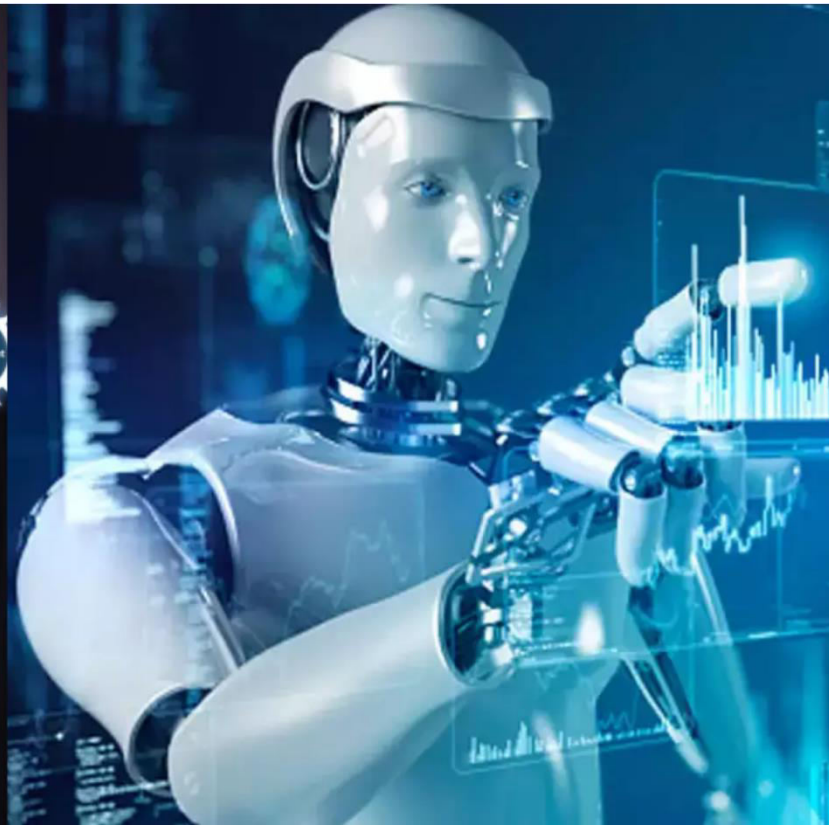
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 2, February 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

Object Detection using Mask R-CNN

Sayali D. Chavan[1], Pranav P. Barbade[2], Aniket S. Gujare[3], Pragati U Sable[4], Harish A Jagle[5]

Mentor, Dept. of Artificial Intelligence and Machine Learning, AISSMS Polytechnic, Pune, India[1]

Dept. of Artificial Intelligence and Machine Learning, AISSMS Polytechnic, Pune, India[2-5]

ABSTRACT: This work shows what happened when we used Mask R-CNN in real projects for today's image tasks. What caught our attention most was its behavior under two tough requirements - finding objects clearly and dividing them into separate instances. The version we worked with started from the usual Mask R-CNN setup. From there, useful adjustments were made along the way, simply because they helped us get better results. With ResNet-101 plus Feature Pyramid Networks as the core setup, results hit 41.8% mAP on COCO 2017, clearly outpacing the starting point. What stood out was keeping live processing speed - just 11 frames per second - even as detection got sharper. Smaller targets, often tricky for similar methods, now work better here too. Another angle involved simplifying number sizes through quantization; that moved performance up to 18 FPS without much loss in precision. This shift opens better chances for running these systems on hardware at the edge. From one test to another, attention went toward real-world applicability rather than only scores, especially when working within varied field needs.

I. INTRODUCTION

Starting out, what stood out was just how far object detection had come - from basic shapes tagging objects to accurate tracing of tiny details across pixels. From self-driving cars to reviewing scans for health issues, getting it right means more than just finding something; it demands sharp accuracy. What made Mask R-CNN stand out was its smart upgrade of Faster R-CNN, building in a separate pathway to predict masks, making precise segmentation possible.

