

A New Convolutional Network-in-Network Structure and Its Applications in Skin Detection, Semantic Segmentation, and Artifact Reduction

Yoonsik Kim, Insung Hwang, Nam Ik Cho
 Dept. of Electrical and Computer Engineering
 Seoul National University, Seoul, Korea

terryoo@ispl.snu.ac.kr, coee55@gmail.com, nicho@snu.ac.kr

Abstract

The inception network has been shown to provide good performance on image classification problems, but there are not much evidences that it is also effective for the image restoration or pixel-wise labeling problems. For image restoration problems, the pooling is generally not used because the decimated features are not helpful for the reconstruction of an image as the output. Moreover, most deep learning architectures for the restoration problems do not use dense prediction that need lots of training parameters. From these observations, for enjoying the performance of inception-like structure on the image based problems we propose a new convolutional network-in-network structure. The proposed network can be considered a modification of inception structure where pool projection and pooling layer are removed for maintaining the entire feature map size, and a larger kernel filter is added instead. Proposed network greatly reduces the number of parameters on account of removed dense prediction and pooling, which is an advantage, but may also reduce the receptive field in each layer. Hence, we add a larger kernel than the original inception structure for not increasing the depth of layers. The proposed structure is applied to typical image-to-image learning problems, i.e., the problems where the size of input and output are same such as skin detection, semantic segmentation, and compression artifacts reduction. Extensive experiments show that the proposed network brings comparable or better results than the state-of-the-art convolutional neural networks for these problems.

1. Introduction

In recent year, various convolutional networks have been developed for the applications from low to high level vision problems such as classification, object detection, image restoration and segmentation problems [23, 37, 40, 8, 44, 12, 28, 45]. Among many problems, this paper focuses

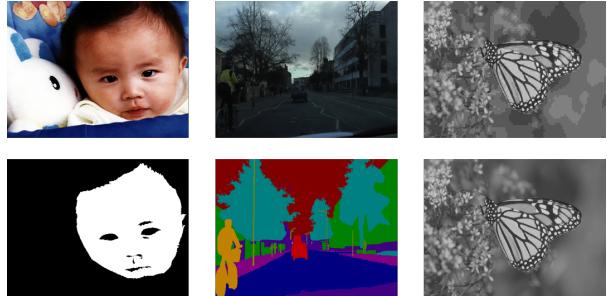


Figure 1. Example of image-to-image deep learning problems where dimension of input and output (label) is same: (from left to right) skin detection, semantic segmentation, and compression artifact reduction

on developing a convolutional network that works for image segmentation and filtering problems. For the semantic segmentation, Long *et al.* proposed a convolutional network [28], which adopts a classification network as an encoder part for exploiting pre-trained features. Then the entire image is observed by using a pooling and converting the fully connected layer to a convolution layer. However, the labeling results seem somewhat coarse due to the limitation of reconstruction part which is consist of deconvolution layer and up sampling layer. Some researchers tried to alleviate this weakness by cascade deconvolutional network training [29] or indexing up sampling layer [1].

Meanwhile, convolutional network approaches for lower class pixel-wise classification such as salient region detection [44, 53], surface normal classification, and edge orientation classification [45] have also been developed, which may be regarded as specific applications of semantic segmentation. In these works, they tried to find appropriate convolutional networks and post processing steps to the convolutional network output. In many of the above stated works and many other recent works, a classification network model [27] is usually adopted for exploiting the pre-trained features. However, it is not clear whether adopting a classification network is also effective for the pixel-wise

labeling problems with small number of labels. Further, for the problems where the input and output have the same dimension as in many of image restoration and labeling problems discussed so far, since successive dimension reduction by pooling reduces specific pixel information, it is not helpful at the image reconstruction stage.

From these reasoning, we propose a new convolutional network-in-network structure that can be applied the image-to-image deep learning problems as in Figure 1. The proposed structure can be considered a modification of inception network, where the pooling is removed and a larger kernel is added instead. Our main contributions are summarized as follows.

- We propose a new inception-like convolutional network-in-network structure, which consists of convolution and rectified linear unit (ReLU) layers only. That is, we exclude pooling and subsampling layer that reduce feature map size, because decimated features are not helpful at the reconstruction stage. Hence, it is able to do one-to-one (pixel wise) matching at the inner network and also intuitive analysis of feature map correlation.
- Proposed architecture is applied to several pixel-wise labeling and restoration problems and it is shown to provide comparable or better performances compared to the state-of-the-art methods.

In the rest of this paper, overview of related methods is introduced in section 2, and the proposed structure is presented in section 3. Analysis and comparison are provided in section 4 and this paper is concluded in section 5.

2. Related Works

Since the proposed architecture is very simple in that it uses only convolution layer and ReLU, which are widely known and used, we skip the review of these. Instead we review the areas that we apply our architecture: skin detection, semantic segmentation, and compression artifacts reduction as illustrated in Figure 1.

