ENHANCING OBJECT DETECTION PERFORMANCE USING TRANSFER LEARNING IN LOW- RESOURCE ENVIRONMENT

Sonalee Singh[1], Mr. Pravin Kumar Pandey[2]

[1]*Master of Technology, Computer Science and Engineering, Veer Bahadur Singh Purvanchal University, Jaunpur, Uttar Pradesh, India*

[2]*Assistant Professor, Department of Computer Science and Engineering, Veer Bahadur Singh Purvanchal University, Jaunpur, Uttar Pradesh, India*

---***---

Abstract - *Object Detection (OD) is one of the most important parts of many smart systems. These include self-driving cars, medical image analysis, precision agriculture, and edge-based video surveillance. There are several obstacles to implementing highly accurate OD algorithms in Low Resource Environments (LRE). One major challenge is that LRE have limited computing resources, limited memory, limited access to large amounts of labeled data, and require OD results in near real-time. Transfer Learning has been shown to be effective at addressing some of the above-mentioned challenges. By leveraging existing pre-trained knowledge and decreasing the amount of time required to train new models, TL enables the development of high performance OD models that can run on LRE. This literature review will look into the evolution of OD techniques, provide a comprehensive evaluation of the most recent advances in using TL to develop OD models, and evaluate how TL-based OD model performance compares to traditional OD models in LRE. Through a thorough literature review, this study demonstrates several key strategies that can help address the challenges of OD in LRE including using lightweight OD architectures; knowledge distillation (KD); selective fine-tuning (SFT); domain adaptation (DA), and data-efficient learning (DEL). This study also provides examples of the use of each strategy in multiple areas of application (agriculture, surveillance, health care, remote sensing, and Internet-of-Things (IoT)) demonstrating the practical value of these techniques. The study concludes with an emphasis on the need for future research to continue developing adaptive and scalable TL frameworks that can enable the widespread adoption of high performance OD models suitable for various real-world applications.*

Key Words: **Transfer Learning, Object Detection, Low-Resource Environments, Lightweight Models, Knowledge Distillation, Domain Adaptation, Edge Computing, Few-Shot Learning**

1. INTRODUCTION

Computer Vision has been at the center of the explosive growth of Object Detection in recent years as it has enabled computers to identify and locate objects of interest inside images and video recordings. The field of Object Detection has enabled the development of an increasing number of automated perception systems for use in various industries by allowing the automation of decision making in the same environment where data is collected [1]. Since Deep Learning technologies have made possible rapid advancements in accuracy and capabilities of Object Detection models over the past ten years, it has created new opportunities in both the research community and the industrial community.

1.1 Brief Overview of Object Detection and Its Growing Role in Real-World Applications

From a hand-crafted feature based methodology to today's advanced deep learning framework object detection has experienced significant improvements in accuracy and processing time [2]. Today's object detection systems such as faster r-cnn, yolov4, ssd and retina net are being embedded into all of our daily life technologies and provide the ability to rapidly process and interpret visual data. The applications for these object detection systems include; autonomous vehicles, intelligent surveillance, medical imaging, precision farming, automated retail and robotics. With the need for smarter systems growing, object detection will continue to be a key technology in areas that require accurate, robust, real-time visual sensing [3].

1.2 Low-Resource Environments Pose Unique Challenges

Even though there have been many developments in object detection, it is still a problem to deploy them in resource-constrained environments. The real world includes a lot of low-resource areas like rural communities, developing countries, small businesses, and IoT devices [4]. In addition to this, they do not have enough resources for high performance computing equipment (GPUs), large amounts of labeled data, reliable network access, or available power. Deep neural networks are resource-intensive, which means they need fast, high-powered GPUs with lots of memory and a long time to train – all things that are not available on most low-power edge devices (e.g., drone, camera, cell phone, etc.). As a result of the above limitations, it is difficult to train and/or deploy state-of-the-art object detection systems at scale; thus limiting their speed and accuracy.

1.3 How Transfer Learning Offers Practical Advantages

Transfer learning is a practical way to get around the problems mentioned above by using pre-trained networks that are trained to learn general image feature representations from large datasets (e.g., ImageNet, COCO, Open Images). By adapting those models through fine-tuning (or partially training) them, transfer learning greatly reduces both the computational costs, time and data requirements necessary to train a model, enabling high performance object detection on low-power hardware with minimal data. Additionally, approaches such as lightweight architectures, knowledge distillation, and domain adaptation make the system even more efficient while maintaining strong detection performance [5].

1.4 Purpose, Scope, and Contributions of the Review

This review aims to investigate how transfer learning will improve object detection in resource-poor conditions (low resources). This study will provide an overview of foundational concepts, a review of recent contributions to the literature and comparative analysis of methods for optimizing detection systems when constrained by both hardware and limited data [6]. In addition to providing practical methodology for improving object detection systems, it identifies typical limitations and reviews emerging areas for research that supports the development of deployable object detection models with scalability and low costs. By combining current state-of-the-art knowledge and research voids, this work hopes to inform future developments which will allow for widely available and sustainably priced AI solutions across many different application domains.

2. FUNDAMENTALS OF OBJECT DETECTION

In computer vision, object detection is a fundamental task, which includes detecting objects inside an image and specifying the location of each object with the use of a bounding box or a segmentation mask. Object detection has been one of the most prominent areas of advancement in computer vision over the years, as it has evolved from conventional machine learning based methods to advanced deep learning architectures. Collectively the advancement of neural networks, availability of larger datasets for training models and improved computing capabilities have created a new area of object detection that can be reliably used by numerous users [7].

