Institut für Industrielle Informationstechnik
Hertzstraße 16, Geb. 06.35
76187 Karlsruhe
www.iiit.kit.edu

SiliconSoftware
speed up your vision

Gradient Based Evolution to Optimize the Structure of CNNs

Norbert Mitschke, Michael Heizmann, Klaus-Henning Noffz and Ralf Wittmann

1. Indroduction

Aspects of the use of convolutional neural networks (CNN) in industrial applications:

- machines are more accurate in detecting defects and optical characters as well as texture analysis
- the classification performance of traditional defect detectors is relatively poor
- industrial task can be solved by relatively small CNNs compared to complex image classification tasks
- falling hardware prices, especially for FPGAs, make it possible to run CNNs economically no longer only on GPUs

The challenge remains to find the topology of a suitable CNN. Therefore, we present a metaheuristic approach, that

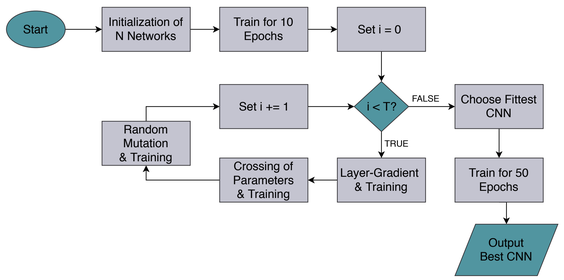
- automatically generates a suitable CNN topology
- also minimizes the needed resources
- only uses the given dataset as prior knowledge
- converges in a relatively short time

2. Approach

Our approach is based on a two-level gradient evolution. After initialization, the algorithm starts an iteration, which consists of three parts:

- gradient evolution with respect to the number of layers
- crossover among similar CNNs
- mutation to increase diversity
- training after each operation and fitness measurement

Weights are stored after training and uploaded to CNN before each step to maintain progress.



Flow chart of the presented metaheuristics.

2.1 Fitness function

We found:

$$F(a_i, l_i, m_i, s_i) = \frac{1}{1 - a_i + \epsilon} + C \cdot a_i + \frac{1}{l_i} - \sqrt{\frac{m_i}{M}} \quad \text{for } s_i < s_{\max}$$

a_i: validation accuracy
l_i: validation crossentropy
m_i: MAC ops for inference
s_i: memory for inference

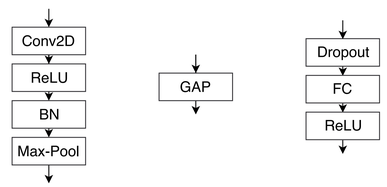
We choose $C = 10$, $M = 10^6$ and $s_{\max} = 28\,\text{MB}$ due to the resources of modern FPGAs. The purposes are:

- determine the quality of a CNN
- reach high accuracies
- lower the resource consumption of CNN's inference
- guarantee convergence

2.2 General topology and initialization

The topology is restricted to three kinds of blocks:

- up to 10 convolutional blocks
- GAP layer for reshaping
- up to 4 FC blocks



The topology is formed by stringing together these 3 elements.

During initialization $N = 24$ topologies are selected from a seed list and trained for 6 Epochs.

2.3 Iteration

10 iterations are performed that include the following steps:

- calculation of the gradient for individual i with c_i Conv and d_i FC layers and update:

$$\mathbf{g_i} = \frac{1}{P} \sum_{p=0}^{P-1} (F(i) - F(p)) \cdot \left(\begin{bmatrix} c_i \\ d_i \end{bmatrix} - \begin{bmatrix} c_p \\ d_p \end{bmatrix} \right)$$

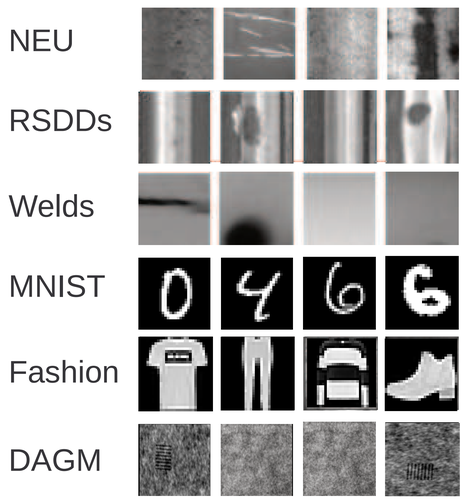
$$\begin{bmatrix} c_i \\ d_i \end{bmatrix} \leftarrow \text{srd}\left(\begin{bmatrix} c_i \\ d_i \end{bmatrix} + r \cdot \mathbf{g_i} \right), \quad r \sim U(0, 0.5)$$

- For the number of kernels we consider the exponentially growing model $k_i(l) = 2^{h_{1,i} \cdot l + h_{0,i}}$. The number of neurons in the FC layers and the parameters $h_{0,i}$ and $h_{1,i}$ of the offspring are calculated by gradient evolution (see above). The strides are determined by 2-point-crossover.
- In the mutation, all modifiable parameters from the earlier steps are randomly changed. The offspring replaces the least fit model in the population.

After each step the offspring is trained for 6 Epochs.

3. Datasets

We used six datasets that are either related to industry or comparable to industrial datasets. The datasets are presented below:



Used Datasets.

The NEU steel dataset consists of six different surface defects, while the RSDDs dataset contains images of defective railroad tracks. Defective welds are examined using the GDXray database. In addition to the more well-known MNIST and Fashion MNIST datasets, a dataset of the DAGM is also considered for which a rotated and translated patch must be recognized.

4. Results

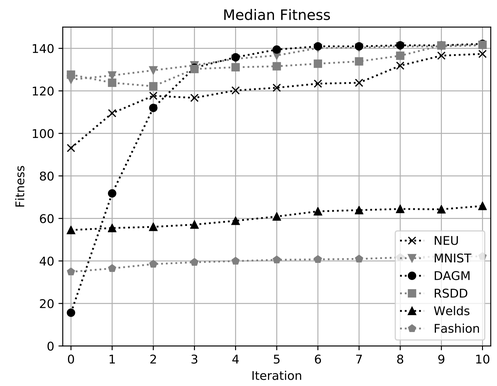
Detailed results are presented in table below. The important findings are:

- similiar CNN structures have resulted in independend runs, but outliers can also occur
- the space of possible solutions is limited by the dataset: the size is limited by overfitting upwards and the poor accuracy of small CNNs limits its size to the bottom
- the presented metaheuristic provides good results for industrial applications
- a trade-off between processing time and convergence for more complex challenges is needed

Dataset	NEU	RSDDs	Welds	MNIST	Fashion	DAGM
Min. Error	0,000	0,011	0,026	0,006	0,077	0,000
Mean Error	0,003	0,015	0,034	0,008	0,087	0,003
Parameters	250k	26k	25k	767k	77k	79k
MAC-Ops	14,6M	0,8M	2,2M	6,5M	1,4M	2,3M

In another experiment we reinitialized the weights of the resulting CNNs and trained them again. The results are:

- no CNN has reached the accuracy of the metaheuristics
- the CNNs tended to overfit
- our method seems also to look implicitly for a suitable initialization of the weights



Progress of the median fitness of all datasets.