DA-Net: A Hybrid Deep Learning Architecture for Robust Crowd Counting in Dense Scenes

Abstract

DA-Net is a crowd counting model that addresses challenges such as scale variations and occlusions by leveraging a combination of advanced techniques. It uses an **EfficientNet-B5** as backbone for feature extraction, a **Feature Pyramid Network (FPN)** for multi-scale fusion of features, and a **spatial attention mechanism** to focus on parts which contribute the most in results and finally a **dilated convolutional layer** which produces rich and semantic density maps. During training, **custom composite loss function** was used to do both pixel-level machine learning and to have consistent total counts. After significant effort in development, DA-Net achieved a **Mean Absolute Error (MAE)** score of **20.99** on a challenging dataset and outperformed existing models.

1. Introduction

Crowd counting has been a prominent problem in computer vision for quite a while; in the past couple of years, we've seen many different approaches to this problem and we have used a density map based approach in our solution. It is a regression-based approach to constructing a 2D density map that allows the counting of individuals via a single integral over any region. The major challenges such as variation in scale, distortion due to perspective, and excessive scene complexity are faced in crowd counting. By overcoming these challenges we were able to achieve a robust crowd counting model which outperformed existing models.

2. Thought Process and Related Work

Our design process started by reviewing seminal works in crowd counting, we used all these works as a base knowledge to this problem. The few papers mentioned below were useful starting points:

- **MCNN (2016):**tackled scale variations with three parallel CNN columns of different sizes of filters, the downside was the later and less efficient information sharing between columns.
- **CSRNet (2018):** made big improvements on previous methods by using a VGG-16 backbone and dilated convolution back end. Dilated convolution produced very large receptive fields to provide contextual information while maintaining spatial resolution.
- **CAN (2019):** was a contemporary extension of CSRNet, introduced an adaptive scale ability through a parallel dilated convolution at different dilated rates and formed an attention mechanism to weigh the various features.

Our approach synthesized these ideas: a modern pre-trained backbone (like CSRNet), principled multi-scale feature fusion (leading to FPN), and an adaptive focus mechanism (inspired by CAN) and we were able to design a good model.

3. Alternative Architectures Explored

After exploring all existing methods and a few papers as mentioned above we implemented a few model architectures to test, experiment and learn from them. These are the alternatives which we explored and evaluated before arriving at the **DA-NET** architecture.

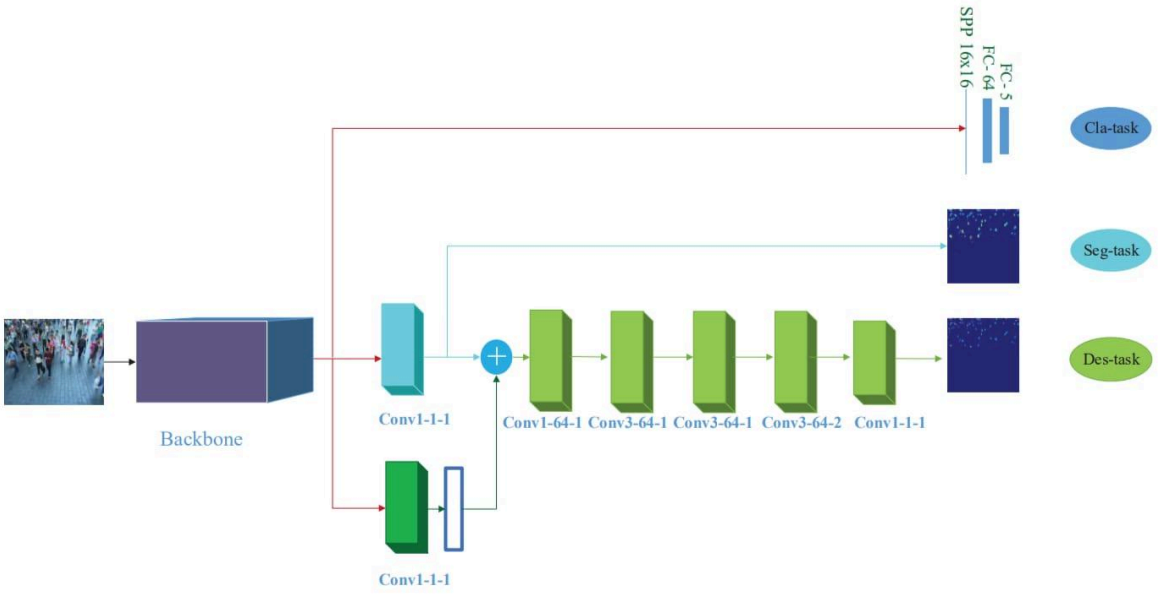
3.1 Baseline Evaluation (Based on CSRNet):