2.1. Skin Detection

Skin detection is to locate skin pixels or regions in images, which is an important pre-processing step in many image processing and computer vision tasks. For example, it can be used in image enhancement [51], face and human detection [13], gesture analysis [11], pornographic contents filtering [16], surveillance systems [52], etc. Hence there have been a large number of skin detection algorithms, which is well summarized in [17, 42, 2]. According to these works, the conventional methods find skin pixels that

fit to parametric models [13, 48, 25], nonparametric models [36, 16, 20], or some skin-cluster defined regions in certain color spaces [47]. There are also some methods that detect human related features (hands, faces, and body) first for finding skin pixels [26, 2], while most of the existing works detect skin pixels for finding human. In addition, there are neural network methods that try to find skin areas such as adaptive neural networks [31] and self-organizing maps [5]. But they are quite old dated and do not use open database so that direct comparison with existing neural network methods seems very difficult. Even though there have been many algorithms, the skin detection is still considered a challenging problem due to diverse variations of images, illumination conditions, skin color variations of races and makeup, and skin-like background.

2.2. Semantic Segmentation

Pixel-wise semantic segmentation problem is to classify each pixel, *i.e.*, to label a pixel as one of classes. As some examples of non-convolutional network approaches, hand-crafted features are extracted from image patches and they are trained by random forest classifier [35, 4] or boosting [38]. After the success of convolutional network to classification problems [23, 37, 40], many researchers tried to apply convolutional network to semantic segmentation problem with patch based method [6, 9, 32]. More recently, FCN [28, 34] which contains classification network and deconvolution layers as encoder and decoder has been proposed. The encoder is a modified VGG [37] network which converts the original fully connected layer to a convolution layer. Therefore, the encoder fully reflects entire image and thus obtained the best performance on benchmark PASCAL VOC 2012 dataset. Many researchers also proposed new networks to improve the performance of FCN [29, 1].

2.3. Reduction of Compression Artifacts

Lossy compression leads to image degradation which is called compression artifacts such as blocking, ringing, and blurring artifacts. Conventional non-convolutional network approaches manipulate the artifacts in spatial domain [33, 46, 43] or frequency domain [10], and more recent work using convolutional network has shown that deep learning approach provides better results than the above state methods [7]. After this work, many researchers also proposed various kinds of artifact reduction networks [39, 50].

3. Proposed Convolutional Network

After the convolutional network with inception module [40] has been developed, there have been many variants and applications adopting the inception module for the classification problems. However, it seems there are not much evidences that the inception based network is also effective for the image-to-image problems. To be precise, al-

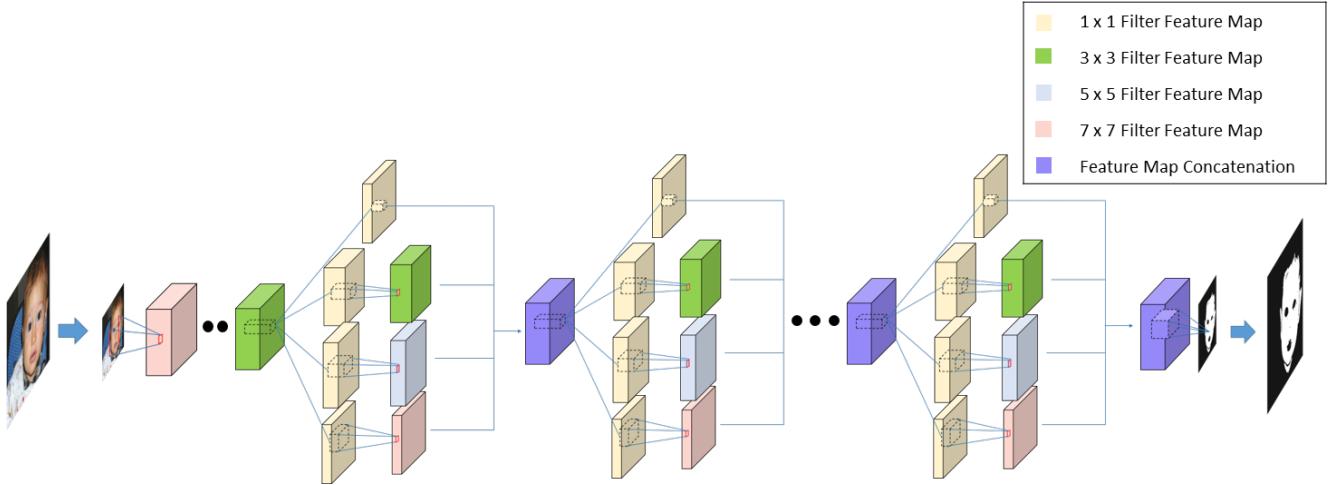


Figure 2. Proposed Network-in-Network Structure applied to skin detection. The architecture is composed of 4 convolution layers with nonlinear layers and 8 modified inception modules. Input image goes through the convolutional network to be a skin probability map at the output.

though the inception module has the advantage that it extracts scale invariant features by using multi-size kernels and thus shows good performance on classification problems, it cannot be effectively used for image-to-image problems due to decimated features by pooling layers. As an example with the image restoration problems, the pooling and decimation are generally not used because we need to keep the features from the original size image for the successful reconstruction of output image [21, 7, 8]. Further the restoration problems do not employ dense prediction method which is usually used in classification problems, because dense prediction needs lots of train parameters. Hence, for taking advantage of network-in-network structure for the image-to-image problems, we propose a new convolutional module which can be considered a modification of inception module. Specifically, we remove pooling in the inception module and add a larger size kernel instead to widen the receptive field which might have been reduced by the removal of pooling.

