**New-generation Operational Neural Networks for Image Processing**

**PM: Esin Guldogan**

**Date: 21.03.2021**

Change History

| Date | Version | Change Description | Author |
| --- | --- | --- | --- |
| **21.03.2021** | 0.1 | Initial draft | Esin Guldogan |
| **21.03.2021** | 0.2 | R0 draft | Serkan Kiranyaz |
|  |  |  |  |

Contents

[1 Project Background Introduction 4](#_Toc987359)

[2 Features and requirements 4](#_Toc987360)

[3 Delivery and delivery date 17](#_Toc987361)

# Introduction

Image-to-Image (I2I) applications using deep Convolutional Neural Networks or their variants have recently become quite popular. To name a few, for instance, *image denoising* is a popular field where deep Convolutional Neural Networks (CNNs) have recently been applied and achieved the state-of-the-art performance [3]-[6]. This was an expected outcome since “convolution” is the basis of the linear filtering and a deep CNN with thousands of sub-band filters that can be tuned to suppress the noise in a near-optimal way is a natural tool for image denoising. Moreover, face or object segmentation (commonly referred as “Semantic Segmentation”) in general is a common application domain especially for deep CNNs [8]-[17]. Image transformation (or sometimes called as image translation) is the process of converting one (set of) image(s) to another. Deep CNNs have recently been used for certain image translation tasks [19], [20] such as edge-to-image, gray-scale-to-color image, day-to-night (or vice versa) photo translation, etc. The conventional CNNs are strictly homogenous with the sole operator, linear convolution. Due to this constraint, while they learn very well those problems with a monotonous, relatively simple and linearly separable solution space, they may entirely fail to do so when the solution space is highly nonlinear and complex. It is, therefore, not surprising that in many challenging problems only the deep CNNs with a massive complexity and depth can achieve the required diversity and the learning performance.

This project proposal stems from the basic fact that current artificial deep neural nets are highly reductive models of biological counterparts. Given the tremendous advances in neuroscience and the recognition of highly nonlinear and heterogeneous networks, elegant wiring and interconnectivity, and diverse neuronal types (and non-neuron brain cells) that contribute to cognition, current deep learning models seem terribly over-simplistic. Indeed, the ancient linear neuron model (McCulloch-Pitts) [21]-[23] from the 1940s and 1950s is still used in feed-forward Artificial Neural Networks (ANNs) such as Multi-Layer Perceptrons (MLPs) and even in the most recent CNNs and their variants, making them entirely homogenous. As a result, despite how deep and complex networks are now being configured, the advances in many applications hit a concrete wall with insignificant performance gains. Accordingly, this project aims to address this fundamental drawback starting from the *core* problem: artificial neurons as the primary learning units. Inspired from the neuroscientific knowledge of biological neurons, a “super-neuron” model will be developed by forming its synaptic connections (synapses) inspired from two basic phenomena: 1) varying synaptic connections of heterogeneous, non-linear neurons in bio-neurological systems such as the mammalian visual system, 2) direct relation between diversity of neural operators and computational power [23] in biological neural networks wherein adding more neural diversity allows the network size and total connections to be reduced [24]. Empirically, these studies have proven that only the heterogeneous networks with the right operator set and proper training can truly provide the required kernel transformation to discriminate the classes of a given problem, or to approximate the underlying complex function. In neuroscience, this fact has been revealed as the “neuro-diversity” or more precisely, “the biochemical diversity of synaptic connections” [25]-[30].

Figure 1 illustrates the natural link between the deep CNNs used today and the ancient linear neuron model from the 1940s, and the evolution of heterogeneous networks with recent non-linear neuron models until now and beyond.CNNs borrowed the essential idea from Multi-Layer Perceptrons (MLPs) while introducing two restrictions: limited connections and weight sharing. CNNs are hence a “restricted” version of MLPs with the addition of sub-sampling layers. The fully-connected layers of CNNs are identical to MLP layers. Therefore, CNNs have inherent MLP limitations when learning a complex task. MLPs are indeed universal approximators; however, for learning tasks with compact configurations, there is a need for a “better” universal approximator. The AI-Core team has started to tackle this well-known drawback by introducing Generalized Operational Perceptrons (GOPs) [31]-[35], which aim to model biological neurons with *distinct* synaptic connections.More precisely, based on the fact that actual learning occurs in the synaptic connections with non-linear operators in general, the crude and fixed model of MLPs can now be generalized by the GOPs to allow any (blend of) non-linear transformations. The GOP model, being a superset of MLPs cannot perform worse than MLPs. This means that only when the operator search indicates that the native MLP operators should be used in all neurons of the GOP, the Homogeneous GOP will eventually become a conventional MLP. However, whenever a “different” and thus “better” operator set is found even for a neuron, or for the neurons of a hidden layer, or the entire GOP network, a superior performance will naturally be achieved. Compared to the MLPs, GOPs have demonstrated a superior diversity, encountered in biological neural networks, which resulted in an elegant performance level on numerous challenging problems where conventional MLPs entirely failed (e.g. Two-Spirals or N-bit parity problems). GOPs have even surpassed the recent Extreme Learning Machines (ELMs) and hence, become the *state-of-the-art* (dense) ANN model [33]-[35]. Following GOPs footsteps, Operational Neural Networks (ONNs) have recently been proposed [36]- [38] as a superset of CNNs. ONNs have not only outperformed CNNs significantly, but they are also capable of learning even those problems where CNNs entirely fail.However**,** ONNs like their ancestor, GOPs, also exhibited certain drawbacks such as strict dependability to the operators in the operator set library, the mandatory search for the best operator set for each layer/neuron, and the need for setting (fixing) the operator sets of the output layer neuron(s) in advance. Such drawbacks yield a limited network heterogeneity and divergence that eventually cause certain issues in learning performance and computational efficiency.