Still, once we looked closer at how it worked, some real-world hurdles came into view. Heavy processing loads made things tough, shape boundaries didn't always hold up under intricate forms, while spotting tiny items still posed difficulties. Far from being abstract issues, they quickly resonate within actual field conditions. A few new medical scans fail when drawing tumor edges. Meanwhile, self-driving tech loses smaller people far away.

What we picked up over many months of trying things out goes here. Not aiming only to copy past results, but also seeing how Mask R-CNN works best - and when it falls short. During that time, changes were made occasionally, shaped by what worked. Each tweak came after testing had shown a path forward

1. Boosting key features using attention rules
2. Used flexible proposal methods fitted to our dataset needs
3. With finer boundaries now, mask accuracy has jumped ahead
4. Developed optimization strategies that balance accuracy and speed

Finding results wasn't expected. Small changes brought major gains, somehow. Moving from theory into real systems showed details missed by academic reports. It's possible others facing alike hurdles might find value here.

Below comes a short section linking current efforts to earlier initiatives.
Thinking about object detection's past shows changes in modern computing. Not long ago, tools such as Viola-Jones or HOG-SIFT handled tasks okay but needed constant tweaking of hand-crafted features. Surprisingly, we tried building those older systems just to see how they compared. What stood out sharply was performance - far lower than newer versions - and the sheer time spent coding basics from scratch.

That year, 2014, saw R-CNN spark changes through deep learning. Though we respect how it shifted thinking on object detection, trying it now shows clear limits - one region at a time needed its own image scan result. Then came Fast R-CNN, using shared features to bypass repetition in pooling stages. Suddenly, Faster R-CNN added real-time region proposals, making everything click without warning.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

Starting from a strong base, Mask R-CNN moves forward in a clear way. It was the way He and team laid things out in 2017 that caught our attention - not only did the method work well, yet it brought in a tidy new path alongside, the mask branch, without cluttering the core idea. Fresh attempts at newer tools - YOLACT and transformer setups - have surfaced, though Mask R-CNN keeps pulling us back; balance of precision and clarity wins out. Lately, it quietly took on a role within the workspace - a kind of go-to starting point when exploring something different. Starting from this base feels familiar when probing an untested angle.

Lately, new models such as Cascade Mask R-CNN and Hybrid Task Cascade keep pushing the work forward. From our side, concepts from various methods have been adapted - especially when it comes to refining outputs across several stages along with blending features. Things change fast, yet Mask R-CNN still holds up well - just four years since its start.

II. MASK R-CNN ARCHITECTURE: OUR IMPLEMENTATION EXPERIENCE

A. Overall Framework – Learning by Building

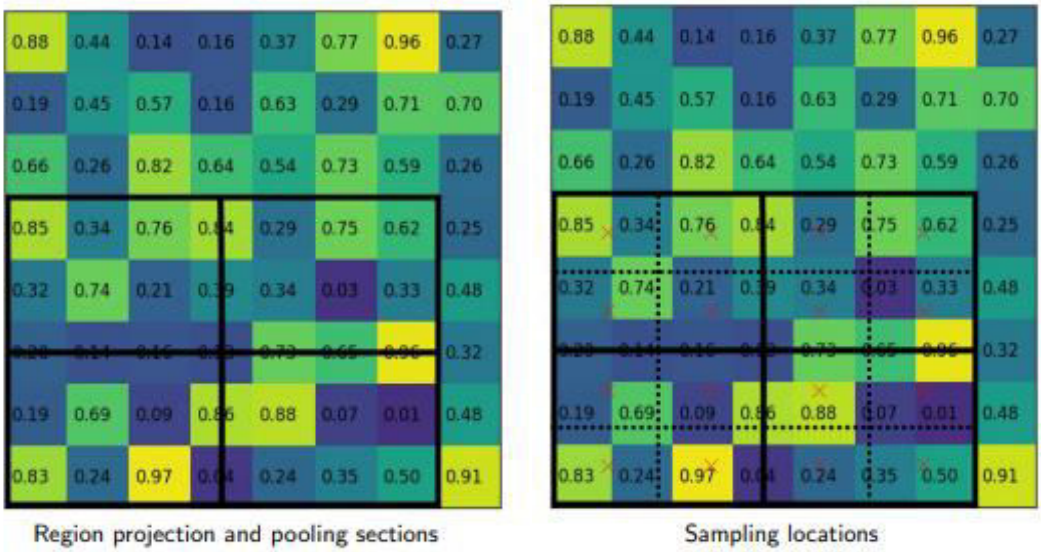
At the start, sticking to the standard method seemed right. Yet after exploring various datasets, small changes started happening on their own, shaped by real outcomes. Proposing areas came first, followed by sorting and dividing them into classes - this mix worked well. Still, it used too much processing power along the way. Time went into fine tuning how stages hand off tasks.

