UNIVERSITY OF TRENTO

Department of Information Engineering and Computer Science
Master's Degree in Artificial Intelligence Systems

Knowledge distillation for fast and accurate **monocular depth estimation** on **mobile devices**

Advanced Computer Vision | 2021-22

Original paper by: Wang et al. (2021)

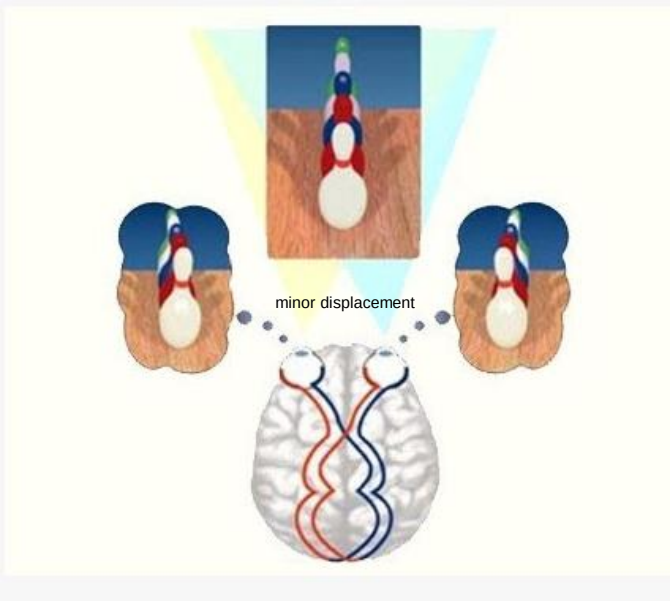
Presentation & improvement:
Wamiq Raza (224824) & Francesco Trono (221723)