Figure 1: The time evolution of the artificial neuron and network models since 1943.

As a solution, the team has recently developed Self-organized ONNs (Self-ONNs) [39], [40] with the generative neuron model that can address all these drawbacks without any prior (operator) search or training, and with elegant computational complexity. During the training of the network, to maximize the learning performance, each generative neuron in a Self-ONN can customize the nodal operators of each kernel connection. This yields an ultimate heterogeneity level that is far beyond what ONNs can offer, and thus, the traditional “weight optimization” of conventional CNNs is entirely turned to be an “operator generation” process. Figure 2 shows a sample “learned” non-linear kernel elements of a Self-ONN versus the linear kernel elements of a deep CNN both of which were trained for image denoising under fair conditions, e.g., they have the same complexity, depth, train- and hyper-parameters.

A picture containing diagram

Description automatically generated

Figure 2: CNNs vs. Self-ONNs on the image denoising application.

# Motivation and Objectives

During the 1st year of the project, we have implemented Self-ONNs and applied to three major problems that were specified by Huawei Finland. As expected, state-of-the-art performance levels were achieved with a remarkable performance gap over deep CNN counterparts. During the 24-months timeline of this project, Self-ONNs will be the starting point and the baseline network model. However, their learning units, the “generative neurons” can only perform “localized” kernel operations, and hence the kernel size of a neuron at a particular layer solely determines the capacity of the receptive fields and the amount of information gathered from the previous layer. Obviously, using a larger-size kernel may partially address this issue; however, this will not only create an increasing complexity issue, but it is also not feasible to determine the optimal kernel size for each connection of the neuron. It is, therefore, desirable to gather information from a larger area in the previous layer maps while keeping the kernel size as is. For certain applications, it might be even more desirable “to learn” or to customize the (central) locations of each kernel during the training process along with the customized nodal operators so that both can be optimized simultaneously.

In brief, super-neurons will have the ability to optimize the (non-linear) transformations for each connection to other neurons with dynamic, and adaptable kernels that have the ability to find the optimal location for the best synaptic (kernel) operation and to adapt its receptive field size accordingly. Moreover, they will not be passive units; rather they can be upgraded or modified based on their current learning capabilities which will be evaluated continuously. Empowered by super-neurons, the new generation Self-ONNs will be self-organized, highly heterogenous, and diversified by individual neurons with distinct synaptic operations. The aimed “*self-organization*” will accomplish the following capabilities:

* *Self-generation* of the nodal operations of each kernel element (already accomplished in generative neuron model),
* *Self-localization* of each kernel operation over the previous layer output maps (ongoing work with promising results),
* *Self-adaptation* of the non-linearity level (the degree of Taylor polynomials) of each nodal operator,
* *Self-awareness* of each neuron’s contribution to the ongoing learning process using Synaptic Plasticity paradigm,
* *Self-attention* to the output maps of the previous layer neurons to scale them during the creation of the input map w.r.t their contribution to the ongoing learning process and
* *Self-determination* and self-*execution* of the necessary actions when the neuron is aware of the deterioration of its contribution over the ongoing learning process. Such actions can be resetting the kernel locations, altering the non-linearity level (order), or simply re-initializing the gradients to initiate retraining with new settings.

Overall, new-generation Self-ONNs with super neurons will have a superior learning capability over the conventional homogenous deep networks with idle, static (localized), and linear neuron model. This will eventually yield that compact configurations can then conveniently be used even for complex problems and large datasets. As a part of such an utmost “self-organization” capability, the creation of the “proto concepts” in such networks will then be possible where they can guide the ongoing learning process towards progressive acquisition in such a way that certain network connections can be easily made to learn complex concepts with little explicit training. In other words, the creation of the “artificial intuition” of these advance networks will be aimed.

**Chart

Description automatically generated**

Figure 3: Localized (left) *vs.* non-localized kernel operations to create the pixel, , from the output maps of the previous layer neurons. At the middle, *randomly localized* (uniformly distributed) kernels within a spatial *bias* range of are shown. At the right, the *BP-optimized* locations of each kernel during a particular iteration of BP training are illustrated.