The overall convolutional network architecture using the proposed module is shown in Figure 2, which is consisted of several convolutional modules and the cascade of proposed multi-kernel network-in-network modules. The figure shows the application of the architecture in skin detection problem, and the same architecture is also applied to semantic segmentation and compression artifacts removal except that the input can be a patch or decimated image, and also that the depth of the network can be changed. In the following, we explain the details of proposed module and overall architecture.

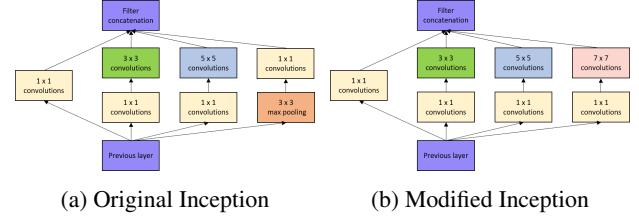


Figure 3. Inception modules

3.1. Proposed Inception Module

Christian *et al.* [40] proposed GoogLeNet with an inception module that extracts multi-scale feature maps by using multiple filters with different sizes, *e.g.*, 1×1 , 3×3 , 5×5 , and also by using pool projection layer which is composed of 1×1 convolution filter and max pooling to make pooling feature maps. To reduce the number of parameters, an additional 1×1 convolution filter (named as “reduction layer”) is followed by comparably expensive 3×3 and 5×5 filters. All the feature maps extracted from the filters are integrated at the end of the inception module. Since object regions change considerably in sizes according to scales and perspective views, the inception module is designed to reflect diverse scales of objects by incorporating multi-size filters. A filter with small kernel has the role of detecting small object regions while a larger filter contributes to not only detecting large object regions but also effectively suppressing false positive regions that have similar properties as the targeting object.

We modify the inception module for image-to-image tasks by adding a larger kernel (7×7) and excluding the pool projection layer as shown in Figure 3 (b), where the original inception module is shown in Figure 3 (a) for com-

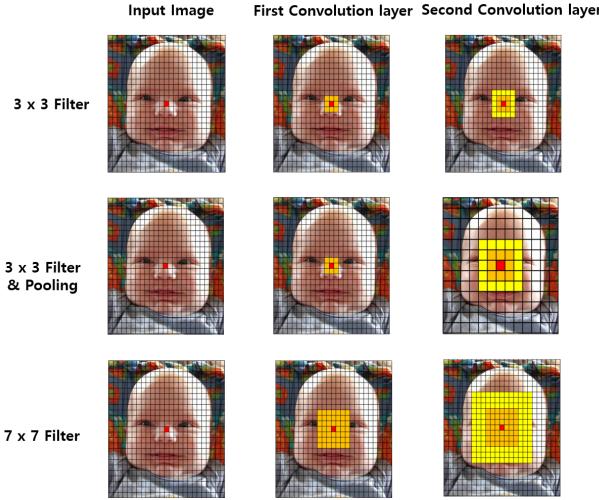


Figure 4. Example of the range of input pixels according to the filter size and pooling layer. It compares the receptive field when the network consists of 3×3 kernel (first row), 3×3 kernel with pooling (second row) and 7×7 kernel (third row). First column is the input, second is the receptive area at the first layer, and the third shows the receptive field after the second layer with the same size kernels or pooling.

parison. The original inception module [40] collects information from a wide range of images due to the overall pooling layers in the network, *i.e.*, the pooling layers enlarge receptive fields by reducing the size of feature dimension. However, since we remove the combination of pooling and fully connected layers in order not to decimate the feature map, we need to include a larger filter for keeping the receptive field large. As a specific example for this, Figure 4 shows how many pixels are involved in detecting a certain pixel according to filter size and the existence of the pooling layer. The first row of Figure 4 shows that the receptive range becomes 5×5 pixels at the second layer when we use 3×3 kernels consecutively. The second row shows that the receptive range becomes wider to 10×10 (at the image before decimation) when we add pooling to the above result. However, since we do not have pooling, we add a larger kernel (7×7) to keep the receptive range as shown in the last row of Figure 4.

In summary, the proposed module is composed of a set of filters with different sizes ($1 \times 1, 3 \times 3, 5 \times 5, 7 \times 7$) as illustrated in Figure 3, each of which consists of 8, 32, 16, and 8 filters respectively. We determine the number of filters in modified inception depending on the size of the filter in order to reduce the number of parameters.

3.2. Overall Architecture

The proposed method is applied to end-to-end mapping problems, which directly generate the overall output map. The network is composed of deep convolution layers and

proposed inception modules in order to exploit global evidence from the entire image. The number of overall parameters is 300K, which is much smaller than the same-depth inception networks adopted for classification problems.

3.2.1 Input and Output Design for the Proposed Architecture

As stated previously, we apply the proposed architecture to three problems: skin detection, semantic segmentation and compression artifacts reduction. In the case of semantic segmentation, since the images in the dataset have the same size, we just put the overall image as the input. In the case of compression artifacts reduction, we set the input and output as 37×37 image patches. For the skin detection problem, the overall image should also be the input because we wish to capture the wide range features rather than just color and texture of small patches. However, since the images in the dataset have various sizes and aspect ratios, we design the input and output as follows:

Input for the training: Decimate the image to a fixed size such that smaller side (horizontal/vertical) is reduced to 50, and stride the 50×50 image to other (larger side) direction.