We used **CSRNet's reported performance (MAE 68.2 on ShanghaiTech Part A)** as a key baseline. This prominent and relatively simpler architecture provided a strong indicator for evaluating our model's progress and we implemented this architecture from scratch for understanding its novelty and tested its performance on our dataset.

3.2 VGG-Based Segmentation-Style Decoder (SegCrowdNet):

After examining CSRnet architecture, we also tried SegCrowdNet and reimplemented it from scratch to test its performance and explore novelty in its architecture. Its architecture used a segmentation-inspired decoder and it looked like this:

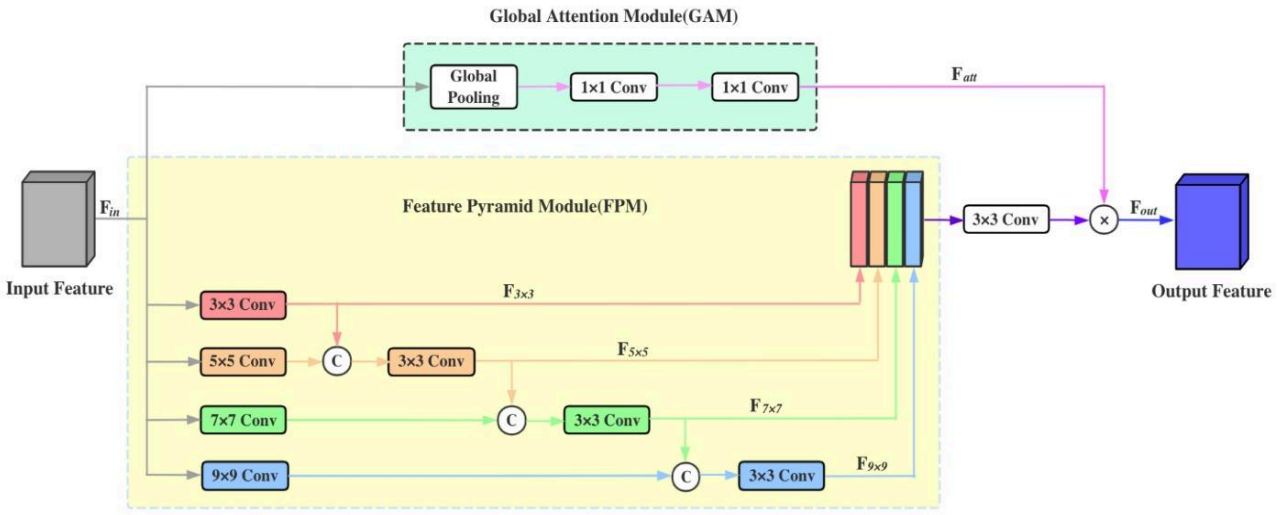
- **Backbone:** Employed a pre-trained **VGG-16** (removing the last max pooling layer) to retain 16x downsampled feature maps.
- **Decoder:** Consisted of convolutional blocks interleaved with three
- **Upsample(scale_factor=2)** layers to reconstruct spatial detail.
- **Ground Truth Generation:** Used a simple resize-and-scale method.
- **Loss Function:** Trained exclusively with **Mean Squared Error (MSE)** loss.
- **Training Strategy:** Utilized **Adam optimizer**, a **StepLR scheduler**, and standard data augmentation.