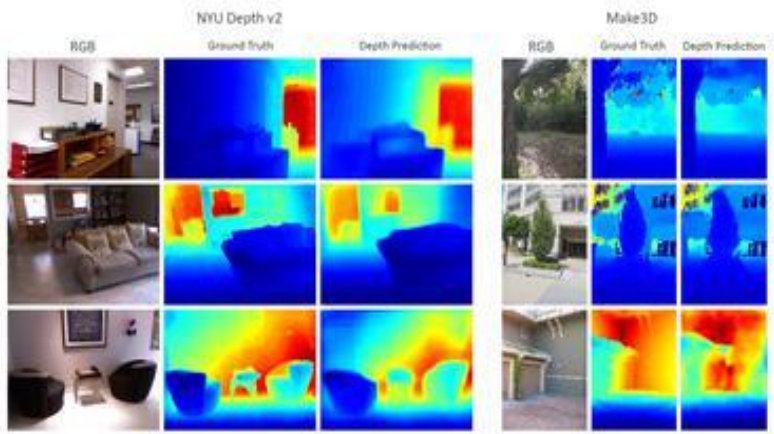
Table of contents

1. Background

Depth estimation & TinyML

What is Depth Estimation



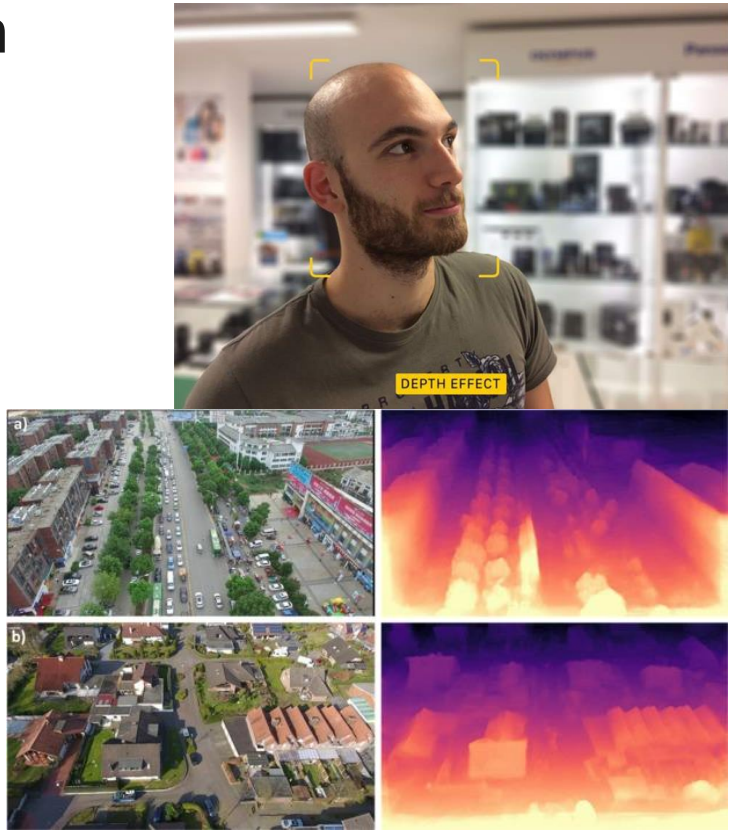


Knowledge Distillation

- What is Knowledge Distillation
  - It’s the procedure or the process for reducing the model complexity and computation overhead while maintaining the performance same as originally. One of the feasible way is to quantize the model, prune the redundant parameters and many more.

Real life examples of application

- Bokeh effect in photos
- Better rendering of 3D scenes
- Self-driving cars
- Grasping in robotics
- Robot-assisted surgery
- Automatic 2D-to-3D conversion in film
- Shadow mapping in 3D computer graphics
- And many more



2. The paper's contribution

Wang et al. (2021)

Idea, architecture & results

Current state of the art

MONOCULAR DEPTH ESTIMATION ("MDE")

- CNN with **CRFs** *(Liu et al, 2015)*
- Pixelwise **attention-based** classification *(Li et al, 2018)*
- Depth maps, **surface normals** & semantic labels estimation *(Eigen et al, 2015)*
- *Midas-v2.1_small* for **mobile** *(Ranftl et al, 2019)*
- ***FastDepth***: real-time depth estimation on embedded systems *(Wofk et al, 2019)*

MOBILE DNN ARCHITECTURES

- ***MobileNet***, with depth-wise **separable convolutions** *(Howard et al, 2017)*
- *MobileNet-V2*, with inverted residuals & linear bottleneck *(Sandler et al, 2018)*
- *MobileNet-V3*, with **neural architecture search** *(Howard et al, 2019)*
- *EfficientNet*: **model scaling** w. ConvNets *(Tan et al, 2019)*

MOBILE DNN LIBRARIES

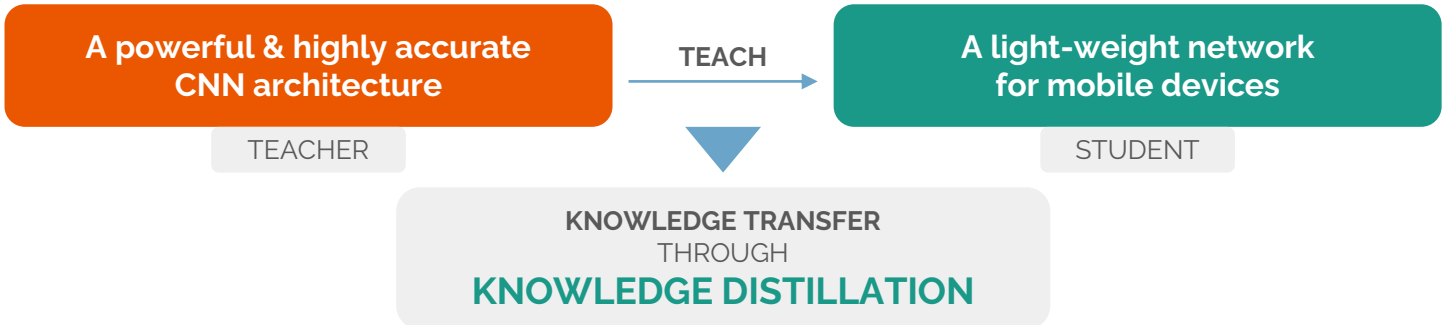
- ***CNNdroid***: GPU-accelerated CNN library *(Oskouei et al, 2016)*
- *SoC-specific SDKs (i.e. from Qualcomm)*
- ***TensorFlow Lite***: optimized kernels & activations

KNOWLEDGE DISTILLATION

- First proposal (***Hinton*** *et al, 2015)*
- **Unsupervised monocular depth estimation** *(Pilzer, Lathuiliere, Sebe, Ricci, 2019)*
- **Structured framework** for knowledge distillation based on conditional GAN learning (Liu et al, 2019)

Main idea of the paper

- Take the **best of the 4 areas**
- Make:



The student network **learns to extract similar feature maps at different scales** as the teacher network does.