For clarity, we draw the focus on the *self-localization* capability. Recall that the generative neurons in the current Self-ONN model can “create” the best possible operator for the kernel of each connection during training. However, they are still “localized” and will have neither the means of locating at the “right place” nor the capability of enhancing their “receptive field” that is limited to the kernel-size. In order to improve the receptive field size and even to find the best possible kernel locations, we will develop non-localized kernel operations for Self-ONNs embedded in a novel and superior neuron model compared to the generative neurons. The new neurons are hence called “super (generative) neurons”. We will focus particularly on two models of the super neurons for the localization process of the kernels: i) randomly localized (uniformly distributed) kernels within a bias range set for each layer, ii) BP-optimized locations of each kernel. Particularly in the latter model, “what” operator should be used and “where” it should be located, are simultaneously optimized during the BP training. In this way, we shall achieve the goal of doing the *right* transformation at the *right* (kernel) location of the *right* connection to maximize the learning performance.

Figure 3 (left) illustrates this where a pixel of the *i*th neuron in layer *l*+1, , is created by the 9 pixels of the previous layer output maps, , for operated with the kernels centered at the same location, , given that is the number of neurons in the previous layer, *l*. This gives rise to an obvious limitation since the kernel is blinded to the neighboring pixels which can potentially provide a meaningful contribution to the input pixel, and hence should not be excluded. This study provides a possible solution by proposing two super neuron models with *non-localized* kernel operations as illustrated in the middle and right of the figure. We define a set of additional parameters, the spatial bias, defined as the deviation of the kernel from the pixel location, , towards x- and y-direction for the *k*th output neuron connection to the *i*th neuron input map at layer *l*+1 represented as . In the (bottom) left illustration, the 3x3 kernels are randomly located within a bias range of **,** and now the pixels within the region of 11x11 pixels can contribute when they belong to the kernel of an individual connection to a particular output map. In the figure, different colored kernels are from different connections and their corresponding bias values within the 11x11 region (the outer red-dashed square) are randomly set in advance. For instance, the bias for the 1st connection (black) is, =4, =3 pixelswhereas for the 3rd connection it is =0, =0, respectively. Finally, in the illustration at the right, the bias values are now real numbers without any range set in advance, since they are iteratively optimized by the BP training along with other network parameters. At the end of the training, they are expected to converge to a (local) optimum point. So, the bottom illustration only shows the instantaneous bias values for some connections at a particular BP epoch.

To obtain such a non-localized kernel for the *i*th neuron in layer *l*+1, connected to the *k*th neuron in layer *l* with integer bias in x- and y-directions, , respectively, let be the shift operator for by the bias, Then we can perform the shift on the output map of the *k*th neuron in layer *l*, to obtain, , and then operate with the original kernel, to obtain the input map of the layer *l*+1, **,** as expressed in Eq. (1). For generative neurons of Self-ONNs recall that is the composite nodal function which is the *Q*th order Mac-Laurin series. For the kernel element, , is expressed in Eq. (2) where the DC bias term, , is omitted. Therefore, the generative neuron has a 3D kernel matrix where the *q*th weight of the kernel element is represented by .

|  |  |
| --- | --- |
|  | (1) |

|  |  |
| --- | --- |
|  | (2) |

We have recently accomplished the derivation of the loss (error) gradient computation for each bias pair, , of each kernel where it can be mathematically proven that the optimal kernel locations can iteratively converge through back-propagation (BP), which demonstrates the “self-location” feature of the super-neurons.

In brief, during the next two years, we are aiming for further performance boost with the following improvements:

* R&D efforts on the new generation Self-ONNs, which have the true “Self-Organization” capabilities mentioned earlier on the new super-neuron variants,
* instead of fixing to some pre-set values in advance, self-optimizing the order of Maclaurin polynomials, *Q*, per layer and even per neuron,
* adapting a better optimization scheme for training, e.g., SGD with momentum [23], AdaGrad [24], RMSProp [25], Adam [26] and its variants [27], all of which should be adapted for Self-ONNs for a proper functioning,
* and optimizing also the pool and activation operators during BP training.

Ultimately, we aim to achieve the learning and generalization capability of deep s-o-a CNNs with new-generation Self-ONNs with the minimal complexity and depth thanks to the its diversity, heterogeneity and self-organization capabilities. Next, we shall share the initial results on one of the challenging image enhancement application, image deblurring.

# Preliminary Results

## This project will develop new generation Self-ONNs for image transformation/enhancement tasks such as super resolution, deblurring and denoising. Preliminary studies have shown outperforming results as some of the selected problems presented below.

1. Real-World Denoising (with *generative* neurons),
2. Deblurring (with super neurons), and
3. Image Transformation (with super neurons)

### Real-World Denoising

We present two sets of comparisons in this section. First, a comparison of Self-ONN networks with the baseline CNN network is presented across the benchmark datasets. Afterward, we evaluate the trained models on public benchmarks provided by SIDD and DnD datasets and compare the Self-ONN networks with the baseline CNN as well as contemporary works.