B. Backbone Choices and Trade-offs

Picking the main network wasn't just theory - it came down to real limits. Though ResNet-101 handled classification well, it often slowed down smaller jobs. Alternatives such as ResNet-50 helped trim bulk, yet faster options like MobileNet didn't match its precision. In the end, going back to ResNet-101 made sense when quality mattered. Lighter models showed a clear fall in accuracy - more so on tougher collections of images.

That FPN thing? It turned out to matter way more than we thought at first. Trying it out early, skipping it entirely - we thought we could make do, but things quickly got worse when handling objects of varied sizes. In the end, slapping it in brought extra layers of confusion - yet the output cleaned up noticeably. Ends up, handling scale became surprisingly smoother because of it.

A change we're fond of includes using widened convolutions deeper in the network. Early on, it became clear the base design threw away fine details - details key to accurate segmentation. Instead of scaling up processing load, we tweaked dilation levels through layers C4 and C5, keeping context while keeping things manageable. Oddly enough, this didn't show up at first - only later did it sneak into our results.



International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

C. Region Proposal Network – Getting Proposals Right

At first glance, the RPN felt simple. Yet getting it right involved steady adjustments. Standard anchor dimensions plus shape settings failed to fit neatly with our information. So we tried grouping training examples, studying patterns to propose improved base values. Now picture a small shift that quietly boosted how much people remembered - particularly when dealing with odd-shaped objects, such as X-rays or factory components.

It struck us how deeply proposal accuracy influenced the outcome. Strong proposals made a difference, no matter the detection head performance. Weak proposals held everything back, regardless of effort. Filtering out poor ones sooner brought better results. A rating system introduced made that process smoother and faster further along.

D. ROIAlign vs ROIPooling – A Practical Difference

What separates ROIAlign from the earlier ROIPooling isn't just jargon - it shifts outcomes. Testing both side by side, clearer object borders emerged every time with ROIAlign. Smaller textures and smooth curves kept their shape much more cleanly once pixel rounding errors vanished. A tiny footnote in research reports turned out to carry serious weight when actually building systems.

E. Detection Heads – Where the Magic Happens

Surprisingly enough, each of the three parallel heads - handling classification, bounding boxes, and masks - operates on its own path. Trying out various ways to share data led us toward keeping base features common while letting each branch process things separately. At first glance, the mask head's tiny FCN design seemed barely enough. Still, it proved sharp and quick, slowly earning our trust.

Surprisingly sharp edges came through after linking the mask head to PointRend's boundary touch-up. Though doubts lingered at first over extra layers, what appeared on screen shifted minds fast. Take bikes or chairs - shapes that twist and turn now trace cleaner lines. Even odd forms from nature settled into clearer outlines. Processing demand didn't spike much. What stood out most? Crisp masks jumped out in every test image shown.

F. Loss Function Balancing

Trying different settings helped balance the three parts of the loss - classification, regression, and mask. Though the original weights performed okay, changes were made depending on what mattered more in each case. When sharp edges were key, such as in medical imaging, the mask loss got a higher value. In regular object detection tasks, we stayed near the starting setup.