2.1 Evolution of Detection Techniques from Traditional to Deep Learning

Traditionally, early object detection used manually designed methods to extract features from images, such as Haar-like features, Histogram of Oriented Gradients (HOG), and Scale-Invariant Feature Transform (SIFT). Typically, they were combined with a variety of classical machine learning classification methods including SVM and AdaBoost. Although useful for simple object detection tasks, these classic methods failed when detecting objects within complex background conditions, varying light conditions, and/or when objects are occluded.

The key turning point in object detection was the advent of Convolutional Neural Networks (CNN's) that allowed the computer to learn hierarchically about an image automatically without the need to use manual feature extraction techniques. This ultimately led to the development of many CNN-based object detection methods such as R-CNN, providing much higher detection accuracy and robustness than previously seen [8]. With continued advances made in architectural design, feature extraction, and faster training times, real time object detection has become possible.

2.2 Key Model Families

Deep-learning-based object detectors are broadly categorized into two major families: two-stage detectors and single-stage detectors, each designed to balance accuracy and processing speed.

2.2.1.Two-Stage Detectors (R-CNN Variants)

The two-stage detector generates region proposals and uses the proposal to classify regions in an object category. This family of detectors includes R-CNN, Fast R-CNN and Faster R-CNN which represent this architecture. The use of a Region Proposal Network (RPN) in Faster R-CNN has greatly reduced processing times as opposed to previous architectures [8]. Two-stage detectors are best used when detection of objects with small dimensions or objects that are overlapped in an image is required. As such, they can be applied to applications requiring detection precision (medical imaging and traffic monitoring).

2.2.2.Single-Stage Detectors (YOLO, SSD, RetinaNet)

A single stage detector bypasses a region of interest proposal stage and predicts object class and bounding box simultaneously within a unified framework. Faster inference is achievable by a single stage detector when compared with multi-stage detectors which include YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector) and RetinaNet. YOLO has become a widely used model due to its fast execution time and ability to be deployed on embedded and edge devices. The focal loss was proposed in RetinaNet to improve class balance while maintaining detection speed. A single stage detector is most suitable for applications where quick decisions need to be made, such as, autonomous navigation and video surveillance [9].

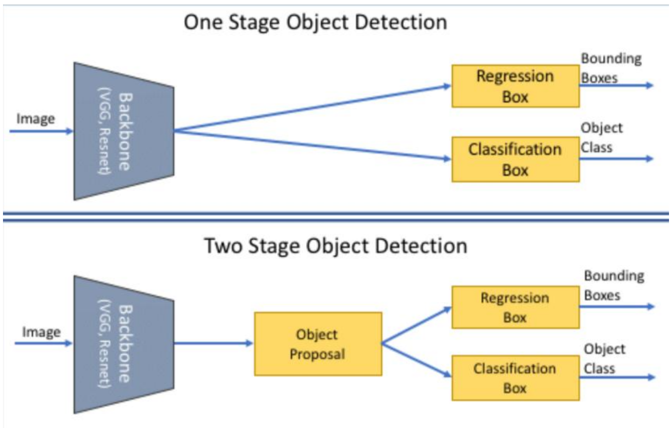

Figure-1: Single and Two-Stage Detector

2.3 Common Performance Metrics and Evaluation Methods

When evaluating object detection models you need to use established methods to measure both how accurate your detector is at identifying objects (classification) and where it has located them on an image (localization). The primary way that object detection models are evaluated is through the use of the Mean Average Precision (mAP). mAP provides a method to evaluate how well your model performs across each class of object, using its ability to achieve precision/recall performance [9-10]. A common evaluation metric to determine if your object detector's predictions are close enough to the ground-truth bounding box to be considered "correct" is the Intersection Over Union (IoU) or Jaccard Index. This provides a clear threshold for what constitutes a successful prediction. Other additional metrics that can provide useful information include FPS (Frames Per Second), Latency and Model Size, which are all particularly relevant when implementing detectors within limited resource environments where Speed/Efficiency are critical. All of these metrics can also be used to compare performance of different models and to select architectures based on performance under real-world constraints [10].

3. TRANSFER LEARNING: CORE CONCEPTS

Transfer Learning is a powerful approach that has been developed as a method to speed up computer vision processing and improves results (object detection), for example, by reducing the need for a large amount of data and long training times. Rather than building a model from start to finish, Transfer Learning allows developers to use knowledge acquired during the training process of an existing model that was trained using a large and varied data set. In addition to providing a strong base of visual feature understanding, the pre-trained model also allows developers to adapt to a new task with limited amounts of data and computational resources. As such, this method dramatically reduces the time it takes to develop and improve results in many cases in limited data environments [11].

3.1 What Transfer Learning Means in the Context of Computer Vision

The concept of transfer learning in computer vision involves using an existing model that was previously trained on a large amount of data (e.g., Image Net, COCO, Open Images) to develop a model to learn a new task or work with new data. In many ways, the pre-trained model has learned generalizable features such as texture, shape, object attributes and has formed a generalized feature representation based on all of the images used during its initial training [11]. As a result of the model having been trained to be able to identify the key elements of these images, when the model's knowledge is applied to a new task, it will require significantly less training time and can also provide good results with much smaller amounts of data. Transfer learning is particularly useful for object detection because manual annotation of data can be expensive and very time consuming.