Label image for the training: Decimated and stridden ground truth binary map in the same way as the input.

Input for inference: Image that is decimated such that smaller side (horizontal/vertical) is reduced to 50.

Final output: Probability or binary maps are obtained from the network, and they are interpolated to the original size.

3.2.2 Architecture Details for the Skin Detection Problem

We explain the details only for the skin detection problem, because other problems use almost the same architecture. The proposed network for the skin detection consists of 4 convolution layers and 8 modified inception modules where each module is composed of 2 convolution layers, so that the depth of the network adds up to 20 layers as described in Table 1 where H and W are decimated height and width. It is noted that “ 3×3 reduction, 5×5 reduction, and 7×7 reduction” are the reduction layers for $3 \times 3, 5 \times 5$ and 7×7 respectively. The ReLU layer is used as an activation function which is defined as:

$$f(x) = \max(0, x) \quad (1)$$

which follows each convolution layer except for the last.

The proposed network can be divided into four parts. The first part consists of 3 convolution layers of $7 \times 7, 1 \times 1$, and 3×3 kernels. The second part is a set of modified inception modules which are described in section 3.1 where all modules are identical. The number of modified inception modules is set to 8 which will be discussed in section

Table 1. Proposed Network-in-Network Architecture based on Modified Inception

type	Kernel Size	Output size	Depth	# 1 × 1	# 3 × 3 Reduce	# 3 × 3	# 5 × 5 Reduce	# 5 × 5	# 7 × 7 Reduce	# 7 × 7	Params
Convolution 1	7 × 7	H × W × 64	1								9K
Convolution 2	3 × 3	H × W × 64	2		64	64					41K
Inception 1		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 2		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 3		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 4		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 5		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 6		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 7		H × W × 64	2	8	32	32	16	16	8	8	23K
Inception 8		H × W × 64	2	8	32	32	16	16	8	8	23K
Convolution 3	5 × 5	H × W × 1	1								16K
Euclidean		H × W × 1	1								

4.2. Then, one convolution layer is used to integrate all the feature maps in a single channel which is a skin map produced by the proposed method. At last, the loss function is included for the training phase.

3.2.3 Loss function

Given training images and ground truth $\{\mathbf{X}_i, \mathbf{Y}_i\}_{i=1}^N$, our goal is to learn the parameters that minimize the loss function which is defined as Euclidean loss:

$$\mathbf{L}(\Theta) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{Y}_i - F(\mathbf{X}_i; \Theta)\|^2, \quad (2)$$

where N is the number of training images and Θ is the parameters of the network. convolutional network with the Euclidean loss function generally employs low learning rate because it easily fails to converge. As low learning rate needs much time for convergence, we employ an adaptive momentum approach [22].

4. Experiments

For validating the effectiveness of the proposed architecture to image-to-image problems, we select some of typical such problems as skin detection, semantic segmentation, and compression artifacts reduction.

4.1. Experiment Setup

Our network is implemented with the Caffe tool box [15], and GTX 980 with 4GB memory is employed both for training and testing. The input images are subtracted by the average intensity to alleviate vanishing and exploding gradient problem in the training phase [24], and all the initial parameters for the convolution layers are set to zeros. There are some hyper parameters for the proposed network which are decided empirically. We set the learning

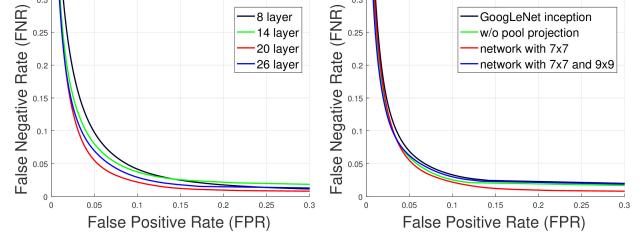


Figure 5. Quantitative comparison of proposed method with variations in network structure: ROC curves depth variation (left), filter set variation (right).

parameters as: initial learning rate = 0.001, weight decay = 0.0002. The depth of our network is set to 20 convolution layers, which is discussed in section 4.2.

4.2. Application to Skin Detection

We adopt ECU dataset [30] for training and validation, because it is the largest reliable set for the skin detection. It contains various ethnic groups, illumination variation, and complex non-skin but skin-colored regions/objects which often occlude skin regions. The dataset consists of 4,000 images, half of which are exploited for training and other 2,000 images are used for evaluation. We adopt the precision recall (PR) and Receiver Operating Characteristic (ROC) curves in evaluating the performance. Moreover, the quality of binarized skin probability map is evaluated with four metrics: *Accuracy*, *Precision*, *Recall*, and *F-measure* in a fixed threshold which are determined to make the maximum F-measure of the dataset.