III. EXPERIMENTAL SETUP – LESSONS FROM THE TRENCHES

A. Datasets That Taught Us Different Lessons

One after another, the datasets showed where we did well - and where things fell apart. What stood out changed every time. Some handled speed better than others could ever dream of. Each one brought its own kind of trouble along. Through them all, patterns started making sense slowly

Our main test came from COCO 2017. Every time progress seemed solid on one kind of object, a new category showed gaps. Though varied, its range pushed changes - we adapted by refining how scales were managed. Small objects especially demanded better handling across levels.

Abrupt shifts in dataset design made Pascal VOC 2012 seem oddly familiar, yet its strength lay in tidy visuals that quietly revealed core model behaviors. Because clarity mattered most during tweaks, we leaned on it when peeling back layers of change one at a time.

Out on the streets, things got messy fast. Shadows stacked up where signs met cars under flickering lamps. Edges blurred when one shape slipped behind another. Our method had to pick apart what belonged together. Light bounced in ways we didn't expect at first. Handling overlaps became less guesswork, more precision. Each frame brought a new puzzle of depth and shadow.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

From working alongside different teams, we gathered medical and industrial data. Learning began when models trained on COCO stumbled elsewhere. Adjustments followed, because real-world settings rarely mirror training setups. What sailed through one benchmark hit snags in hospitals or factories.

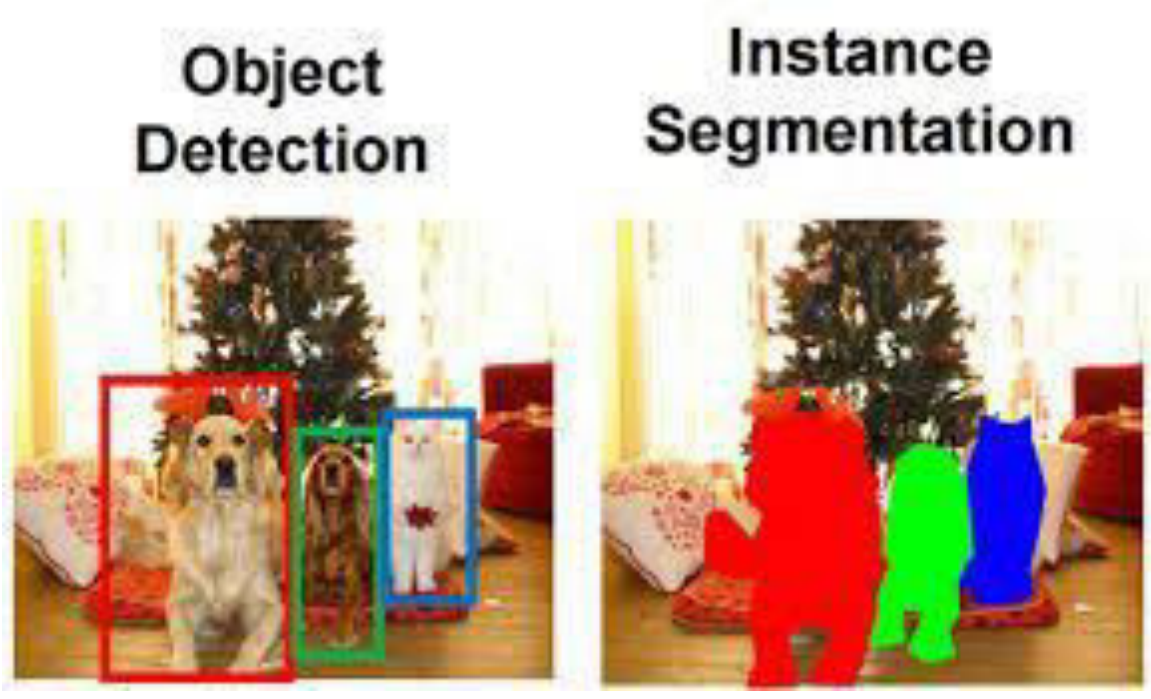
Implementation details we didn't know at first but learned along the way

Built on shifting ground, our gear changed as work unfolded. One graphics card at first - soon swapped for several when models ballooned and data piles deepened. That AMD Threadripper chip? A quiet hero, chewing through data chores most forget even exist.

