

Comparing between three models

1 Architecture Idea

Model	Core Idea
ResNet	Uses residual (skip) connections to solve vanishing gradient problems
InceptionV3	Uses parallel convolution paths with different filter sizes
EfficientNetB0	Scales depth, width, and resolution together (compound scaling)

2 Network Depth & Complexity

Model	Depth / Complexity
ResNet (e.g. ResNet50)	Deep network (50+ layers)
InceptionV3	Deep but structured with Inception modules
EfficientNetB0	Relatively shallow but very efficient

3 Parameters & Model Size

Model	Approx. Parameters	Model Size
ResNet50	~25 million	Large
InceptionV3	~23 million	Large
EfficientNetB0	~5 million	Small (lightweight)

✔ EfficientNetB0 is the most parameter-efficient

4 Training Speed & Resource Usage

Model	Training Speed	Hardware Demand
ResNet	Medium	High
InceptionV3	Slow	Very High
EfficientNetB0	Fast	Low

✓ If you trained on **limited GPU/CPU**, EfficientNetB0 is the best choice.

5 Accuracy vs Efficiency Trade-off


Model	Accuracy	Efficiency
ResNet	High	Medium
InceptionV3	Very High	Low
EfficientNetB0	High	Very High

💡 EfficientNetB0 often gives **near Inception accuracy with much fewer resources**.

6 Overfitting Risk

Model	Overfitting Risk
ResNet	Medium
InceptionV3	High (complex architecture)
EfficientNetB0	Low (good regularization & scaling)

7 Practical Recommendation (Important for Your Project)

Scenario	Best Model
Small or medium dataset	EfficientNetB0
Very large dataset	InceptionV3
Classic deep learning baseline	ResNet
Limited hardware	EfficientNetB0
Best balance overall	EfficientNetB0 

8 One-Line Summary (Great for Reports)

- **ResNet**: Deep and stable, good baseline for CNN tasks
- **InceptionV3**: Powerful but computationally expensive
- **EfficientNetB0**: Best accuracy–efficiency trade-off, ideal for most real-world tasks