We conduct two experiments to determine the optimal network structure for skin detection. First we evaluate the performance according to network depth: 8, 14, 20, and 26, where the number of filters and their sizes are all the same. We show the comparison results with ROC curve in Figure 5

at left the column. It can be seen that the performance increases as the depth grows up to 20, but the performance is lowered when 26. More fine variation in depth (not shown here) shows that performance saturates from the depth of 20. We think that 20 layers are enough, because this number can cover entire region as the receptive field when the input image size is 50×50 as stated previously. Hence, we set the depth of layers to 20 in the rest of skin detection experiments.

Second, we show the effectiveness of filter composition of network module (number of kernels and sizes at each layer). We conduct experiments with four different variations. The first module is composed of 1×1 , 3×3 , 5×5 , and pool projection which are the same composition as GoogLeNet, so we denote it as *GoogLeNet inception*. The second module is composed of 1×1 , 3×3 , 5×5 which removed pool projection layer in *GoogLeNet inception*, so we denote it as *w/o pool projection* and the third consists of 1×1 , 3×3 , 5×5 , and 7×7 in addition, so we denote it as *network with 7×7* . The final filter set consists of 1×1 , 3×3 , 5×5 , 7×7 , and 9×9 filters, so it is denoted as *network with 7×7 and 9×9* . The comparison results with ROC curve is shown in Figure 5 at the right column. It can be seen that the *GoogLeNet inception* case has lower performance than the others. It means that pool projection does not help to make effective feature maps, because some of spatial information can be lost due to max pooling. Since it is observed that *network with 7×7* and *network with 9×9* have similar performance on ROC curves, we determine to use *network with 7×7* in the rest of experiments.

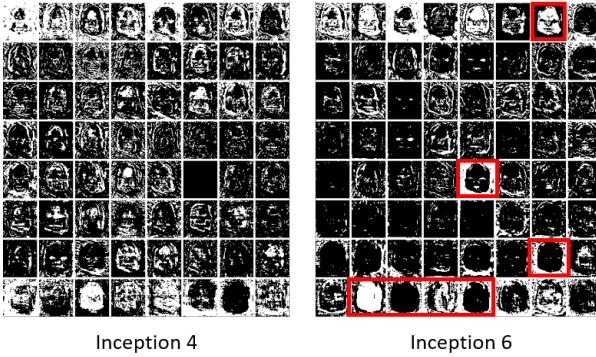


Figure 6. Visualization of feature maps.

We also visualize feature maps in Figure 6 to show how our inception module works. The first row shows 1×1 feature maps, second to fifth show 3×3 , sixth to seventh 5×5 , and eighth shows 7×7 feature maps. We can see in “Inception 4” feature maps that small filters usually make features from texture information, while large filters (7×7 filter) group skin and non skin region from shape information. At “Inception 6” feature maps, we can find that small filters can suppress small non skin region such as pupils and

mouth, and larger filters can find overall face as skin regions which were missed in small filters (red box). From these observation, adding a 7×7 filter helps to group skin region more effectively.

4.3. Comparison with Other Skin Detection Methods

We compare the proposed method using benchmark skin dataset with other methods: Bayesian [16], FPSD [18], DSPF [19], FSD [41], and LASD [14]. FPSD and DSPF are based on seed propagation over the graph representation of images. FSD is the mix of dynamic threshold and Gaussian model method. LASD is a luminance adapted color space method which optimizes least square error. We also test the proposed method only with gray channel input to show that our method effectively exploits structure information, and this structure is denoted as “Proposed (gray)”.

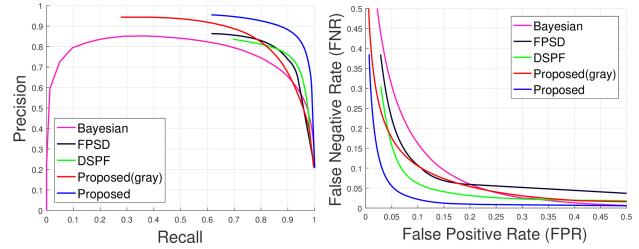


Figure 7. Comparison of PR (left) and ROC (right) curves on ECU dataset.

Table 2. Evaluation of skin detection on ECU dataset at peak F-measure.

Method	Accuracy	Precision	Recall	F-measure
Bayesian [16]	0.8910	0.7292	0.8220	0.7728
FPSD [18]	0.9106	0.7948	0.8534	0.8231
DSPF [19]	0.9190	0.7713	0.8864	0.8249
Proposed (gray)	0.9276	0.8137	0.8087	0.8112
Proposed	0.9562	0.8720	0.9122	0.8917

We conduct test for two datasets: ECU [30] (without trained images) and Pratheepan [49]. The Pratheepan is 32 facial images, each of which has a single face. First we compare the proposed method and other methods for ECU dataset which are plotted in Figure 7. Our method outperforms other methods by a large margin in terms of PR and ROC curves. We can also find that using a single gray channel image is also slightly better than Bayesian method which employ color information. It shows that our approach uses shape of human parts (body, hands, and faces) as well as color information.

We also evaluate the quality of binary map obtained by thresholding the above probability map. The threshold is set as to maximize F-measure which is differently selected

at each method. Table 2 shows this on ECU dataset, and our method has the best performance.