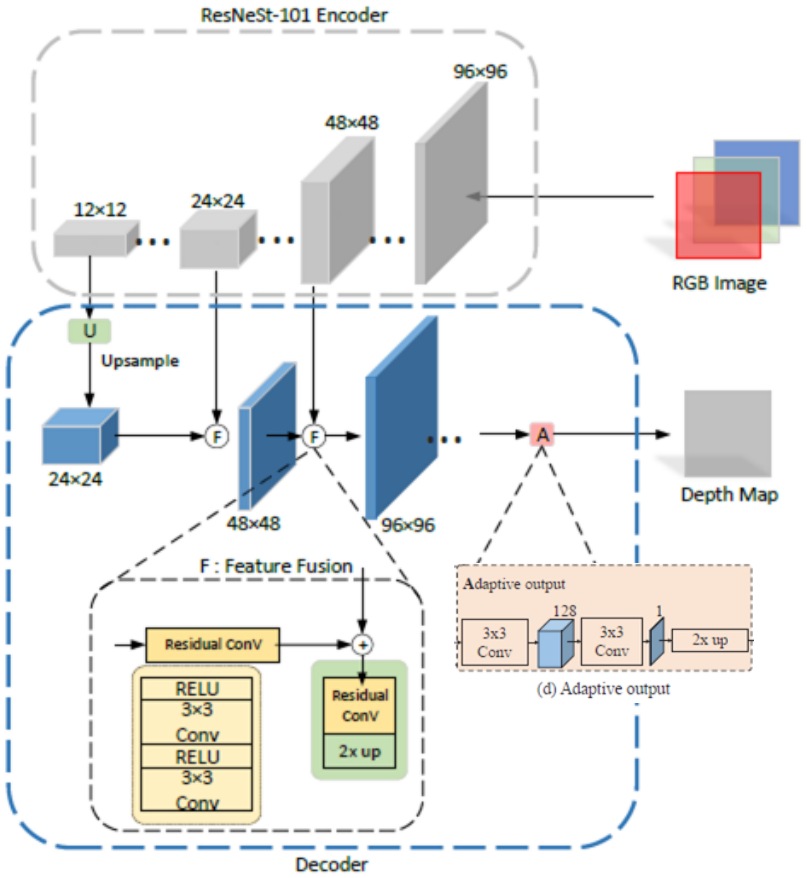
ADVANTAGE:
Higher accuracy of mobile NN with a **lighter architecture**.

1) Teacher network

Feature Fusion architecture

(Xian et al, 2018)

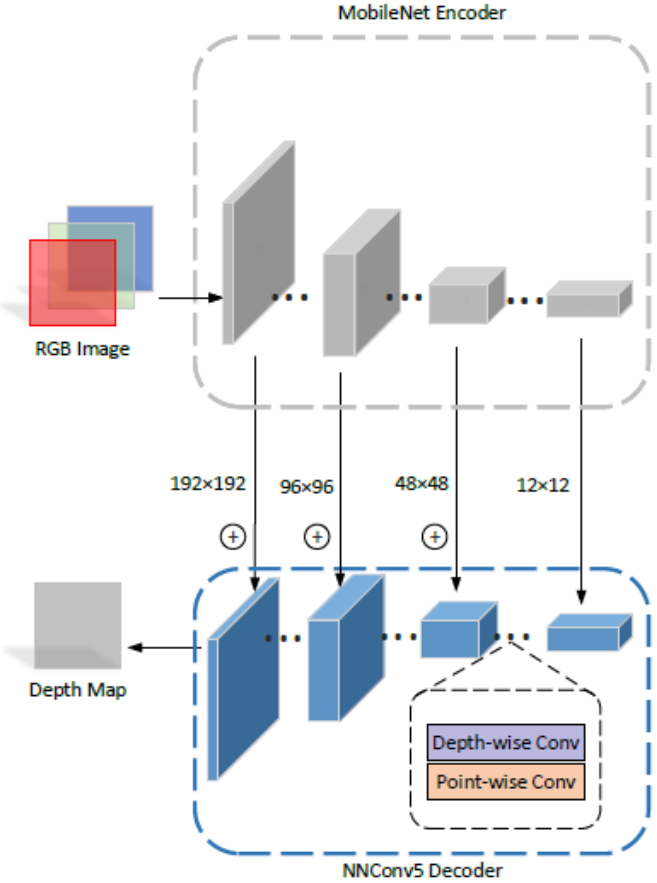
- **ENCODER** → **features extraction**
  - ResNeSt-101
- **DECODER** → **density map**
  - Multi-scale **Feature Fusion** modules:
    - **Transfer feature maps** from specific layers of the encoder using **residual convolutional blocks**
    - Align number of channels through **3x3 conv**
    - **Merge** new maps with **fused feature maps** produced by the previous decoder module via **summation**
    - **Upsample** to next module's input size
  - Adaptive output block (channel adj.):
    - **2 x (3x3 conv)**
    - **bilinear interpolation (KNN)**
- **SKIP CONNECTIONS** → **preserve semantic information**