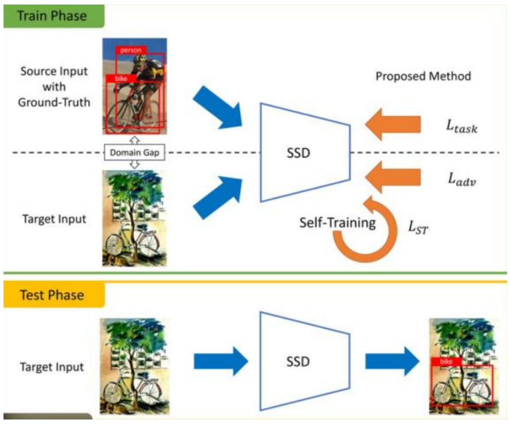

Figure-2: Workflow for transfer learning

3.2 Categories of Transfer Learning: Feature Extraction, Fine-Tuning, and Domain Adaptation

Several transfer learning strategies exist based on the specific task needs and resource availability. The most common methods include feature extraction (involves using a pre-trained model as a feature extractor, so that it's representations are used to feed new layers for classification or detection tasks, but does not modify the pre-trained model), fine-tuning (feature extraction with additional pre-trained layer(s) trained on the target data set), and domain adaptation (when the source data set has significant differences than the target environment, techniques like adversarial learning or style transfer are employed to allow the model to learn to perform well despite differences in lighting, object appearance, etc.).

3.3 Popular Pre trained Backbones and Their Computational Footprints

All contemporary object detection systems utilize an efficient backbone architecture to generate the most optimal feature maps. The pre-trained backbone architectures that have become popular include ResNet, VGG, MobileNet, EfficientNet, DenseNet, and variants of the Vision Transformer (ViT). As each backbone has different architectural characteristics including their respective depths, number of parameters, and computation cost, they can be classified into two categories [12]. First, there are backbones with high accuracy but also high computational cost such as ResNet and VGG, which are more appropriate for a GPU-based setup. Second, there are low-computation backbones such as MobileNet, ShuffleNet, and EfficientNet, which are optimized for low-power computing platforms and edge devices, and achieve the best trade-off between speed and accuracy. Therefore, the correct backbone selection is important because it must take into account the performance constraints.

3.4 Transfer Learning Is Ideal for Low-Resource Conditions

Transfer learning can be particularly valuable when resources are very limited (in terms of computing power, memory, or training data). The fact that pre-trained models have learned general knowledge about features means less training is needed with fewer images (i.e., less time and energy), allowing object detection to run on mobile phones, drones, cameras, IoT devices etc. using only low-power hardware (not requiring expensive GPUs) [13]. Transfer learning allows the combination of layer-freezing, quantization and other approaches to build small, but still accurate object detection systems for use in resource-constrained situations, so transfer learning has an important function in closing the gap between AI possibilities and real world deployment.

4. CONSTRAINTS IN LOW-RESOURCE ENVIRONMENTS

While there have been a number of studies focused on developing and testing the performance of deep-learning-based object detection algorithms for use in various low-resource environments (e.g., rural surveillance systems, embedded IoT devices, agricultural drones and mobile health applications), all of these studies point out the need to improve the efficiency of training and inference processes to enable such models to be implemented in a variety of low-resource settings. This is particularly relevant since current state-of-the-art models exhibit excellent accuracy on benchmark datasets, but were primarily developed and tested using large amounts of data and high-powered GPU processors found in laboratory settings [14].

4.1 Hardware Constraints on Edge Devices and Embedded Platforms

Studies also show that deploying object detection using edge computing hardware causes serious computation restrictions. The edge devices (for example: Raspberry Pi, NVIDIA Jetson Nano, ARM-based processor, drone, and mobile devices) do not have sufficient memory bandwidth, or GPU-acceleration or the necessary parallel processing for many DNNs. In addition to reporting that edge-AI model deployments are challenging with models based on Faster R-CNN and ResNet-based backbones because these models require too much computational overhead and memory usage to perform in real time [15]. Due to these requirements lightweight networks and transfer learning techniques have been developed in order to allow object detection to occur in environments where power consumption and device temperature along with battery-life need to be strictly monitored.

4.2 Limited Training Data and Annotation Challenges

Research also points out that many low-resource environments do not contain large, high-quality labeled sets necessary for initial model development using a new, deep detection model. Collecting large datasets to train models is typically very time consuming and/or cost prohibitive for areas of study such as precision agriculture, medical diagnosis, and remote sensing. Additionally, many field-specific datasets will have problems associated with class imbalances, rarity of target items, or noise in the annotations. It has been demonstrated through research that when a model is trained with insufficient data it tends to fit too well to the small dataset, resulting in poor performance outside the training environment [16]. As a result, transfer learning, few-shot learning, and generating synthetic data are being used in greater numbers as practical methods to address the data-based challenges.

4.3 Restrictions on Model Size, Latency, and Energy Usage

Complex deep object detection architectures have the potential to be millions of parameters, thus requiring substantial amounts of memory and computational resources for both training and inference tasks. The previous work established that these large complex architectures will incur substantially increased latency, which is a challenge to real-time processing within constrained resource environments. High-capacity neural networks in robotics and UAV's battery powered and portable applications consume excessive amounts of power, resulting in reduced operational time. As a result, it is necessary to strike an optimal balance among model size, model accuracy, and inference speed. Model pruning, knowledge distillation, and quantization are popular techniques used in the literature to minimize the amount of computations required by a neural network while minimizing loss of model accuracy [17-18].