***Table 1. Comparison of baseline CNN with corresponding Self-ONN architectures.***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **RENOIR** | | **CC** | | **SIDD** | |
|  | ***PSNR*** | ***SSIM*** | ***PSNR*** | ***SSIM*** | ***PSNR*** | ***SSIM*** |
| Noisy (Input) | 27.38 | 0.7459 | 34.68 | 0.9436 | 23.66 | 0.5413 |
| DnCNN (Baseline CNN) | 32.34 | 0.9034 | 36.25 | 0.9689 | 36.06 | 0.9380 |
| Dn-SelfONN-17 | **34.10↑** | 0.9317**↑** | **37.52↑** | ***0.9800*↑** | **36.95↑** | ***0.9506*↑** |
| Dn-SelfONN-8 | 33.88**↑** | ***0.9350*↑** | 35.20↓ | 0.9760**↑** | 36.24**↑** | 0.9491**↑** |
| Dn-SelfONN-4\_3 | 33.06**↑** | 0.9227**↑** | 33.83↓ | 0.9679↓ | 35.39↓ | 0.9437**↑** |
| Dn-SelfONN-4\_5 | 33.26**↑** | 0.9261**↑** | 34.85↓ | 0.9711**↑** | 35.41↓ | 0.9444**↑** |

Table 1 and Table 2 present the results and comparisons over the s-o-a DnCNN (baseline CNN) and other contemporary s-o-a denoising methods, respectively. Visual comparison of denoising results on high-resolution noisy images from the RENOIR and CCNoise datasets can be seen in Figure 4 and Figure 5, respectively.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Noisy** | **Clean** | **DnCNN-17** | **Dn-SelfONN-17** | **Dn-SelfONN-8** | **Dn-SelfONN-4** |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

Figure 4. Visual comparison of denoising results on high-resolution noisy images from the RENOIR dataset.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Noisy** | **Clean** | **DnCNN** | **Dn-SelfONN-17** | **Dn-SelfONN-8** | **Dn-SelfONN-4** |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |

Figure 5. Denoising results of the networks used in this study on example images from the CCNoise dataset.

***Table 2. Results on DND public benchmark.***

|  |  |  |
| --- | --- | --- |
| **Network** | **PSNR** | **SSIM** |
| Noisy | 29.84 | 0.7018 |
| BM3D | 34.51 | 0.8507 |
| MLP | 34.23 | 0.8331 |
| TNRD | 33.65 | 0.8306 |
| KSVD | 36.49 | 0.8978 |
| FFDNet+ | 37.61 | 0.9415 |
| DnCNN+ | 37.90 | 0.9430 |
| TWSC | 37.94 | 0.9416 |
| CBDNet | 38.06 | 0.9421 |
| DnCNN (Baseline) | 38.10 | 0.9323 |
| Dn-SelfONN-17 | 38.48 | 0.9404 |
| Dn-SelfONN-8 | 38.12 | 0.9356 |
| Dn-SelfONN-4\_3 | 37.56 | 0.9288 |
| Dn-SelfONN-4\_5 | 37.62 | 0.9283 |

### Deblurring

Image deblurring [41]-[53] can broadly be categorized as: 1) kernel-based estimation [41]- [47], 2) end-to-end [48]-[53]. Deep CNNs have been used for each category but in this study, we shall evaluate the networks in an “end-to-end” configuration, that does not have to estimate the blurring kernel, rather the blurred image is directly transformed into the restored (deblurred) image. We expect that super-neurons with the non-localized kernel operations would have a superior performance because image deblurring usually requires a large receptive field for enhanced global knowledge while the conventional CNNs (and Self-ONNs) can provide local knowledge limited with the size of their filters. We consider two blurring problems:

* *Disc()* blurring: a circular averaging filter (pillbox) within the square matrix of size, .
* *Motion* blurring: the linear motion of a camera where   specifies the length of the motion and  specifies the angle of motion in degrees in a counter-clockwise direction.

Our aim is to evaluate the learning capability of the super-neurons over the conventional neuron models under harsh conditions and for this reason, along with the aforementioned restrictions, we apply a severe blurring with the following parameter settings: . *Disc()* basically applies an averaging of pixels and *Motion* approximates a linear motion of 11 pixels diagonally. Both blurring artifacts can sometimes cause such a severe image degradation that makes it difficult or even infeasible to comprehend the content of the image. Finally, for the random bias ranges for super-neurons where , are set as for the1st, 2nd and output layers, respectively. Thus, with this setting, for instance, the 1st layer super-neurons will have the improved size of the receptive fields as 11x11 pixels, which is significantly larger than the original kernel size of 3x3 pixels.

.

A close up of a map

Description automatically generated A close up of a map

Description automatically generated

A close up of a map

Description automatically generated A close up of a map

Description automatically generated

Figure 6: Best SNR levels for each *Disc-5* (top) and *Motion* (bottom) deblurring fold achieved by the corresponding Self-ONN (with no, random and BP-optimized spatial biases) and the CNNx4. The 11th fold result is the average of the 10-folds.

Finally, for the random bias ranges for super-neurons where , are set as for the1st, 2nd and output layers, respectively. Thus, with this setting, for instance, the 1st layer super-neurons will have the improved size of the receptive fields as 11x11 pixels, which is significantly larger than the original kernel size of 3x3 pixels.