Subjective comparison is presented in Figure 9. It shows that our method robustly detects the skins of various races, rejects skin-like background region (non-skin region), and works robustly to illumination variations. Specifically, at the first row of Figure 9, the illumination variation on the baby's face is very severe, which lowered the performance of other methods, while our method works robustly as it uses color and shape information effectively as illustrated in Figure 6. It is also shown that other methods often fail to suppress skin-like background clutter (row 2,3, and 4), whereas our method can overcome these difficulties very well.

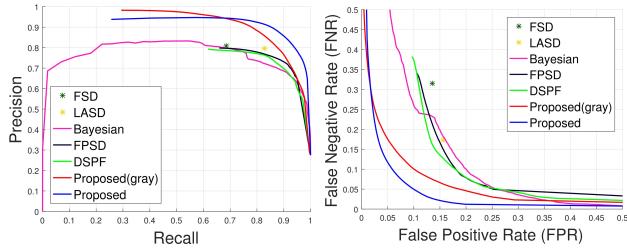


Figure 8. Comparison of PR (left) and ROC (right) curves on Pratheepan dataset.

Table 3. Evaluation of skin detection on Prateepan dataset at peak F-measure.

Method	Accuracy	Precision	Recall	F-measure
Bayesian [16]	0.8237	0.6881	0.8972	0.7788
FSD [41]	0.8255	0.8077	0.6851	0.7414
LASD [14]	0.8361	0.7954	0.8275	0.8111
FPSD [18]	0.8419	0.7837	0.8991	0.8070
DSPF [19]	0.8521	0.7543	0.8436	0.7964
Proposed (gray)	0.9211	0.8296	0.8379	0.8337
Proposed	0.9483	0.9003	0.8912	0.8957

Comparison on Pratheepan dataset is shown in Figure 8, where it can be seen that the proposed method has the best performance on both PR and ROC curves. Moreover, it has much better performance than others even in the case of using only gray channel image. Table 3 also shows that proposed method yields good results in terms of many performance measures. Finally, the subjective comparison for Pratheepan dataset is shown in Figure 10.

4.4. Application to Semantic Segmentation

We also apply the proposed network to semantic segmentation problem, where the only difference from the above skin detection network is the loss function. Specifically, we use softmax as a loss function and the other parameters (depth, number of filters, learning rate, etc) are kept the same as the skin detection work. We train and eval-



Figure 9. Subjective comparison of skin detection methods on ECU dataset: (from left to right) input, Bayesian, DSPF, FPSD, FSD, LASD, proposed (gray), proposed, and ground truth.



Figure 10. Subjective comparison of other methods on Prateepan dataset: (from left to right) input, Bayesian, DSPF, FPSD, FSD, LASD, proposed (gray), proposed, and ground truth.

uate proposed architecture using CamVid [3] set which is a 11 class set (mainly road, building, sky, and car). We train our network employing 367 training images and test with 233 RGB images at 360×480 resolution. It is compared with FCN [28] and SegNet [1] which are the state of the art semantic segmentation convolutional networks. We adopt three commonly employed evaluation measure: *Accuracy* for global pixel-wise accuracy, *Class Mean* for class average accuracy, and *I/U* is the mean of intersection over union.

Table 4. Evaluation of semantic segmentation on CamVid dataset.

Method	Accuracy	Class Mean	I/U
FCN [28]	0.839	0.556	0.450
SegNet [1]	0.842	0.565	0.477
Proposed	0.855	0.651	0.517

Quantitative results in Table 4 show that the proposed method yields better performance than the others in terms of all measure metrics. Visualization of output is presented in Figure 12. Proposed method classifies road and car very well, but it has limitations when the trees occlude buildings.

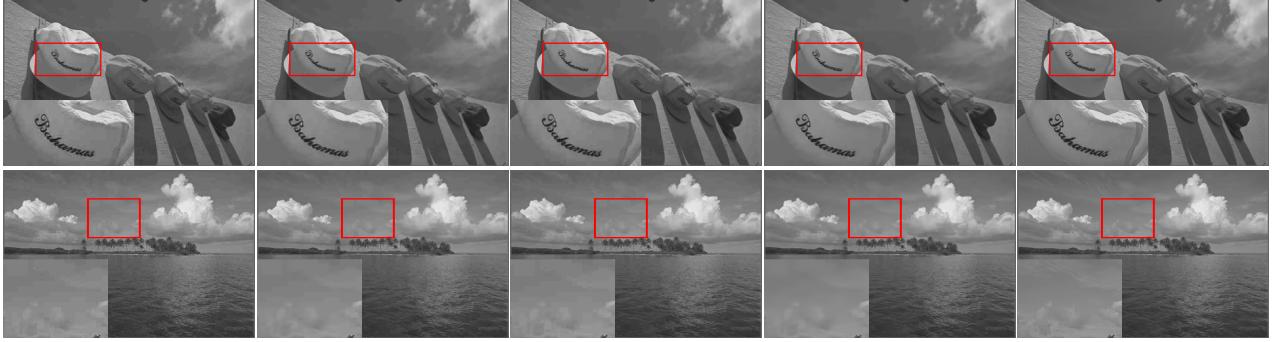


Figure 11. Subjective comparison of the proposed method with others: (from left to right) JPEG, AR-CNN, RAR-CNN, proposed, and original.