3.3 VGG-Based Multi-scale Decoder with Message Passing and Global Attention:

After SegNET we explored this architecture, it enhanced the multi-column concept (like MCNN) by improving inter-scale information flow and adding attention and this is how its architecture is:

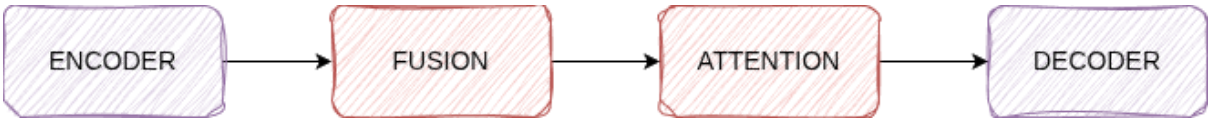
- **Backbone:** Uses the first 10 layers of a pre-trained **VGG-16** as an encoder, downsampling features by 8x.
- **Core Modules (Three Custom Modules):** These modules are the primary innovation, processing and refining multi-scale features.
    - **Module Structure:** Each module features parallel branches utilizing **dilated convolutions** for large receptive fields.
    - **Message Passing Mechanism:** This allows explicit interaction between features from different scales. Lower branch outputs concatenate with upper branch outputs, fused by a 3x3 convolution. This enables cross-scale contextual understanding, unlike isolated multi-column networks.
    - **Global Attention Module (GAM):** A **Global Average Pooling (GAP)** layer summarizes global context. This context is processed by two 1x1 convolutions (with a reduction ratio, typically 16) and a sigmoid activation to learn channel-wise attention weights, highlighting important feature channels.
    - **Module Output:** Parallel branch outputs (refined by Message Passing) are fused and multiplied by the GAM's channel attention weights.
- **Decoder:** A lightweight decoder uses upsampling layers to reconstruct the final density map from the processed multi-scale features.



4. Proposed Methodology: The DA-Net Architecture

After all that implementation, research and learnings we arrived at our final model DA-Net which is designed as a modular system where each component is chosen to solve a specific sub-problem of crowd counting and give overall better performance by overcoming major issues of crowd counting.