Figure 5 (top part) shows SNR plots of the best (in training) *Disc-5* Deblurring results per fold over both train (left) and test (right) partitions. The average test SNR levels achieved by the three Self-ONNs are 5.79dB (no bias = ‘0-0-0’), 6.63dB (random bias = ‘4-4-2’), 6.71dB (BP-optimized bias = ‘Opt’), respectively whilst the CNNx4 has the SNR of 5.17dB. In both train and test partitions and all folds Self-ONNs achieve significantly higher performance as compared to CNNs despite the fact that they have four times less neurons. In particular, the average performance gap between CNNs and Self-ONNs with super-neurons is widened around 1.5dB and 1.2dB in train and test partitions, respectively. Finally, the Self-ONNs with super-neurons can achieve higher than 0.8dB (train) and 0.6dB (test) on average compared to Self-ONNs with generative neurons.

For a visual evaluation, Figure 6 shows a set of original (target), *Disc-5* and *Motion* blurred (input) images along with the corresponding outputs of CNNx4, Self-ONNs (with no, random and BP-optimized spatial biases) from the test partition. The superior deblurring performance of Self-ONNs with super-neurons is visible in all outputs.



Figure 7: Some typical original (target) and *Disc-5* blurred (left) and *Motion* blurred (right) input images and the corresponding outputs of the CNNx4 and the three Self-ONNs (with no, random and BP-optimized spatial bias) from the *test* partition.

### Image Transformation

In this example we tackled a more challenging image transformation, which is transforming an image to entirely different image. This is also much harder than the image syntheses problem because this time the problem is the creation of a (set of) image(s) from another with a distinct pattern and texture. To make the problem even more challenging, we have trained a single network to (learn to) transform 4 (target) images from 4 input images, as illustrated in Figure 8. In the first fold, we have further tested whether the networks are capable of learning the “inverse” problems, which means, the same network can transform a pair of input images to another pair of output images and also do the opposite (output images become the input images).

Table 3: Image Transformation results with compact networks.

|  |  |  |  |
| --- | --- | --- | --- |
| **Network** | **Neurons** | **# of Parameters (k)** | **PSNR (dB)** |
| CNN | 96 | 54401 | 19.22 |
| Self-ONN with generative neurons | 48 | 43905 | 24.22 |
| Self-ONN with super neurons (Random bias) | 48 | 43905 | 28.46 |
| Self-ONN with super neurons (Optimized bias) | 48 | 43969 | **28.54** |