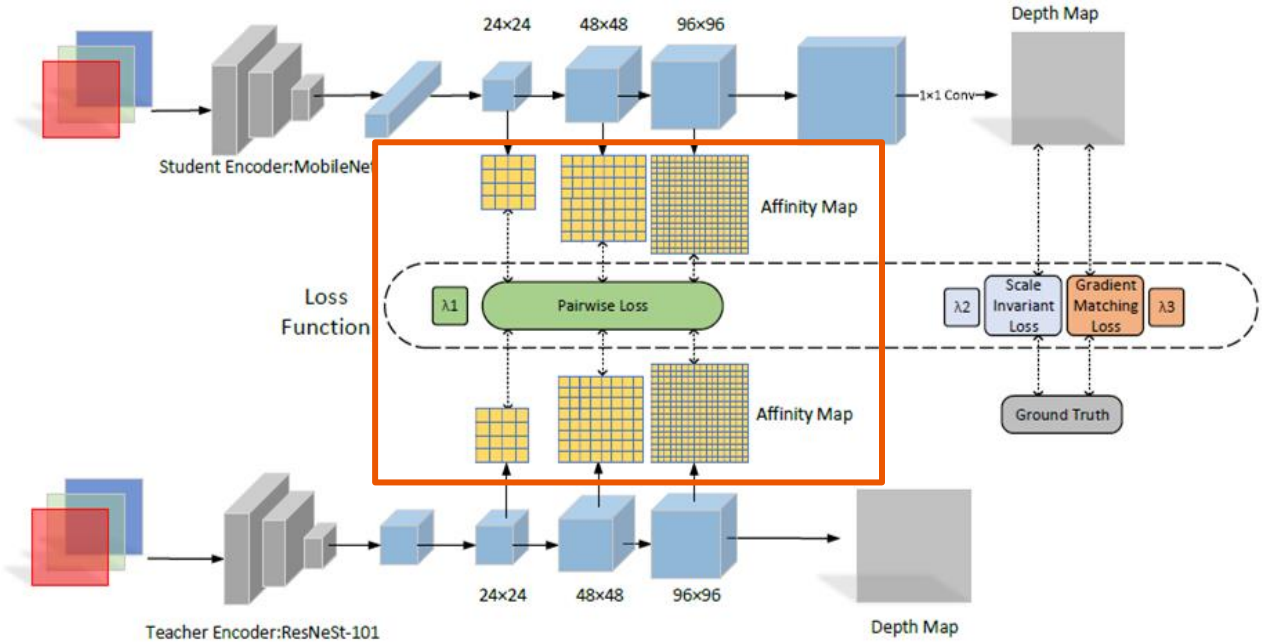
2) Student network

FastDepth architecture
(Wofk et al, 2019)

- **ENCODER** → **features extraction**
  - MobileNet, with:
    - **depthwise decompositions**
    - **pointwise** convolutions (1x1)
- **DECODER** → **density map**
  - NNConv5 (5 conv. layers + 1 pointwise conv.), with:
    - **5x5 masks** (upsample)
    - **depthwise decompositions** (to half the number of channels)
    - **bilinear interpolation (KNN)** (to double the spatial resolution)
- **SKIP CONNECTIONS** → **preserve semantic information**



Knowledge Distillation (KD)