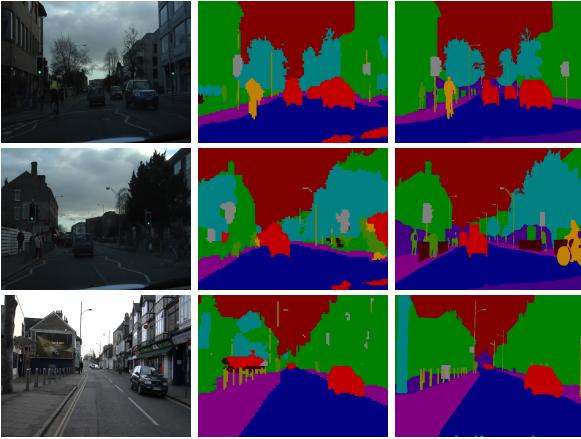


Figure 12. Visualization on CamVid dataset: (from left to right) input, proposed, and ground truth.

4.5. Application to Compression Artifacts Reduction

Our network is also applied to reducing the compression artifacts, especially the compression noise in JPEG images. The number of training images is 400 and test dataset is 5 classical images and LIVE1 which contains 29 images. We compare the proposed algorithm to SA-DCT [10] and convolutional network such as AR-CNN [7] and RAR-CNN [39]. SA-DCT is deblocking oriented method at frequency domain which is regarded as the state-of-the-art method before using a convolutional network. AR-CNN consist of four convolution layers and RAR-CNN is composed of four convolution layer which is trained with residual learning. Quantitative results evaluated on 5 classical images and LIVE1 dataset are presented in Table 5 and 6, which show that the proposed method yields the best performance for the compressed images at the quality factor of 10 and 20 in terms of PSNR and SSIM. Subjective results are shown in Figure 11, where enlarged image can be found in the supplementary file.

Table 5. Image reconstruction quality on 5 classical validation images for JPEG quality 10 and 20.

Methods	Q = 10		Q = 20	
	PSNR	SSIM	PSNR	SSIM
JPEG	27.82	0.780	30.12	0.854
SA-DCT [10]	28.88	0.807	30.92	0.8663
AR-CNN [7]	29.04	0.811	31.16	0.869
Proposed	29.34	0.820	31.51	0.876

Table 6. Image reconstruction quality on LIVE1 validation dataset for JPEG quality 10 and 20.

Methods	Q = 10		Q = 20	
	PSNR	SSIM	PSNR	SSIM
JPEG	27.77	0.791	30.07	0.868
SA-DCT [10]	28.65	0.809	30.81	0.878
AR-CNN [7]	28.98	0.822	31.29	0.887
RAR-CNN [39]	29.08	0.824	31.42	0.891
Proposed	29.25	0.828	31.60	0.894

5. Conclusions

We have proposed a network-in-network structure based on the inception module of GoogLeNet, which can be effectively used for the problems of reconstructing an image as the output. Specifically, the pool projection is removed from the original inception module in order not to decimate the features, and 7×7 convolutional network is added instead to keep the receptive field wide. The proposed architecture is applied to skin detection, semantic segmentation and compression artifacts reduction, and it is shown to yield competent results compared with the state-of-the-art methods. We believe that the proposed network widens the application areas of inception-like modules to the image-to-image problems, *i.e.*, it can be adopted to many other image restoration problems with some modification and tuning.

References

- [1] V. Badrinarayanan, A. Handa, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv preprint arXiv:1505.07293*, 2015. [1](#), [2](#), [7](#)
- [2] S. Bianco, F. Gasparini, and R. Schettini. Adaptive skin classification using face and body detection. *IEEE Transactions on Image Processing*, 24(12):4756–4765, 2015. [2](#)
- [3] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009. [7](#)
- [4] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *Proceedings of the European Conference on Computer Vision*, pages 44–57, 2008. [2](#)
- [5] D. A. Brown, I. Craw, and J. Lewthwaite. A som based approach to skin detection with application in real time systems. In *Proceedings of the British Machine Vision Conference*, pages 51.1–51.10, 2001. [2](#)
- [6] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in Neural Information Processing Systems*, pages 2843–2851. 2012. [2](#)
- [7] C. Dong, Y. Deng, C. Change Loy, and X. Tang. Compression artifacts reduction by a deep convolutional network. In *IEEE International Conference on Computer Vision (ICCV)*, pages 576–584, 2015. [2](#), [3](#), [8](#)
- [8] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, 2016. [1](#), [3](#)
- [9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013. [2](#)
- [10] A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. *IEEE Transactions on Image Processing*, 16(5):1395–1411, 2007. [2](#), [8](#)
- [11] H. Francke, J. Ruiz-del Solar, and R. Verschae. Real-time hand gesture detection and recognition using boosted classifiers and active learning. In *Pacific Rim Conference on Advances in Image and Video Technology*, pages 533–547, 2007. [2](#)
- [12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. [1](#)
- [13] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):696–706, 2002. [2](#)
- [14] I. Hwang, S. H. Lee, B. Min, and N. I. Cho. Luminance adapted skin color modeling for the robust detection of skin areas. In *IEEE International Conference on Image Processing*, pages 2622–2625, 2013. [6](#), [7](#)
- [15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678, 2014. [5](#)
- [16] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002. [2](#), [6](#), [7](#)
- [17] P. Kakumanu, S. Makrigiannis, and N. Bourbakis. A survey of skin-color modeling and detection methods. *Pattern Recognition*, 40(3):1106–1122, 2007. [2](#)
- [18] M. Kawulok. Fast propagation-based skin regions segmentation in color images. In *IEEE International Conference on Workshops Automatic Face and Gesture Recognition*, pages 1–7, 2013. [6](#), [7](#)
- [19] M. Kawulok, J. Kawulok, and J. Nalepa. Spatial-based skin detection using discriminative skin-presence features. *Pattern Recognition Letters*, 41:3–13, 2014. [6](#), [7](#)
- [20] R. Khan, A. Hanbury, and J. Stoettinger. Skin detection: A random forest approach. In *IEEE International Conference on Image Processing*, pages 4613–4616, 2010. [2](#)
- [21] J. Kim, J. Kwon Lee, and K. Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1516–1519, 2016. [3](#)
- [22] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. [5](#)
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012. [1](#), [2](#)
- [24] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. 2012. [5](#)
- [25] J. Y. Lee and S. I. Yoo. An elliptical boundary model for skin color detection. In *International Conference on Imaging Science, Systems, and Technology*, 2002. [2](#)
- [26] W.-H. Liao and Y.-H. Chi. Estimation of skin color range using achromatic features. In *International Conference on Intelligent Systems Design and Applications*, volume 2, pages 493–497, 2008. [2](#)
- [27] N. Liu and J. Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 678–686, 2016. [1](#)
- [28] 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*, pages 3431–3440, 2015. [1](#), [2](#), [7](#)
- [29] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1520–1528, 2015. [1](#), [2](#)
- [30] S. L. Phung, A. Bouzerdoum, and D. Chai. Skin segmentation using color pixel classification: analysis and comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):148–154, 2005. [5](#), [6](#)