4.4 Effect of Connectivity and Deployment Conditions

Reliability of Network Connectivity is a common issue with Low-Resource Deployment Research. While powerful, cloud-based processing relies on stable high-bandwidth connection to communicate image data, which is often unavailable in remote areas of the world. As shown by various studies, network latency, bandwidth limitations, and security issues can limit the use of cloud-based processing for several real-time applications such as emergency response or rural surveillance. In addition to network connectivity issues, environmental factors such as extreme temperatures, vibrations, and low light conditions will also negatively affect the reliability and robustness of the model when it has been deployed.

5. LITERATURE REVIEW

5.1 Overview of Significant Studies Applying Transfer Learning in Resource-Constrained Object Detection

Recent research has shown that transfer learning with lightweight detection frameworks are viable options for implementing object detection in very low resource or very constrained environments. As an example, a study reviewing many deep-learning-based lightweight object detectors for use on edge devices examines different backbone architectures to demonstrate how pre-trained models adapted through transfer learning can be used effectively to deploy on the edge [19].

Another area of research is TranSDet — TranSDet uses transfer learning with a dynamic resolution adaptation scheme to enhance small-object detection when limited datasets exist. By adapting a pre-trained model to detect objects in images at multiple low resolutions, the detection model becomes better able to detect small objects, which is often a requirement in resource-constrained and domain-specific applications [20].

There are also works, such as LSTD (Low-Shot Transfer Detector), that focus on "low-shot" scenarios, i.e., where only a few labeled samples are available in the target application domain. LSTD uses knowledge from a source domain and fine-tunes to the target domain with minimal labeled examples, demonstrating comparable results to full-domain methods even when minimal amounts of training data are available.

More recently, studies like Tiny-DSOD have illustrated the design of extremely efficient detection architectures that are targeted toward resource-constrained devices, e.g., by using depth-wise dense blocks and lightweight feature pyramid networks to provide high quality object detection while minimizing both the number of parameters required and the amount of computational resources required.

Collectively, all of these areas of research support the idea that the combination of transfer learning with model design and optimization are viable approaches to enabling object detection in constrained environments such as embedded systems, edge devices, Unmanned Aerial Vehicles (UAVs), and small-domain datasets.

5.2 Comparison of Methods, Datasets, Preprocessing, and Training Strategies

The reviewed works vary along multiple dimensions: detection architectures, dataset size and type, preprocessing and augmentation strategies, and training protocols. The table below summarizes a subset of representative studies to highlight their differences and outcomes.

Table 1 : Comparison of Representative Studies

Study / Method	Target Scenario / Constraint	Strategy (Transfer / Lightweight / Data-Efficient)	Dataset (s) / Domain	Key Results / Observations
TranSDet	Small-object detection, limited dataset size	Transfer learning + resolution adaptation + modified FPN & anchor module	TT100K-Lite, BUUISE-MO-Lite, COCO	Significant mAP gains over baseline (e.g. +8.0% for Faster R-CNN, +22.7% for RetinaNet on TT100K-Lite)
LSTD	Low-shot detection (few target examples)	Low-shot transfer learning with regularization for background suppression	Various target-domain sets (few samples)	Outperforms standard detectors under limited data, showing robustness in low-shot regimes
Tiny-DSOD	Resource-restricted deploymen	Lightweight backbone + efficient	PASCAL VOC, KITTI,	Achieved ~72.1% mAP with

	t (embedded /low-power devices)	feature-pyramid network (depthwise convs, parameter-efficient)	COCO	only ~0.95M parameters and 1.06B FLOPs — outperforms many earlier lightweight detectors.
General lightweight detectors (survey)	Edge devices, embedded platforms	Use of mobile-friendly backbones (MobileNet, ShuffleNet, etc.), model compression, quantization/pruning	Standard benchmarks (MS-COCO, PASCAL-VOC) + varied real-world domains	Demonstrates trade-off between accuracy and resource usage; good performance with careful backbone and architecture choice.

These studies collectively show how transfer learning and architecture design (lightweight networks, efficient feature pyramids, resolution adaptation) — combined with smart training strategies — enable object detection even when resources or data are scarce.

5.3 Trends, Resource Savings & Performance Gains

Table 2 : Trends in Resource-Constrained Object Detection via Transfer Learning

Trend / Technique	When It Helps Most	Typical Benefit (compared to baseline / from literature)
Pretrained models + transfer learning (fine-tuning or low-shot)	Limited target data; new domain with small dataset	Allows working with few samples; maintains competitive accuracy (e.g. LSTD results) (arXiv)
Resolution adaptation + small-object specialization (e.g. TransDet)	Small-object detection, limited training data	Significant mAP improvement over standard detectors under same constraints (MDPI)
Lightweight backbone + efficient network design (e.g. Tiny-DSOD)	Low-power, embedded or edge platforms	Drastic reduction in parameters/FLOPs; acceptable mAP for many tasks (arXiv)
Model compression / pruning / quantization + transfer learning	Embedded deployment, latency/energy-critical systems	Reduced inference latency and memory footprint; feasible deployment on Jetson, Raspberry Pi, etc. (ACM Digital Library)

5.4.Key Design Features Identified from Prior Studies

Based on a synthesis of prior studies, the following key reading features consistently influence performance in low-resource object detection:

Table-3: Related Reading Features Identified in Literature

Feature	Description	Supporting Studies
Lightweight Backbones	MobileNet, ShuffleNet, EfficientNet reduce FLOPs	Tiny-DSOD, Mittal (2024)
Feature Pyramid Networks (FPN)	Multi-scale object representation	TranSDet
Knowledge Transfer	Teacher–student model compression	LSTD, DSOD
Selective Fine-Tuning	Freeze early layers, adapt higher layers	Few-shot OD surveys
Data Efficiency	Few-shot learning, augmentation	LSTD

5.5 Summary of Literature Review

The study shows that using transfer learning along with efficient architectures, smart network design and domain specific adaptation can create a viable way to deploy an object detection system on constrained devices [22]. As such, while it has been shown that (e.g., LSTD, Tran SDet) with little training data and/or small objects, good enough object detection results have been obtained, also it has been demonstrated (Tiny DSOD) that object detection can be performed at a computational cost low enough to be deployed at the edge.

While many challenges exist including tradeoffs between detection quality and detection speed, and the ability to generalize from one environment to another, detecting small objects under constrained conditions is still less robust than detecting objects under unconstrained conditions. Additionally, very few standard benchmark tests and device deployments exist, which complicates the comparison of the effectiveness of research-based methods versus their translation into practice [23].

In summary, this area of research appears to be a developing field with a great deal of promise for use with constrained resources, but much additional development is required to create robust, generalizable and deployable object detection systems across a wide variety of real world conditions.

6. METHODOLOGICAL FRAMEWORK ADOPTED IN PRIOR STUDIES

Most transfer-learning–based object detection studies in low-resource environments follow a four-stage methodology: (i) backbone pertaining, (ii) transfer learning adaptation, (iii) lightweight optimization, and (iv) constrained deployment evaluation.

A pretrained backbone $f_{\theta s}$, trained on a large-scale dataset Ds (e.g., ImageNet or MS-COCO), is adapted to a target dataset Dt with limited samples by minimizing the target loss:

$$L_t = L_{cls} + \lambda L_{reg}$$

where

$$L_{cls} = -\Sigma\, y_i \log(\hat{y}_i)$$ is the classification loss

$L_{reg} = \Sigma\, \mathrm{SmoothL1}(b_j - \hat{b}_j)$ is the bounding box regression loss (used in Faster R-CNN, SSD, and YOLO variants).

TranSDet further incorporates resolution-adaptive feature learning, where feature maps at multiple resolutions rk are fused:

$$F = \Sigma\, w_k F_{rk}$$

This approach leads to up to +22.7% mAP improvement for RetinaNet on TT100K-Lite under limited data conditions.

7. MATHEMATICAL MODELS AND ALGORITHMS USED IN TRANSFER LEARNING–BASED OBJECT DETECTION

Knowledge Distillation Algorithm (Tiny-DSOD, LSTD)

Let T be a teacher model and S a lightweight student model. The distillation loss is defined as:

$$L_{KD} = \alpha\, L_{det}(S) + (1 - \alpha)\, L_{soft}$$

where

$$L_{soft} = KL(\sigma(z_T / T)\ ||\ \sigma(z_S / T))$$

zT and zS are logits from the teacher and student models respectively; T is the temperature parameter and α is the balancing coefficient.

Tiny-DSOD achieves 72.1% mAP on PASCAL-VOC with only 0.95M parameters, compared to more than 20M parameters in conventional detectors, making it suitable for edge devices.

8. COMPARISON OF DATASETS AND QUANTITATIVE PERFORMANCE

Table-4: Dataset-Level Comparison from Prior Studies.

Study	Dataset	Params	FLOPs	mAP (%)	Environment
Faster R-CNN (baseline)	COCO	~41M	180B	36.2	GPU
TranSDet	TT100K-Lite	~38M	165B	+22.7% ↑	Limited data
Tiny-DSOD	PASCAL VOC	0.95M	1.06B	72.1	Edge
LSTD	Custom few-shot	~15M	–	+8–12%	Low-shot

9. TRANSFER-LEARNING–BASED STRATEGIES FOR LOW-RESOURCE SCENARIOS

Recent literature has been working on using many different ways to use combination of efficient network architecture, data efficient techniques, and transfer learning to allow for object detection in constrained environments such as UAVs or edge devices and in very limited training data (few-data) applications. Below, I discuss several methods of doing this by theme; describe how they are used; describe results from prior studies; and explain why they will be useful to low-resource object detection applications.

9.1 Lightweight and Efficient Architectures

The most commonly applied method to make object detection systems suitable for low resource applications, is by using lightweight and efficient backbones and detection architectures that consume less computational resources. The authors of a 2024 review titled "Deep Learning–Based Lightweight Object Detection Models for Edge Devices" discuss how backbone architectures such as MobileNet, ShuffleNet (or other depth separable/efficient convolutional networks) and lightweight detection frameworks such as EfficientDet are being successfully employed for object detection at reasonable accuracy levels, while maintaining low model sizes, low inference times, and low memory footprints [24].

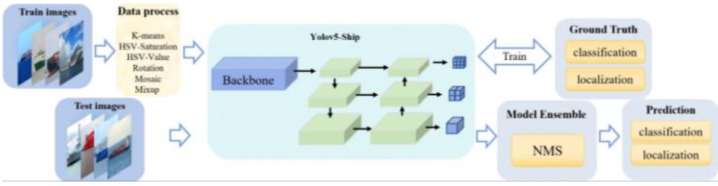

Figure-3: Lightweight / mobile-optimized backbone architectures

In general, these lightweight architectures utilize fewer parameters than their "heavier" counterparts, smaller feature maps, and use depth wise-separable convolutions to decrease both floating point operations (FLOPS) and memory consumption which make them better suited to devices with limited processing power and/or memory (e.g., mobile devices, embedded systems, and edge hardware).