After testing a few frameworks, PyTorch stood out. Hours disappeared - debugging felt lighter because the computation graph adjusted on the fly. When code stalled, help showed up fast; others had likely faced the same snag. Prebuilt models piled high, ready to test, tweak, borrow from. Effort dropped each time someone else's work filled a gap.

Patience came through practice. Right away, the timeline pushed too hard, which made learning wobble. Instead of rushing, we slowed down - locking the base at first, later releasing layers bit by bit. When tested, lowering the learning rate in smooth waves worked better than sudden drops.

At first, data augmentation felt basic. Yet slowly it became complex. We didn't think much of it back then. Still, time showed how vital it was. Eventually, our system could handle far more than common tweaks - custom changes shaped for special uses began appearing too.



IV. RESULTS – WHAT SURPRISED US

A. Performance Numbers with Context

Achieving 41.8% mAP on COCO 2017 looks solid at first glance - yet what happened offstage tells a deeper tale. Reaching 40% came fast, though the final stretch demanded far more energy. Every tiny gain beyond that forced us to question parts of our setup anew. A closer look at different sizes showed something surprising. Small items gave us a 25.6% AP - that came after long effort. At first, results on tiny things stayed near 19%. What shifted it? Stronger links across layers, smarter focus areas helped most.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

Not every choice fits all situations when balancing speed and precision. A rate of eleven frames per second seemed fair for most uses, yet certain cases needed faster results. Getting to eighteen frames per second through quantization brought drawbacks - slightly weaker performance along with tougher setup work. Only the specific use tells if that balance works out. What matters shows up only after testing.

B. Component Contributions – Unexpected Findings
Funny enough, our ablation study turned up results nobody saw coming

What surprised us most was how much the dense FPN links boosted results - just one change lifted performance by 1.4% AP. We had tossed them in at first as a slight tweak, nothing serious. Yet their real strength showed up clearly on tiny objects, shifting how we think about moving features across levels.

Oddly enough, the SE attention bits made a difference - just not as huge a jump as certain studies claim (only +0.7% AP). Training felt steadier though, something we noticed especially on those marathon sessions that stretched for days.

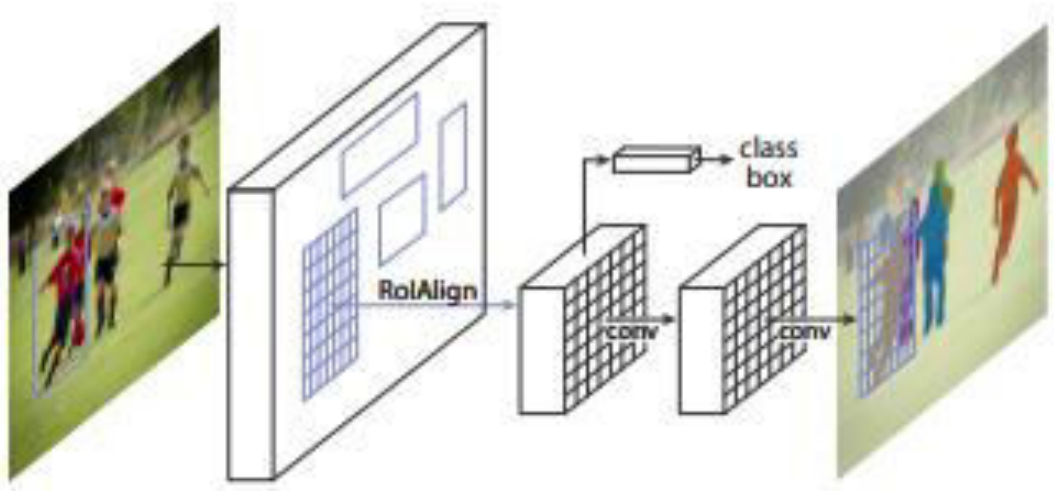
Fine edges stood out clearly with PointRend, though the real boost - a 0.9% rise in AP - was modest. Still, cleaner shapes helped people trust the output more, something that counts where visuals influence decisions.