- [31] S. L. Phung, D. Chai, and A. Bouzerdoum. A universal and robust human skin color model using neural networks. In *Proceedings of the IEEE International Joint Conference on Neural Network.*, volume 4, pages 2844–2849, 2001. 2
- [32] P. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *International Conference on Machine Learning*, pages 82–90, 2014. 2
- [33] H. C. Reeve III and J. S. Lim. Reduction of blocking effects in image coding. *Optical Engineering*, 23(1):230134–230134, 1984. 2
- [34] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2016. 2
- [35] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 2
- [36] L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):862–877, 2004. 2
- [37] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 2
- [38] P. Sturges, K. Alahari, L. Ladicky, and P. H. Torr. Combining appearance and structure from motion features for road scene understanding. In *Proceedings of the British Machine Vision Conference*, 2009. 2
- [39] P. Svoboda, M. Hradis, D. Barina, and P. Zemcik. Compression artifacts removal using convolutional neural networks. *arXiv preprint arXiv:1605.00366*, 2016. 2, 8
- [40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 1, 2, 3, 4
- [41] W. R. Tan, C. S. Chan, P. Yogarajah, and J. Condell. A fusion approach for efficient human skin detection. *IEEE Transactions on Industrial Informatics*, 8(1):138–147, 2012. 6, 7
- [42] V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In *Graphicon*, volume 3, pages 85–92, 2003. 2
- [43] C. Wang, J. Zhou, and S. Liu. Adaptive non-local means filter for image deblocking. *Signal Processing: Image Communication*, 28(5):522–530, 2013. 2
- [44] L. Wang, H. Lu, X. Ruan, and M.-H. Yang. Deep networks for saliency detection via local estimation and global search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3183–3192, 2015. 1
- [45] X. Wang, D. Fouhey, and A. Gupta. Designing deep networks for surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 539–547, 2015. 1
- [46] L. Xu, C. Lu, Y. Xu, and J. Jia. Image smoothing via L0 gradient minimization. *ACM Transactions on Graphics*, 30(6):174:1–174:12, 2011. 2
- [47] J. Yang, W. Lu, and A. Waibel. Skin-color modeling and adaptation. In *Proceedings of the Asian Conference on Computer Vision*, pages 687–694, 1998. 2
- [48] M.-H. Yang and N. Ahuja. Gaussian mixture model for human skin color and its applications in image and video databases. In *SPIE: Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 458–466, 1998. 2
- [49] P. Yogarajah, J. Condell, K. Curran, A. Cheddad, and P. McKeown. A dynamic threshold approach for skin segmentation in color images. In *IEEE International Conference on Image Processing*, pages 2225–2228, 2010. 6
- [50] K. Yu, C. Dong, C. C. Loy, and X. Tang. Deep convolution networks for compression artifacts reduction. *arXiv preprint arXiv:1608.02778*, 2016. 2
- [51] B. Zafarifar, E. Bellers, and P. de With. Application and evaluation of texture-adaptive skin detection in tv image enhancement. In *IEEE International Conference on Consumer Electronics*, pages 88–91, 2013. 2
- [52] Z. Zhang, H. Gunes, and M. Piccardi. Head detection for video surveillance based on categorical hair and skin colour models. In *IEEE International Conference on Image Processing*, pages 1137–1140, 2009. 2
- [53] R. Zhao, W. Ouyang, H. Li, and X. Wang. Saliency detection by multi-context deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1265–1274, 2015. 1