DISTILLATION LOSS

$$\mathcal{L}_d = \sum_{i=1}^{K} \mathcal{L}_{pa}\left(M_{t,i}, M_{s,i}, \theta_s\right)$$

- **Compare k pairs of feature maps** (from teacher & student)
- Teacher network's parameters are **fixed** when training the student

Loss functions

DEPTH MAP

DISTILLATION

1) SCALE-INVARIANT LOSS

(Eigen et al, 2014)

$$\mathcal{L}_s\left(d, d^*, \theta_s\right) = \frac{1}{n}\sum_i g_i^2 - \frac{1}{n^2}\left(\sum_i g_i\right)^2$$

- Delta between **output** and **ground truth** depth map
- Calculated **pixelwise** (*n* pixels): $\mathbf{g_i = \log d_i - \log d^*_i}$

2) GRADIENT MATCHING LOSS

(Ranftl et al, 2019)

$$\mathcal{L}_{gml}\left(d, d^*, \theta_s\right) = \frac{1}{M}\sum_{k=1}^{K}\sum_{i=1}^{M}\left(|\nabla_x^k W_x^k| + |\nabla_y^k W_y^k|\right)$$

- Forces to output depth maps with **sharp edges & smooth internal areas** coinciding with ground truth
- Calculated on ***k* levels of resolution**
- ∇ = gradients of **prediction**
- W = gradients of **ground truth**

3) PAIRWISE DISTILLATION LOSS

(Liu et al, 2019)

$$\mathcal{L}_{pa}(m_s, m_t) = \frac{1}{w \times h}\sum_i\sum_j\left(a_{ij}^s - a_{ij}^t\right)^2$$

- Supervises the **knowledge transfer**
- Compares **pairs of feature maps** from teacher & student
- **2 steps:**
  - Compute pairs' **affinity maps**
  - Compute **MSE** between aff. maps

OVERALL LOSS FUNCTION

$$\mathcal{L}(I, d^*, \theta_s) = \lambda_s \cdot \mathcal{L}_s\left(d, d^*, \theta_s\right) + \lambda_{gml} \cdot \mathcal{L}_{gml}(d, d^*, \theta_s) + \lambda_d \cdot \mathcal{L}_d(M_t, M_s, \theta_s)$$

Weighted sum of the 3 losses

KD: Effectiveness

DATASET
MAI2021 Monocular Depth
7385 RGB outdoor images
+ depth maps (up to 40 m)

ABLATION STUDY

Knowledge distillation impact
Student network training results comparison:

Method	si-RMSE	RMSE	rel
Direct training (no KD)	0.295	4.042	0.276
Knowledge distillation	0.282	3.951	0.263