4.1. Overall Architecture:

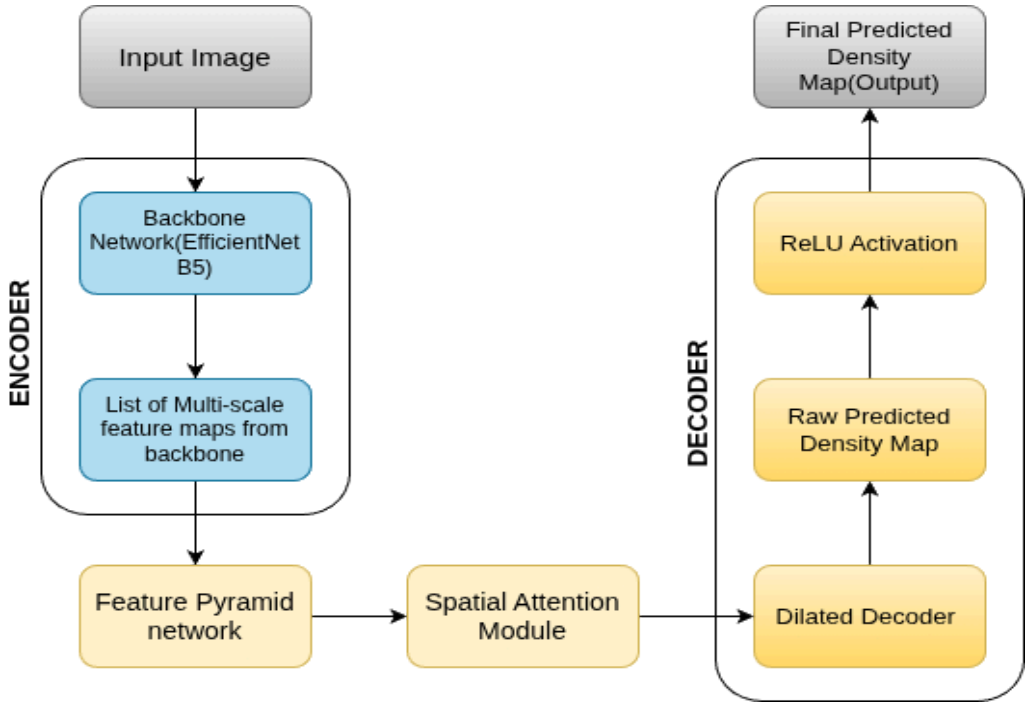


4.2. Encoder: Transfer Learning with EfficientNet-B5:
We have used EfficientNet-B5 as our backbone in our model architecture as it allows us to take multiple feature maps as output from different stages which helps us in overcoming the issue of scale variation as early layers contain fine features and helps in detecting tiny people and later layers helps in getting complex features and have rich semantic information of large crowd .

4.3. Multi-Scale Fusion: Feature Pyramid Network (FPN):
As we get Multiple feature maps, we employ a standard FPN. The FPN takes all feature maps and merges them through a series of top-down and lateral connections and produces a list of features with high semantic information and with fine details (we use the feature map with highest resolution) .

4.4. Adaptive Focus: Spatial Attention Module:
Inspired by CAN, we have used a lightweight Spatial Attention Module. This module operates on the fused feature map from the FPN and computes a "relevance mask.". By multiplying the feature map by this mask, the network learns to dynamically allocate its capacity to the most important parts of the scene.

4.5. Decoder: Dilated Convolution Backend:
After the attention layer, Our decoder uses a series of 3x3 convolutions with a fixed dilation rate of 2. This allows the decoder to aggregate contextual information from a large receptive field without downsampling the feature map and produces a high resolution density map that is both spatially accurate and contextually aware.



5. Experimental Setup and Training Strategy

5.1. Dataset:

We used the given dataset for training and evaluation, Density maps were generated using a python script by combining both ground truth and images and saved as .npy files

5.2. Data Augmentation:

After loading the dataset we have used several data augmentation techniques for better result and generalization of our DA net model.

- **Random Crop (512x512)**
- **ShiftScaleRotate**
- **Horizontal Flip**

5.3. Implementation Details & Training:

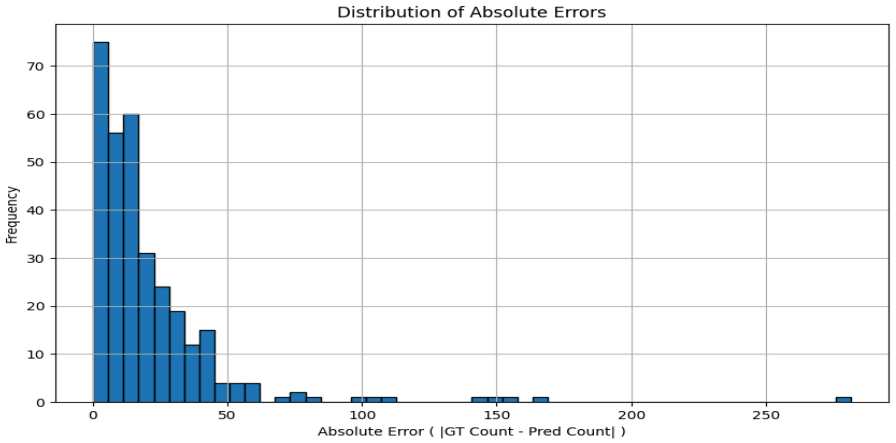
- **Loss Function**: mse_count_mae_loss_pytorch with count_loss_weight of 0.05
    - The final loss function is a weighted sum of two components:
    - The final loss is: **Total Loss=mse+weight∗count_mae.**
- **Training Loop:** Single-stage, 100 epochs.
- **Optimizer: AdamW**, with a learning rate of 5e-5 and weight_decay of 0.1.
- **Model Input/Output:** Input images were 256x256 (most suitable for B5); ground truth density maps were 128x128. Model output was bilinearly upsampled to 128x128 for loss calculation.

