A Gift from Knowledge Distillation
Fast Optimization, Network Minimisation and Transfer Learning

Problem Statement:
A deep network with many parameters requires heavy computation for both training and testing. These deep networks are difficult to use in real-life applications because a normal computer cannot handle this work, let alone mobile devices. Therefore, many researchers have been trying to make networks smaller while maintaining the performance level using knowledge distillation techniques.

Contribution:
This paper defines the distilled knowledge to be transferred in terms of flow between layers, which is calculated by computing the inner product between features from two layers.

Method:
The flow of the solution procedure can be defined by the relationship between two intermediate results. In the case of a DNN, the relationship can be mathematically considered by the direction between features of two layers. The authors designed FSP matrix to represent the flow of the solution process. The FSP matrix $G \in \mathbb{R}^{m\times n}$ is generated by the features from two layers. Let one of the selected layers generate the feature map $F^1 \in \mathbb{R}^{h\times w\times m}$, where h, w, and m represent the height, width, and number of channels, respectively. The other selected layer generates the feature map $F^2 \in \mathbb{R}^{h\times w\times n}$. Then, the FSP matrix $G \in \mathbb{R}^{m\times n}$ is calculated by

$$G_{i,j}(x;W) = \sum_{s=1}^{h}\sum_{t=1}^{w}\frac{F^1_{s,t,i}(x;W)\times F^2_{s,t,j}(x;W)}{h\times w}$$

The cost function of transferring the distilled knowledge task is defined as

$$L_{FSP}(W_t,W_s) = \frac{1}{N}\sum_{x}\sum_{i=1}^{n}\lambda_i \times \|(G_i^T(x,W_t) - G_i^S(x,W_s))\|_2^2$$

Algorithm:
Stage 1: Learning the FSP matrix
Weights of the student and the teacher networks: W_s, W_t
$W_s = arg\ min_{W_s} L_{FSP}(W_t, W_s)$

Stage 2: Training for the original task
$W_s = arg\ min_{W_s} L_{ori}(W_s)$