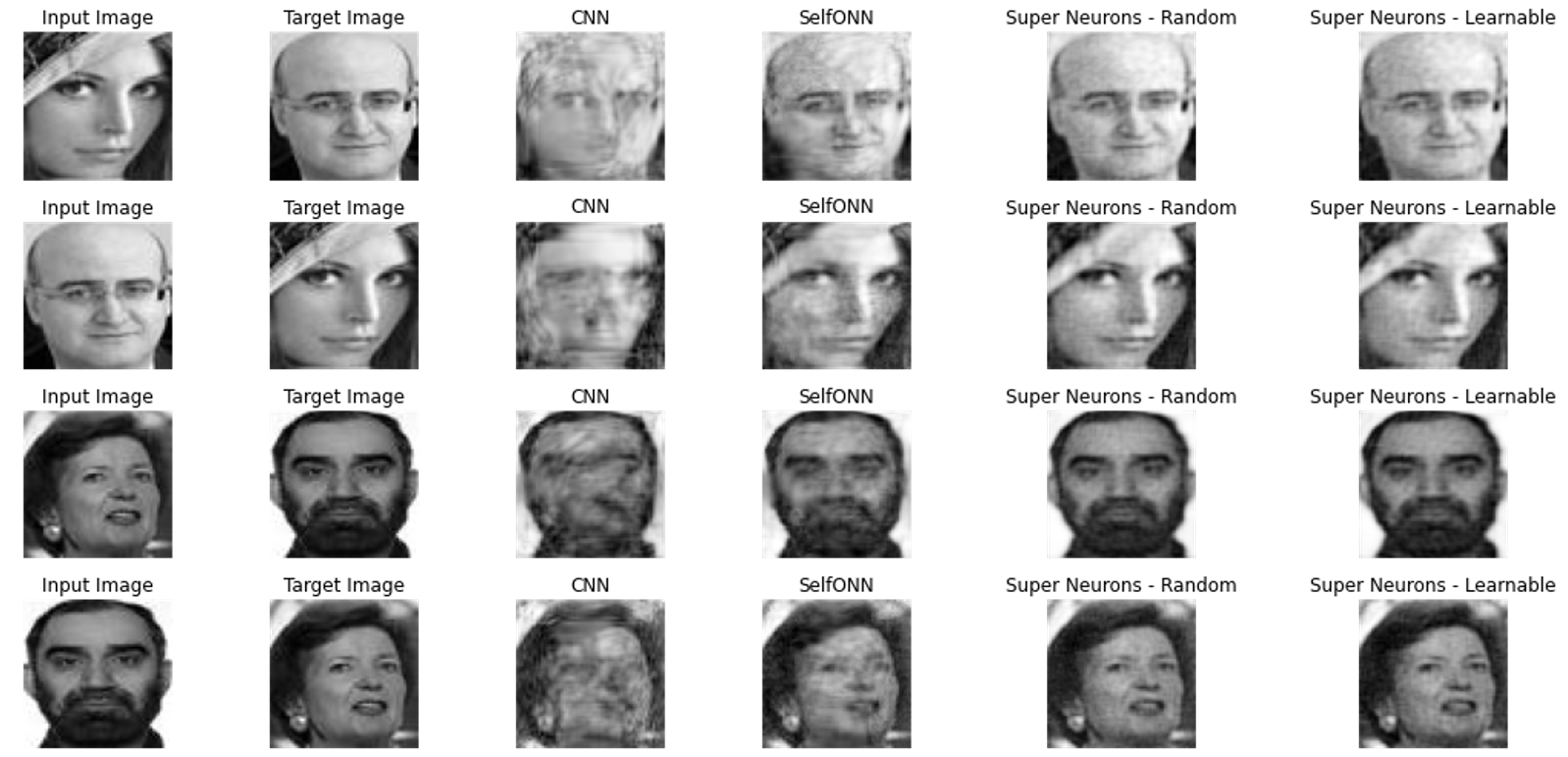


Figure 8: Image transformation results of the three Self-ONNs (one with generative and two with super-neurons) and the CNN with twice more neurons.

# Work plan

### Left it as a TO DO for Esin/Mehmet:

This can be a 2-year project where the 1st year is for the theory, R&D on super neurons and finalizing the new-generation Self-ONN while the 2nd year can focus on the applications:

* Denoising
* Super Resolution
* Deblurring
* Object detection/localization (Creating a ONN-YOLO with minimal depth/complexity)
* Face segmentation/recognition.

### Work Package 1: Theoretical R&D of the ONNs such as finding the optimal operator sets for each hidden neuron, optimizing the ONN configuration improving BP convergence, developing new learning techniques such as modified Adam or Nesterov for ONNs, etc.

### Work Package 2: SW development of the Python-Cuda code for ONNs along with the speed optimization, and ultimately to test the ONN on benchmark datasets such as benchmark (or Huawei specific) denoising-deblurring datasets and object classification datasets (e.g., CIFAR-100, Caltech-101, MNIST, MNIST Fashion, and ImageNet).

**Table 2-1: Key requirements**

| No. | Key requirements | Requirement descriptions (Please list the cooperation content and technical metrics, and specify Technological requirements for unquantifiable items.) |
| --- | --- | --- |
| *1* | Data acquisition | * ONN will be used as main engine for image enhancement methods. As a consequence, databases for training ONN are needed and will be captured during the work. * The captured databases will be made available to Huawei. * The methods to capture the database and their content will also be agreed between the student, and the Huawei and TAU supervisors of the student. * SW utilized in the process of capturing data and building the databases will also be shared with Huawei. |
| *2* | *ONN system* | * An optimized and compact ONN will be designed and implemented for selected image enhancement task such as denoising, deblurring etc. * Source codes + models will be delivered to Huawei. |
| *3* | *Evaluation and reporting* | * Performance of the proposed system will be compared against the state-of-art systems over benchmark datasets. * Results and final report will be delivered to Huawei. * One scientific publication submitted to top tier peer-reviewed conference or journal. For instance, CVPR, ICCV, ECCV, NeurIPS, ICML, ACCV, BMVC, 3DV, IROS, ICRA, ICPR, WACV, FUSION, AISTATS, TPAMI, IJCV or PR. |

# Delivery and delivery date

| Phase | Deliverables | Delivery Date |
| --- | --- | --- |
| 1 | * Potential collected dataset(s) will be delivered to Huawei * Source codes, models and results will be delivered * Final report with evaluation results * One scientific publication submitted to top tier peer-reviewed conference or journal. For instance CVPR, ICCV, ECCV, NeurIPS, ICML, ICIP, ACCV, BMVC, 3DV, IROS, ICRA, ICPR, WACV, FUSION, AISTATS, TPAMI, IJCV or PR. | *latest 24 months from the project start date* |

5. Budget

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Description or Item** | **Name** | **Title** | **Contributes to task** | **Workload in man months** | **Price rate/man month** | **Cost（EUR）** |
| Staff Cost | NN1 | Researcher | WP1 | 11 | 7575 | 83325 |
| NN2 | Researcher | WP2 | 11 | 7575 | 83325 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Travel Cost (Please indicate the number of trips and people to Huawei China, if any) | NN | Researcher | Conference |  |  | 2000 |
| NN | Researcher | Conference |  |  | 2000 |
| Serkan Kiranyaz  Moncef Gabbouj | Professor  Professor | WP1-WP2  1-2 persons visiting Huawei, China |  |  | 3000  3000 |
| Others |  |  |  |  |  |  |
| Overhead(Please input percentage, e.g. 10%) @10% |  |  |  |  |  | 17665 |
| Total |  |  |  |  |  | 194315 |