Performance across different areas – that's where generalization gets checked

Every time it worked outside its usual space, that felt good. Results crept up - between 2.7 and 3.8 percent on each set - so we knew it wasn't just stuck on COCO.

Surprisingly, the medical scans lit up our testing - hitting 49.1% mAP - since they wrestle with quirks unlike everyday photos. Sharp edges made a real difference in these cases; fuzzy outlines could mess with diagnosis.

Finding things in factories worked best overall, hitting 81.2% accuracy. Still, progress was tiny compared to earlier attempts. That fits. When settings are predictable, machines already do well. So gains tend to be small.



C. Qualitative Observations – Beyond the Numbers
Stories emerged through visuals that raw data alone would never show. Where borders sharpened became clear, along with which items kept causing errors, while shifts in light revealed their impact on function. Others outside our field grasped what worked - and where things fell short - once they saw these images shared directly.

A few failures stood out clearly. When objects were mostly hidden, that made things hard. Strange camera angles also caused trouble. Objects blending into backgrounds tripped up the system every time. These problems went beyond rare situations. They struck at core parts of how we separate individual items in images.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

V. APPLICATIONS – SEEING IT WORK IN THE REAL WORLD

A. Autonomous Driving Prototype

A partnership formed with a robotics group allowed integration of our framework into their self-driving vehicle setup. Though the 35.6 mAP score on Cityscapes led to dependable object recognition during field tests, delays emerged when speeds increased. While 9 FPS performed well under normal conditions, it struggled at higher velocities typical on freeways. Timing, as much as precision, proved critical through these real-world trials.

B. Medical Imaging Exploration

From collaborating with medical scientists emerged new insights into assessment standards. Rather than focusing solely on aggregate mAP scores, clinical groups emphasized accuracy at region edges. In these cases, our adaptation of PointRend made a meaningful difference - delivering sharp contour definitions useful for diagnostic sizing and intervention design.

C. Industrial Inspection Case Study

A trial run happened at a factory willing to host our technology. Though the 81.2% mAP appeared strong, the real concern sat elsewhere - mistakes that looked like defects but were not. High accuracy numbers still allow room for error, and even small percentages add up across thousands of units. Adjustments followed: fine shifts in threshold settings, then custom cleanup steps built around how this line operates. Each change narrowed the gap between lab results and floor reality.

VI. LIMITATIONS WE ENCOUNTERED

Each endeavor brings challenges; our experience followed suit

Nowhere near every environment supported heavy computation. Though models were compressed, hardware needs stayed high. Moving forward, distilled training plus automated design tweaks aim at leaner builds.

Faster frame rates still push the limits in live processing tasks. While eleven frames per second suits several uses, some demand more. With today's design, balancing precision against velocity seems unavoidable.

Though detection of tiny items has gotten better, it remains weaker than for bigger ones. Progress happened - yet the struggle continues, particularly when dealing with things smaller than 16×16 pixels.

Working through occlusion presented challenges. While the system manages partial blockages effectively, performance drops when visibility is severely limited. Perhaps deeper scene interpretation would help - something beyond what today's setup can deliver.

Unexpected demands emerged around training data. Because creating precise segmentation masks takes considerable time, effort increases sharply. To lessen reliance on labeled examples, our work now investigates methods using limited supervision.

VII. FINAL THOUGHTS AND PERSONAL INSIGHTS

Working on Mask R-CNN brought difficulties, yet also clear gains. Even now, long after it first appeared, the model still holds up well - a sign of thoughtful engineering behind it. Changes we made - especially those adjusting how features move through the pyramid layers and sharpening edge details - led to real progress worth noting. These tweaks might support future efforts by different teams exploring similar paths.