Experimental results have shown that lightweight detectors are able to achieve an acceptable level of performance in terms of detecting objects but not necessarily at the same level as the best performing ("heavy") detectors; however, they can provide a reasonable tradeoff among speed, resources, and object detection performance [25].

9.2 Knowledge Distillation to Compress Heavy Models

A possible approach is to utilize knowledge distillation — a process of passing information from a larger, well-trained "teacher" model to a smaller "student" model — as an approach to reduce size of object detection models; this permits compact models to capture similar performance characteristics of their heavier counterparts, but still be capable of being used at lower levels of resources [26].

An example of this can be seen in a recent study employing knowledge distillation for few-shot object detection — in the case of the proposed method, it utilized a "bag-of-visual-words" (BoVW) representation learned from a small number of images, and then aligned the student detector's feature space to the BoVW embedding representations to provide guidance to the learning process, thereby minimizing over fitting when training using a very small number of examples.

Similarly, a 2024 research effort was able to demonstrate the feasibility of using a combination of knowledge distillation, and a design technique called feature-adaptive backbone design to improve the UAV (drone) object detection — demonstrating the ability to create a balance between object detection capability and resource utilization that was applicable to airborne embedded systems.

These studies show that knowledge distillation will remain one of the most effective ways to "shrink" the amount of computational resources needed by heavy object detectors, while maintaining as much of their learned representation capabilities as possible — enabling object detection to become more feasible for use on devices or within systems that have very limited amounts of computing power, memory, or electrical energy available [27].

9.3 Layer-Freezing and Selective Fine-Tuning Approaches

Transfer learning is typically carried out by beginning with a pre-trained "backbone" and then tailoring it to another application using fine-tuning. However, under tight computational resources, limited data, or both, completely training all of its layers again could be prohibitive for achieving acceptable performance or might lead to over fitting. Many researchers have thus developed various methods of selectively freezing layers (especially lower layers) to preserve the general feature extraction capabilities learned in the source domain, while only fine-tuning upper layers (that are most relevant to the detection task or the new classes). Selective fine-tuning also offers a reduced computational overhead as well as smaller data requirements compared to full-layer retraining. Lightweight backbones or distillation combined with selective freezing of layers helps improve deployment efficiency for object detectors within resource-constrained environments. The majority of studies on few-shot detection have adopted similar fine-tuning methodologies when transferring from base classes to new classes [28].

Regardless of the specifics, selective fine-tuning is widely used due to the fact that it uses minimal resources to adapt the model, but still leverages the broad knowledge of visual patterns and concepts that were learned in the source domain.

9.4 Cross-Domain Adaptation and Reusable Feature Representations

A number of studies have looked at developing ways to adapt feature representation across various deployment domains that are different than the training domain (e.g. different lighting, backgrounds, objects, aerial vs. ground views).

To give an example: The primary problem faced by many few-shot detectors is the domain shift issue when there is a large difference in the base class versus the novel class; as stated above, projects such as DCNet (Dense Relation Distillation with Context-aware Aggregation) creates a density match between the support and query features; which will help improve the detector's ability to generalize when there is a large domain shift.

Causal inspired knowledge distillation has also been proposed to be applied to the few-shot detection task to handle issues of over-fitting and poor generalization to new classes when the base and new classes are very different; using causal inspired knowledge distillation allows the model to learn more stable, transferable feature representations [29].

Using these cross-domain adaptations along with the use of a single, reused pre trained backbone can provide the flexibility and robustness needed for deploying models in diverse real world scenarios even when there is limited amounts of data available or the domain changes greatly.

10. REAL-WORLD APPLICATIONS

Transfer learning for object detection has provided the ability to deploy object detection in a variety of real world applications with limited computing power, processing capability and access to training data. Several studies have shown that utilizing pre-trained backbones, along with training methods that conserve compute resources, enables successful deployment of object detection in various practical fields such as agriculture, public safety, medical imaging, remote sensing and IoT-based smart devices. The next sections detail specific application areas, their limitations and what can be learned from the field experience reported by prior researchers.

10.1 Agriculture and Precision Farming

Object Detection in agriculture is a key task that has numerous uses including crop monitoring, fruit counting, identifying diseases of plants, and detecting pests by analyzing images taken from Unmanned Aerial Vehicles (UAV) or collected from sensors placed on the ground. Studies have demonstrated that when transfer learning is used on an agricultural dataset that a pre-trained model can greatly decrease the amount of data required to be annotated manually, which is a limitation of agriculture since it requires a high cost to annotate and there is a high degree of specialization in annotating agricultural data. Studies have shown that lightweight versions of object detection algorithms such as YOLOv4-Tiny and EfficientDet-D0 are capable of running on small platforms like drones and mobile devices and perform fruit detection and crop classification at speeds of 30 frames per second, while demonstrating significant increases in performance over training from scratch. Studies conducted in actual field environments have demonstrated that lightweight models are critical for battery powered equipment operating in the fields, as these models require less memory and energy than traditional models [30]. These studies have also demonstrated that lightweight models enable adaptation to varying environmental conditions such as changing light levels, occlusion from leaves, and irregularly shaped objects.