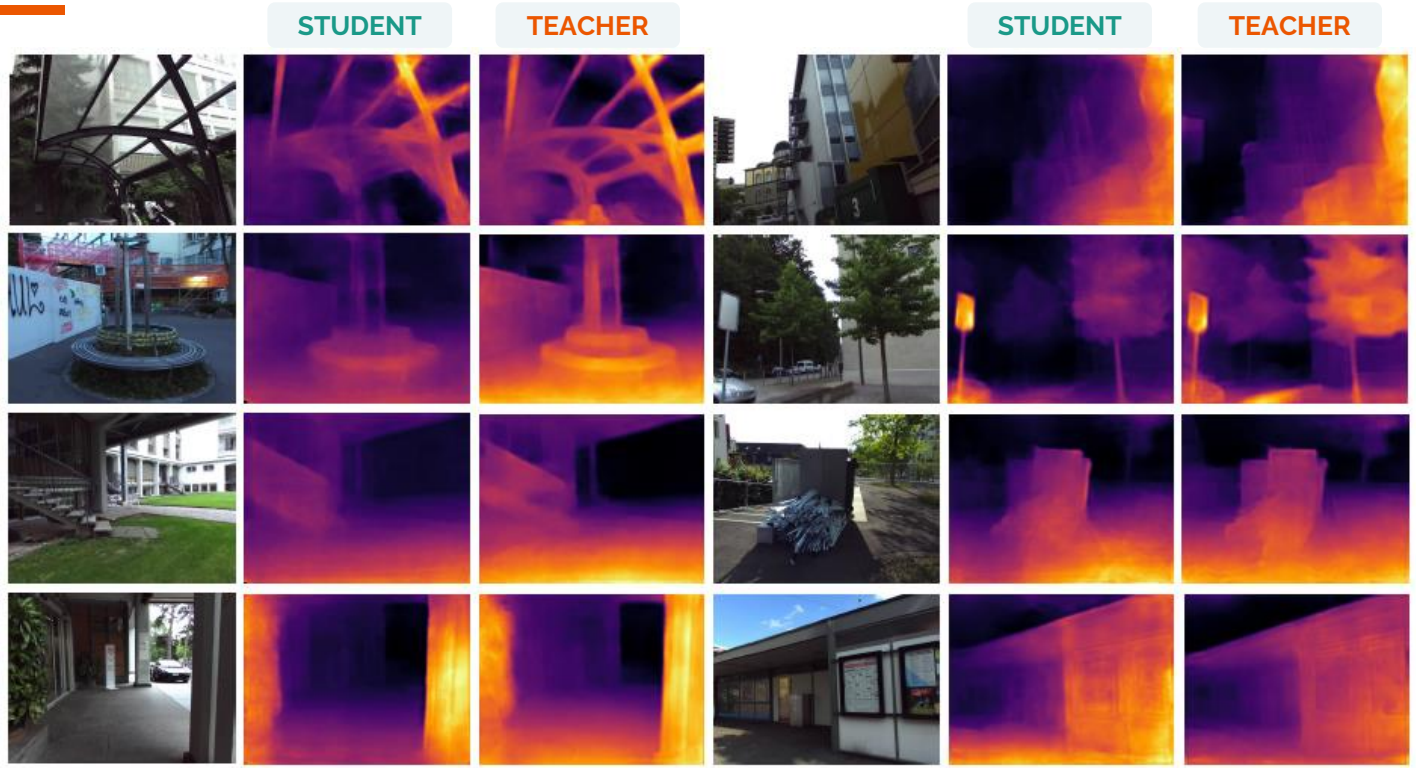
- **RMSE**: *Root MSE*
- **si-RMSE**: *Scatter Index RMSE (normalized by data mean)*
- **rel:** *Relative Deviation (normalized by ground truth)*

INFERENCE ON SMARTPHONE

Execution runtime
High-throughput inference at **20 FPS**

SoC	Device	Avg/ms	STD/ms
Snapdragon 888	CPU	**102.0**	4.02
Snapdragon 888	CPU Delegate	**46.9**	0.79
Kirin 970	CPU	**261.0**	4.20
Kirin 970	CPU Delegate	**76.8**	5.84

KD: Results



Thanks!

Q&A

REFERENCES

Wang, Yiran, et al. *"Knowledge distillation for fast and accurate monocular depth estimation on mobile devices."* In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

Zhang, Hang, et al. *"Resnest: Split-attention networks."* arXiv preprint, arXiv:2004.08955, 2020.

Xian, Ke, et al. *"Monocular relative depth perception with web stereo data supervision."* In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

Wofk, Diana, et al. *"FastDepth: Fast monocular depth estimation on embedded systems"*. In International Conference on Robotics and Automation (ICRA), pages 6101–6108. IEEE, 2019.

Howard, Andrew G , et al. *"Mobilenets: Efficient convolutional neural networks for mobile vision applications."* arXiv preprint, arXiv:1704.04861, 2017.

Eigen, David, et al. *"Depth map prediction from a single image using a multi-scale deep network."* arXiv preprint arXiv:1406.2283, 2014.

Ranftl, René, et al. *"Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer."* arXiv preprint arXiv:1907.01341, 2019.

Liu, Yifan, et al. *"Structured knowledge distillation for dense prediction."* arXiv preprint arXiv:1903.04197, 2019.

Raza, Wamiq. et al. *"Energy-Efficient Inference on the Edge Exploiting TinyML Capabilities for UAVs"*. Drones, 5, 127, 2021.

TensorFlowLite. *"Homepage"*. URL: https://www.tensorflow.org/lite/ Accessed: November 2021.

Vaswani, Ashish, et al. *"Attention is all you need"*. In Advances in neural information processing systems,, 2017.

Dosovitskiy, Alexey, et al. *"An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale."* In International Conference on Learning Representations (ICLR), 2021.

Mehta, Sachin, et al. *"MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer."* arXiv preprint, arXiv:2110.02178, 2021.

Fu, Jun. *"Dual attention network for scene segmentation."* In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.