranteed - especially where signals fade or computing muscle runs thin.

Ref	Method	Focus Area	Strengths	Limitations
[1]	Multigra nu-larity Attention Re-ID	Video based Re-ID	Strong temporal and global features	High computation
[2]	Transformer based Re-ID	Occlude d Re-ID	Effective occlusion handling	Large data requireme nt
[3]	Positional Transformer	Visible– Infrared Re-ID	Robust multi sensor recognition	Model complexity

Fig.4 Literature Comparison Table

VI. Conclusion

Looking into how people are spotted in busy scenes, this review used Python along with OpenCV tools. Instead of just listing methods, it compared older ways to newer ones powered by deep learning. Even though some models handle partial blocking better, they often struggle when lighting shifts suddenly. Changes in body position tend to trip up systems that work fine on straight-on views. Crowded spaces add noise, making detection less reliable even for advanced setups. Newer ideas built around Transformers do recognize individuals more correctly than before. Still, these need powerful machines plus vast amounts of labeled examples to run well. While progress is clear, heavy demand for computing power remains a stumbling block.

One way to speed things up is using OpenCV for live video analysis, combining detected objects, distance data, because markers help fine-tune alignment. Though less precise than heavy neural networks when recognizing people, this setup still works well enough outside labs, especially where quick results matter more. Next steps might explore smaller AI models that run faster, better ways to track folks when others block the view, along with methods keeping identities private in busy areas.

References

[1]. Sun, D., Huang, J., Hu, L., Tang, J., & Ding, Z. (2022). "Multitask Multigranularity Aggregation with Global Guided Attention for Video Person Re-Identification". IEEE Transactions on Circuits and Systems for Video Technology, 32(11), 7758–7771.
[2]. Jia, M., Cheng, X., Lu, S., & Zhang, J. (2022). "Learning Disentangled Representation Implicitly via Transformer for Occluded Person Re-Identification". IEEE Transactions on Multimedia, 2022.
[3]. (IEEE) "Structure-aware Positional Transformer for Visible-Infrared Person Re-Identification" — Chen et al., IEEE Transactions on Image Processing (2022). Useful when multi-sensor surveillance is used in crowds.

[4]. (Springer) Zhang, Y., Kang, Y., Shen, J. (2022). "Modality-synergy complement learning with cascaded aggregation for visible-infrared person re-identification". (ECCV 2022 workshop / Springer LNCS proceedings) — cross-modality features for robust Re-ID (useful in night/day crowded scenarios).

[5]. Xu, B., He, L., Liang, J., & Sun, Z. (2023). "Learning Feature Recovery Transformer for Occluded Person Re Identification". (paper published on IEEE arXiv → then journal / IEEE venues follow; key occlusion technique for crowds).

[6]. Xu, T., Chai, T., et al. (2022). "Video Person Re Identification Using Attribute-Enhanced Feature Presentation". IEEE Transactions on Circuits and Systems for Video Technology (2022). (video Re-ID methods that exploit temporal/attribute cues in crowded videos).

[7]. (Elsevier / ScienceDirect) Dou, S., et al. (2023). "Human co-parsing guided alignment with multi-task learning for occluded person Re-ID". Elsevier / Pattern Recognition or Image Processing journal (2023) — occlusion-aware alignment techniques.

[8]. (Elsevier) Xu, R., et al. (2023). "Person re identification based on improved attention mechanisms". Neurocomputing / Elsevier (2023) — attention/transformer variants for Re-ID in cluttered scenes.

[9]. (Springer) Dong, N., et al. (2024). "Multi-View Information Integration and Propagation for Occluded Person Re-Identification". Neural Computing & Applications (Springer) — uses multi-view (multi-camera) information to resolve occlusion — very relevant to crowded multi-camera setups.

[10]. (Springer) Zhao, R., et al. (2024). "Situational diversity in video person re-identification". Springer Journal (Engineering with Computers / Intelligent Systems type) — new video Re-ID dataset MSA-BUPT and techniques for complex, crowded urban scenes.