6. Results and Analysis

6.1. Qualitative Analysis:

The provided visualizations offer deep insights into the model's behavior.

- **Error Distribution:**



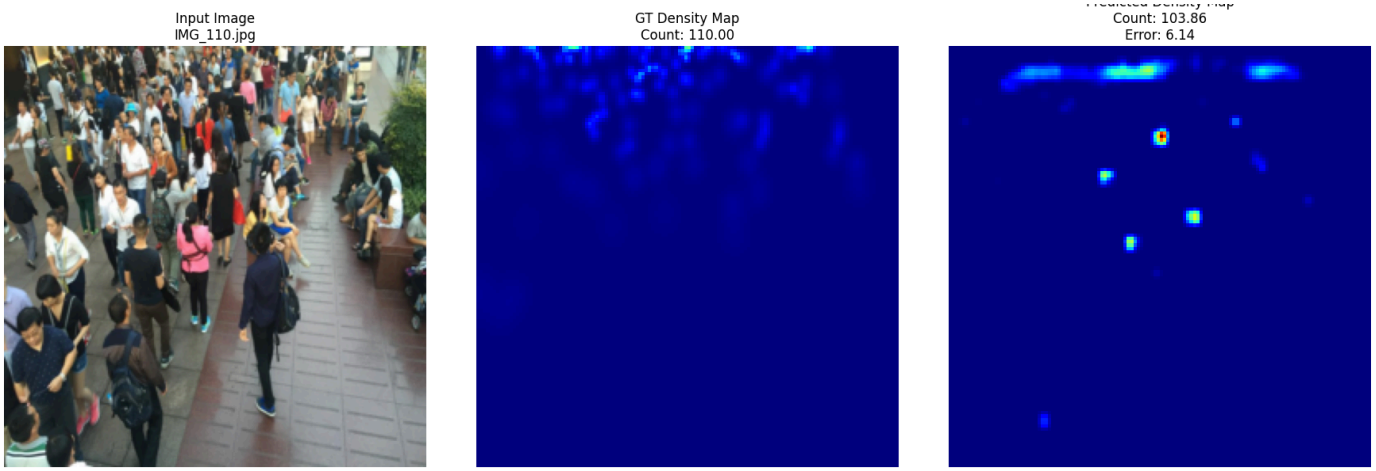
6.2. Quantitative Results:

After full training, DA-Net was evaluated on the unseen test set using Test-Time Augmentation (TTA). The model achieved the following results:

- **Mean Absolute Error (MAE): 20.99**
- **Root Mean Squared Error (RMSE): 34.35**

MODEL	MAE	OUR DA-Net MAE
CSRNet	62.2	**20.99**
SegCrowdNet	54.6	-
PSNet	26.50	-

- **Visual Example:**



7. Novelties

- **Attention mechanism:** provides a focus on important components of the crowd.
- **Multiple feature maps from different levels:** made model scale invariant.
- **Dilated convolution:** provides richer and semantic density maps as it does not down sample input.

8. Overcoming Blockers

The development faced significant training instabilities due to a complex, multi-component loss function (including Optimal Transport and SSIM).

- **Exploding Loss:** The un-normalized Optimal Transport (OT) loss component dominated, causing massive, unstable gradient updates and destroying model weights early on.
- **Stagnant Training:** Attempts to balance the complex loss with small weights resulted in the model failing to learn.

The turning point was simplifying the objective to a stable, two-component MSE and Count MAE loss, which immediately resolved instabilities.

9. Conclusion

This project successfully developed **DA-Net**, a robust and accurate crowd counting model, by employing a **hybrid, synthesis-based approach**. It intelligently combines state-of-the-art components: an **EfficientNet-B5** backbone and **FPN** for feature extraction and multi-scale fusion, **Spatial Attention** for adaptive focus, and a **Dilated Decoder** for refined output. Through iterative development, including exploring alternative architectures (e.g., **CSRNet, VGG-PSMNet**), rigorous debugging, and refining data augmentation and loss functions, DA-Net achieved an **MAE of 20.99** on a challenging dataset, significantly outperforming baselines.