**References**

1. I. Soltesz, “Diversity in the neuronal machine: order and variability in interneuronal microcircuits,” Oxford University Press, USA, 2002
2. Z. Nusser, “Variability in the subcellular distribution of ion channels increases neuronal diversity,” Trends Neuroscience, pp. 267-274, 2009.
3. V. Jain, S. Seung, "Natural image denoising with convolutional networks", Proc. Adv. Neural Inf. Process. Syst., pp. 769-776, 2009.
4. S. Lefkimmiatis, "Non-local color image denoising with convolutional neural networks", Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 3587-3596, 2017.
5. D. Yang and J. Sun, "BM3D-Net: A Convolutional Neural Network for Transform-Domain Collaborative Filtering," in IEEE Signal Processing Letters, vol. 25, no. 1, pp. 55-59, Jan. 2018. doi: 10.1109/LSP.2017.2768660
6. Y. Chen, W. Yu, T. Pock, "On learning optimized reaction diffusion processes for effective image restoration", Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 5261-5269, 2015.
7. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion", J. Mach. Learn. Res., vol. 11, no. Dec., pp. 3371-3408, 2010
8. S. Kiranyaz, M.-A. Waris, I. Ahmad, R. Hamila, and M. Gabbouj, “Face Segmentation ın thumbnail Images by Data-Adaptive Convolutional Segmentation Networks”, Int. Conf. on Image Processing, ICIP’16, Phoenix, Arizona, USA, September 25-28, 2016.
9. Li Y., Qi H., Dai J., Ji X., Wei Y. “ Fully convolutional instance aware semantic segmentation”, in Computer Vision and Pattern Recognition (CVPR), pp. 4438–4446, 2017.
10. Lin T., Maire M., Belongie S., Hays J., Perona P., Ramanan D., Dollar´ P., Zitnick L. “Microsoft COCO: Common objects in context,” in ECCV, pp. 740–755, 2014.
11. H. Zhu, F. Meng, J. Cai, and S. Lu, “Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation,” Journal of Visual Communication and Image Representation, vol. 34, pp. 12–27, 2016.
12. J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440, 2015.
13. O. Ronneberger, P. Fischer, T. Brox, "UNet: convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234- 241, 2015.
14. J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 3431-3440. doi: 10.1109/CVPR.2015.7298965
15. E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, “Labeled faces in the wild: A survey,” in Advances in Face Detection and Facial Image Analysis, pp. 189–248, Springer Nature, 2016.
16. F. Yu and V. Koltun, “Multi-scale context aggregation by dilated convolutions,” CoRR, vol. abs/1511.07122, 2015.
17. P. Luc, C. Couprie, S. Chintala, and J. Verbeek, “Semantic segmentation using adversarial networks,” CoRR, vol. abs/1611.08408, 2016.
18. V. Jain, E. Learned-Miller, “FDDB: A benchmark for face detection in unconstrained settings”, Technical Report UM-CS2010-009, University of Massachusetts, Amherst (2010).
19. P. Isola, J. Y. Zhu, T Zhou, A. A. Efros , "Image-to-image translation with conditional adversarial networks", Proc. of the IEEE Conf.on Computer Vision and Pattern Recognition (CVPR), 2017.
20. J. Y. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks", Proc. of the IEEE Conf.on Computer Vision and Pattern Recognition (CVPR), 2017.
21. A. Cochocki and R. Unbehauen, “Neural Networks for Optimization and Signal Processing” (1st ed.). Book: John Wiley & Sons, Inc., New York, NY, USA, 1993.
22. S. Haykin, "Neural Networks and Learning Machines (3rd Edition)", Book: Prentice Hall, Nov. 2008.
23. W. McCulloch and [W.](https://en.wikipedia.org/wiki/Walter_Pitts) Pitts, “A logical calculus of the ideas immanent in nervous activity”, Bulletin of Mathematical Biophysics, 5:115–133, 1943.
24. N.s B. Karayiannis, M. M. Randolph-Gips, “The Construction and Training of Reformulated Radial Basis Function Neural Networks”, *IEEE Trans. on Neural Networks*, pp. 835-844, 2003.
25. T. Klausberger, P. Somogyi, “Neuronal diversity and temporal dynamics: the unity of hippocampal circuit operations” , *Science*, 321, pp. 53-57, (5885), 2008.
26. Z. Nusser, “Variability in the subcellular distribution of ion channels increases neuronal diversity”, *Trends Neuroscience*, pp. 267-274, May 2009.
27. E. Marder, J.M.Goaillard, “Variability, compensation and homeostasisin neuron and network function”, *Nature Reviews* Neuroscience, pp. 563-574, 2006.
28. R. H. Masland, “Neuronal diversity in the retina”, Current Opinion in Neurobiology, pp. 431-436, 2001.
29. C. I. Moore, M. Carlen, U. Knoblich, J. A. Cardin, “Neocortical interneurons: from diversity, strength”, *Cell*, pp. 189-193, 2010.
30. I. Soltesz, “Diversity in the Neuronal Machine: Order and Variability in Interneuronal Microcircuits”, Oxford University Press, USA, 2002.