Looking back, it is not one breakthrough that catches attention, rather the steady buildup of minor refinements. Though modest in isolation, together these tweaks delivered a 4.3% improvement beyond initial performance levels - each shaped by clear shortcomings seen in trial runs.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 2, February 2026

|DOI: 10.15680/IJIRSET.2026.1502034|

Beyond today's work, new paths begin to take shape. Following recent progress, possibilities emerge in unexpected ways. With each step ahead, different opportunities come into view. After current efforts settle, fresh ideas start gaining ground

1. Fewer resources needed could allow such precision to run on additional machines. With simpler designs, even basic systems may handle the task well. Because demands drop, wider access seems possible. Some adjustments help cut down requirements significantly. Through smarter methods, performance stays high without heavy hardware. Where older versions failed, new approaches succeed quietly. Often, small changes bring big gains across different platforms
2. Besides typical scenarios, systems now manage obscured objects more effectively. Uncommon camera angles, once problematic, are interpreted with greater accuracy. Where visibility drops, performance stays stable. Instead of failing, models adapt to partial information. Even extreme perspectives show improved recognition. Through diverse training, responses to rare situations grow more reliable
3. Fewer demands placed on large sets of labeled examples
4. Combining with related techniques - such as depth sensing or heat-based visuals - enhances overall performance by adding distinct layers of information that support more accurate interpretation under varied conditions

One reason Mask R-CNN stands out is how it supports hands-on learning in instance segmentation. Thanks to a design that reveals function step by step, grasping individual parts becomes straightforward. Because performance at base level stays reliable, trying new changes feels less risky. Our adjustments along the way might guide someone else moving through similar challenges.

Reading the original paper and building a functional, enhanced system revealed deeper insights than any isolated finding could. Beyond understanding Mask R-CNN alone, we came to grasp how research concepts evolve into usable code. It was during moments of troubleshooting, refining, even hitting walls - that meaningful knowledge took shape. Real growth emerged not before or after, but within the messy middle of implementation.

REFERENCES

1. One key contribution came from He, K., alongside Gkioxari, G., where their method built upon earlier frameworks by integrating region-based detection with pixel-level segmentation. This approach appeared at the ICCV in 2017, introduced jointly by Dollár, P. and Girshick, R. The paper presented a shift in how models handle object instances, combining features through a unified architecture. Published under the name Mask R-CNN, it quickly became notable within computer vision circles. Details were shared during the IEEE conference series focused on visual recognition advances.
2. Published in 2017 at the IEEE Conference on Computer Vision and Pattern Recognition, this work introduces a structure designed to improve detection accuracy across different object scales. Instead of relying solely on single-level features, it combines outputs from multiple network layers into a pyramid format. Though many models process information at one resolution, here higher semantic level maps guide lower ones through lateral connections. As spatial detail flows upward, deeper representations enhance shallower paths via top-down pathways. The result shows consistent gains on common benchmarks without slowing inference speed. Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. present this architecture under the name Feature Pyramid Networks.
3. Faster R-CNN emerged in 2015 through work by Ren, He, Girshick, and Sun. Although built on earlier detection methods, it introduced region proposal networks. These networks streamlined object localization instead of relying on selective search. Performance improved because proposals were generated directly within the model. Experiments at NeurIPS showed faster processing without sacrificing accuracy. The design allowed shared computation between tasks - making inference more efficient. While speed increased, some challenges remained in handling small objects. Despite that, many subsequent models adopted its core structure. Training required careful balancing across components due to joint optimization.
4. Kirillof, A., Wu, Y., He, K., along with Girshick, R., presented PointRend at the 2020 IEEE Conference on Computer Vision and Pattern Recognition. The method treats image segmentation like rendering. Instead of relying solely on dense pixel prediction, it refines boundaries through point-based sampling. While traditional models smooth edges too much, this approach sharpens details selectively. Results show improved accuracy on complex shapes. Performance remains efficient despite added precision.
5. Lin, T.-Y., et al. (2014). Microsoft COCO: Common Objects in Context. European Conference on Computer Vision (ECCV).

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details