10.2 Intelligent Surveillance and Public Safety

The ability to detect objects within video feeds is a key function of many types of surveillance systems. Examples include cameras placed at public locations such as traffic intersections, parks and near security sensitive buildings. In addition, there are a variety of research studies which demonstrate that using lightweight object detection systems that have been pre-trained and then deployed onto edge computing hardware (i.e., Jetson Nano, Raspberry Pi, etc.) can provide a means to perform real time vehicle detection, pedestrian detection and abnormal activity detection without the need to send the video feed to be processed centrally through a cloud environment [31]. The move toward edge based AI models is primarily due to bandwidth limitations when communicating video to remote servers; the desire to protect user's private information and the need for rapid action in emergency situations. It has also been shown that by selectively fine tuning an edge based model and compressing its weights, high accuracy performance may be achieved with edge based models. However, it has also been noted that there will be a need to balance model complexity with responsiveness when implementing practical solutions. Transfer learning will continue to play a major role in optimizing the use of surveillance technology by providing a method to achieve both high accuracy and fast response times.

10.3 Medical Imaging and Clinical Diagnostics

Transfer learning is also used with medical image modalities (i.e., X-rays, CT scans, MRIs, and ultrasounds) to enhance computer-aided-diagnosis. Because there is a lack of annotated medical images and due to the high cost associated with annotating them by experts; many researchers use pre-trained convolutional neural networks to detect tumors, segment organs, or identify anomalies on these images; they then fine-tune those networks to specific detection tasks. The literature shows that transfer learning is very useful for reducing dependence upon large amounts of data and it can provide higher diagnostic accuracy than traditional feature-based methods. Furthermore, the ability to create lighter versions of transfer learned models provides additional opportunities for deploying them in portable medical devices, mobile health apps, and field diagnostic stations which have limited computing capabilities and/or connectivity [32]. Practical experience with domain adaptation was also discussed in clinical studies because when trained on one population and equipment, pre-trained models need to be fine-tuned to match another population and possibly another equipment.

10.4 Remote Sensing and Aerial Object Detection

Object detection is a key element for remote sensing applications including land use/cover mapping and damage assessment based on images obtained from either satellite or Unmanned Aerial Vehicle (UAV) platforms; thus transfer learning-based detectors are better at handling domain variability due to differences in altitude, scale and viewpoint than learning from data from scratch. Lightweight detector studies for UAV-based object tracking in disaster zones demonstrate that using knowledge distillation along with lightweight detectors achieves real time inference on airborne hardware with very limited battery power. Experiences during deployment of these systems has demonstrated that local onboard processing (as opposed to relying upon cloud-based systems) will significantly reduce latency which can be critical in mission time sensitive areas such as search and rescue.

10.5 IoT and Edge-AI Devices

Object detection has been shown to be useful for both smart-home monitoring (IoT) and industrial automation and/or environmental sensing. The limitations of edge-AI systems in terms of CPU/GPU capacity, memory and power consumption make it difficult to implement object detection as a reliable and efficient process; thus the need for using transfer learning along with model compression techniques such as pruning and quantization, or efficient architecture designs to allow for fast object detection; the case studies of implementing object detection in micro-edge devices have demonstrated that even when there is little-to-no connectivity, and the ability to operate in an offline mode, a compressed transfer learned model can provide reliable results; these are examples of how the use of object detection in edge-AI devices are successful when trade-off between accuracy and available resources is well-balanced.

10.6 Deployment Insights and Practical Lessons

Practical observations are common across various use cases of AI. The first observation is that feature representation transfer from a broad general dataset (COCO, ImageNet etc.) will improve both the accuracy and decrease training times when there is a lack of specific data for your domain. The second observation is that selecting a model size based on available device capabilities is crucial — the larger the model, the greater its latency and memory requirements, and the less it may be able to functionally operate at acceptable levels on smaller edge devices. The third is that the best possible trade off for operational efficiency in real-time applications is found through the use of a lightweight backbone model paired with either selective fine tuning or knowledge distillation. Lastly, deployment research shows that environmental variability (weather, lighting, noise, cluttered backgrounds) have a strong impact on overall performance, which highlights the importance of domain adaptation and developing new techniques to generalize better.

11. CONCLUSION

Object Detection is playing a critical role across various real-world domains, however, the deployment of High Performance Models in Low Resource Environments remains a serious problem due to limits of Hardware Capability, Energy Availability, Dataset Size, and Connectivity. This Review demonstrates how Transfer Learning has evolved to become a very effective method to overcome many of the problems associated with the deployment of Deep-Learning models, by allowing them to utilize Pretrained Feature Representations, simplify model training and achieve similar or better Accuracy than larger models using significantly less Computationally and Data Resources. The use of strategies such as Lightweight Architectures, Knowledge Distillation, Selective Fine-Tuning, Domain Adaptation, and Data-Efficient Learning have consistently shown to close the gap between Advanced Research in Deep Learning and Practical Deployment Requirements. Studies across Agriculture, Surveillance, Healthcare, Remote Sensing, and IoT Devices have demonstrated that the use of Transfer Learning enables Object Detection to be implemented in Constrained Environments in a more Accessible, Deployable, and Robust manner. However, despite the progress made in this area, there are still Challenges to be addressed in the areas of Balancing Efficiency and Accuracy, Managing Variability in Domains, and Improving Generalization when working with Extremely Limited Amounts of Data. Therefore, it is anticipated that further Research into Adaptive and Scalable Frameworks for Transfer Learning will continue to be important for developing the Next Generation of Resilient, Intelligent, and Resource Efficient Systems for Object Detection, which can be widely adopted in Real-World Applications.