31. S. Kiranyaz, T. Ince, A. Iosifidis and M. Gabbouj, “Generalized model of biological neural networks: progressive operational perceptrons,” IJCNN, 2017
32. S. Kiranyaz, T. Ince, A. Iosifidis and M. Gabbouj, “Progressive operational perceptrons,” Neurocomputing, 224:142–154, 2017
33. D.T. Tran, S. Kiranyaz, M. Gabbouj and A. Iosifidis, “Progressive operational perceptron with memory,” arXiv:1808.06377, 2018
34. D.T. Tran, S. Kiranyaz, M. Gabbouj and A. Iosifidis, “Heterogeneous Multilayer Generalized Operational Perceptron”, *IEEE Trans. on Neural Networks and Learning Systems*, pp. 710-724, vol. 31, May 2019.
35. D.T. Tran, S. Kiranyaz, M. Gabbouj and A. Iosifidis, "Knowledge Transfer for Face Verification Using Heterogeneous Generalized Operational Perceptrons", IEEE Int. Conf. on Image Processing, Taipei, Taiwan, 2019.
36. S. Kiranyaz, T. Ince, A. Iosifidis and M. Gabbouj, “Operational Neural Networks”, Neural Computing and Applications (Springer-Nature), pp. 1-24, Mar. 2020. DOI: <https://doi.org/10.1007/s00521-020-04780-3>
37. S. Kiranyaz, J. Malik, H. B. Abdallah, T. Ince, A. Iosifidis, M. Gabbouj, “Exploiting Heterogeneity in Operational Neural Networks by Synaptic Plasticity”, Neural Computing and Applications (Springer-Nature), arXiv:2009.08934*,* 2020.
38. J. Malik, S. Kiranyaz, M. Gabbouj, “FastONN--Python based open-source GPU implementation for Operational Neural Networks”, arXiv preprint arXiv:2006.02267, Mar. 2020.
39. S. Kiranyaz, J. Malik, H. B. Abdallah, T. Ince, A. Iosifidis, M. Gabbouj, “Self-Organized Operational Neural Networks with Generative Neurons”, *arXiv:2004.11778,* 2020.
40. J. Malik, S. Kiranyaz, M. Gabbouj, “Self-Organized Operational Neural Networks for Severe Image Restoration Problems”, Neural Networks (Elsevier), 2021. (In Print)
41. A. Chakrabarti, “A neural approach to blind motion deblurring,” CoRR, vol. abs/1603.04771, 2016. [Online]. Available: <http://arxiv.org/abs/1603.04771>
42. D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. van den Hengel, and Q. Shi, “From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.
43. M. Noroozi, P. Chandramouli, and P. Favaro, “Motion deblurring in the wild,” CoRR, vol. abs/1701.01486, 2017. [Online]. Available: <http://arxiv.org/abs/1701.01486>
44. C. J. Schuler, M. Hirsch, S. Harmeling, and B. Sch ¨olkopf, “Learning to deblur,” CoRR, vol. abs/1406.7444, 2014. [Online]. Available: http://arxiv.org/abs/1406.7444
45. J. Sun, W. Cao, Z. Xu, and J. Ponce, “Learning a convolutional neural network for non-uniform motion blur removal,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
46. U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, “Discriminative non-blind deblurring,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2013.
47. C. J. Schuler, H. Christopher Burger, S. Harmeling, and B. Scholkopf, “A machine learning approach for non-blind image deconvolution,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2013.
48. O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, “Deblurgan: Blind motion deblurring using conditional adversarial networks,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.
49. S. Nah, T. Hyun Kim, and K. Mu Lee, “Deep multi-scale convolutional neural network for dynamic scene deblurring,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.
50. S. Ramakrishnan, S. Pachori, A. Gangopadhyay, and S. Raman, “Deep generative filter for motion deblurring,” CoRR, vol. abs/1709.03481, 2017. [Online]. Available: http://arxiv.org/abs/1709.03481
51. X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, “Scale-recurrent network for deep image deblurring,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.
52. J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M.-H. Yang, “Dynamic scene deblurring using spatially variant recurrent neural networks,” in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.
53. A. Odena, V. Dumoulin, and C. Olah, “Deconvolution and checkerboard artifacts,” Distill, 2016. [Online]. Available: http://distill.pub/2016/deconv-checkerboard
54. N. Qian, On the momentum term in gradient descent learning algorithms, Neural Networks. 12 (1999) 145–151. doi:10.1016/S0893-6080(98)00116-6.
55. J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, COLT 2010 - 23rd Conf. Learn. Theory. (2010) 257–269.
56. T. Tieleman, G. Hinton, Lecture 6.5 - RMSProp, Neural Networks for Machine Learning | Coursera, (n.d.).
57. K. Diederik, J.L. Ba, ADAM: A Method for Stochastic Optimization, AIP Conf. Proc. 1631 (2014) 58–62. doi:10.1063/1.4902458.
58. S. Ruder, An overview of gradient descent optimization algorithms, (2016).