REFERENCES

[1] X. Xu, H. Zhang, Y. Ma, K. Liu, H. Bao, and X. Qian, “TranSDet: Toward Effective Transfer Learning for Small-Object Detection,” Remote Sensing, vol. 15, no. 14, 2023.

[2] H. Chen, Y. Wang, G. Wang, and Y. Qiao, “LSTD: A Low-Shot Transfer Detector for Object Detection,” in Proceedings of AAAI, 2018.

[3] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, “DSOD: Learning Deeply Supervised Object Detectors from Scratch,” arXiv preprint arXiv:1708.01241, 2017.

[4] B. Wu, A. Wan, X. Yue, K. Keutzer, “Tiny-DSOD: Lightweight Object Detection for Resource-Restricted Usage,” arXiv preprint arXiv:1807.11013, 2018.

[5] Q. Huang, H. Zhang, M. Song, and others, “A Survey of Deep Learning for Low-Shot Object Detection,” arXiv preprint arXiv:2112.02814, 2021.

[6] Q. Feng, X. Xu, and Z. Wang, “Deep learning–based small object detection: A survey,” Mathematical Biosciences and Engineering, vol. 20, no. 4, pp. 6551-6590, 2023.

[7] J. Wu, “Data-Efficient Object Detection Combining YOLO with Few-Shot Learning Techniques,” Journal of Computer Signal and System Research, vol. 2, no. 2, pp. 70–85, 2025.

[8] J. Lee, B. Varghese, R. Woods, and H. Vandierendonck, "TOD: Transprecise Object Detection to Maximise Real-Time Accuracy on the Edge," arXiv preprint arXiv:2105.08668, 2021.

[9] P. Mittal, "A comprehensive survey of deep learning–based lightweight object detection models for edge devices," Artificial Intelligence Review, 2024.

[10] "Comparison of Transfer Learning Techniques for Object Detection," IJISAE, 2023.

[11] "Object Detection and Deep Learning," IJIR-CST, 2025.

[12] "Object Detection Deep Learning Research Papers," Kaggle-compiled list (2019).

[13] "A comprehensive survey of deep learning–based lightweight object detection models for edge devices," Consensus / SemanticScholar (2024).

[14] M.-C. van Leeuwen, et al., "Toward Versatile Small Object Detection with Temporal-YOLOv8," 2024.

[15] "Small Object Detection: A Comprehensive Survey on Challenges," arXiv preprint arXiv:2503.20516, 2025.

[16] S. Han, H. Mao, and W. J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding," in ICLR, 2016. (relates to model compression for low-resource deployment)

[17] "SqueezeNet: AlexNet-level accuracy with 50× fewer parameters," 2016. (a pioneering lightweight backbone)

[18] P. Mittal, "A comprehensive survey of deep learning–based lightweight object detection models for edge devices," Artificial Intelligence Review, 2024.

[19] C. Su, "Nav-YOLO: A Lightweight and Efficient Object Detection Network," ISPRS International Journal of Geo-Information, vol. 14, no. 9, 2025.

[20] Y. Zhuang et al., "Few-shot learning for novel object detection in autonomous driving," Journal of Transportation Computer Vision and Applications, 2025.

[21] M. Chao et al., "A lightweight small object detection model for UAV images," Scientific Reports, 2025.

[22] R. Chen et al., "An optimized lightweight real-time detection network model for IoT embedded devices (FRYOLO)," Scientific Reports, 2025.

[23] B. Song et al., "YOLO-ELWNet: A lightweight object detection network for onboard applications," Neurocomputing, 2025.

[24] W. Wu et al., "EUAVDet: An efficient and lightweight object detector for edge-based UAV devices," Drones, vol. 8, no. 6, 2024.

[25] C. Liu, J. Wen, J. Huang, W. Lin, B. Wu, N. Xie, and T. Zou, "Lightweight Underwater Object Detection Algorithm for Embedded Deployment Using Higher-Order Information and Image Enhancement," Journal of Marine Science and Engineering, vol. 12, no. 3, 2024.

[26] Y. Zhang, "A lightweight and effective few-shot object detector for ore images," Computers & Geosciences, 2023.

[27] A. Bouguettaya, "A Survey on Lightweight CNN-Based Object Detection for Embedded Targets," Turkish Journal of Electrical Engineering & Computer Science, 2023.

[28] Z. Yang, C. Zhang, R. Li, Y. Xu, G. Lin, "Efficient Few-Shot Object Detection via Knowledge Inheritance," arXiv preprint, 2022.

[29] G. Han, S. Huang, J. Ma, Y. He, S.-F. Chang, "Meta Faster R-CNN: Towards Accurate Few-Shot Object Detection with Attentive Feature Alignment," arXiv preprint, 2021.

[30] Y. Li, J. Li, W. Lin, and J. Li, "Tiny-DSOD: Lightweight Object Detection for Resource-Restricted Usages," arXiv preprint, 2018.

[31] Z. Fan, Y. Ma, Z. Li, J. Sun, "Generalized Few-Shot Object Detection without Forgetting," in CVPR, 2021.

[32] "Benchmarking Deep Learning Models for Object Detection on Edge Devices," arXiv preprint, 2024.