2D Iterative MAP Detection: Principles and Applications in Image Restoration

Daniel KEKRT, Tomáš LUKEŠ, Miloš KLÍMA, Karel FLIEGEL

Dept. of Radioelectronics, Czech Technical University in Prague, Technická 2, 166 27 Prague 6, Czech Republic

{kekrtd1, lukestom, klima, fliegek}@fel.cvut.cz

Abstract. *The paper provides a theoretical framework for the two-dimensional iterative maximum a posteriori detection. This generalization is based on the concept of detection algorithms BCJR and SOVA, i.e., the classical (one-dimensional) iterative detectors used in telecommunication applications. We generalize the one-dimensional detection problem considering the spatial ISI kernel as a two-dimensional finite state machine (2D FSM) representing a network of the spatially concatenated elements. The cellular structure topology defines the design of the 2D Iterative decoding network, where each cell is a general combination-marginalization statistical element (SISO module) exchanging discrete probability density functions (information metrics) with neighboring cells. In this paper, we statistically analyse the performance of various topologies with respect to their application in the field of image restoration. The iterative detection algorithm was applied on the task of binarization of images taken from a CCD camera. The reconstruction includes suppression of the defocus caused by the lens, CCD sensor noise suppression and interpolation (demosaicing). The simulations prove that the algorithm provides satisfactory results even in the case of an input image that is under-sampled due to the Bayer mask.*

Keywords

Iterative detection, 2D iterative decoding netwoks, maximum a posteriori probability criterion, defocus suppression, deconvolution, denoising, de-mosaicing, binary image restoration, image processing.

1. Introduction

The iterative detection based on the criterion of maximum a posteriori probability (MAP) is broadly used within the Turbo code detection [10]-[13]. The principle of Turbo decoder is closely related to the concept of the Viterbi algorithm [18]. Specifically, it includes the forward-backward recurrent algorithm (FBA-BCJR [16]), or the Viterbi algorithm with soft output (SOVA [17]) for code detection with binary input data. These systems work with one-dimensional signals where the independent variable represents time. The basic idea of two-dimensional iterative data detection was presented in [9]. Thiennviboon et al. applied iterative detection to digital image half-toning [7], [8]. The efficiency of the iterative detection has been shown on simulations with black and white defocused images [1], [3] and motion blurred images [2], where Kekrt et al. also introduced new variants and concatenations of network topologies.

The aim of the paper is to provide a theoretical framework for the iterative detection in two-dimensional form with emphasis on applications in the field of image processing. We extended the layered topology and provide a comparison of the existing topologies with respect to their applicability for image reconstruction. The input data are considered to be two-dimensional and they are coded by a spatial kernel. The spatial kernel introduces a general inter-symbol interference (ISI) to the signal. The spatial ISI channel can be seen as a two-dimensional finite state machine (2D FSM) and it can be decomposed into a horizontal-vertical network of simple combinational logic (cells). There are several possible ways how a coding network can be arranged. The topology of the coding network is further a template for an iterative decoding network where each cell is created by a general combination-marginalization statistical element or a SISO module. The goal of such a detector is to reconstruct the original (input) data, thus the inter-symbol interference and additive noise are suppressed. The detection process is given by a sequential activation of all cells of the network in a predefined order. Within this process, the cells are exchanging soft information (the discrete probability density) between each other.

In the first part of the paper, the theoretical framework of the two-dimensional iterative detection is described. Further various topologies of detection network are described including their activation schedules. An application for the reconstruction of black and white images is shown using selected topologies. The algorithm performs deconvolution, noise suppression and interpolation (de-mosaicing) of the binary image. The performance of the network topologies is statistically analysed by Monte Carlo method using random data with different levels of noise and blur. At the end, computational complexity and implementation issues are discussed.

2. General System Model

We assume a two-dimensional system (Fig. 1). The matrix **D** contains discrete values of input data. In the ISI channel, generally two-dimensional, inter-symbol interference is introduced into the input data. The output of the ISI channel is denoted by the matrix **Q**. The elements of the matrix **Q** are in discrete values. They are further referred to as code symbols. The **Q** matrix is modified in a front-end channel block defined by the function $f_s(.)$. This block provides an unambiguous mapping $\mathbf{S} = f_s(\mathbf{Q})$ from code symbols to the signal that is transmitted to the transmission channel. The front-end channel block transformation function $f_s(.)$ is memory-less and often linear. In the transmission channel, the signal **S** is corrupted by a signal interference **W**. This stochastic process can be described as an additive uncorrelated noise. In the model, we assume additive white Gaussian noise (AWGN). The corrupted signal **R** enters into the detector, which is composed from a network front-end (a gateway) and an iterative decoding network (IDN). The IDN works with probability densities instead of isolated values. Therefore, there is a gateway in front of the IDN that calculates these probability densities (a posteriori metrics) based on the obtained realization **R** and a known noise distribution in the transmission channel. The metrics are further denoted by $\mathcal{S}(\check{\xi})$ because it may refer to probability or its version transformed by an unambiguous mapping. Index F marks metrics considered as a posteriori ("forward") metrics. Index B marks a priori ("backward") metrics. An argument of each soft metric consists of a testing estimator $\check{\xi}$. It represents a possible value from a range of values of a given random variable and it is used to address a given metric.

2.1 Deterministic ISI Channel

The two-dimensional ISI channel can be defined by the convolution:

$$\begin{aligned} q[k,l] &= f(\mathcal{G},\mathcal{N}[k,l]) \\ &= \sum_{k',l' \in \mathcal{L}(\mathcal{G})} g[k',l']d[k+k',l+l'] \end{aligned} \tag{1}$$

where $\mathcal{G} = \{g[k',l']\}_{k',l' \in \mathcal{L}(\mathcal{G})}$ is a convolution kernel of the channel, $\mathcal{L}(\mathcal{G})$ denotes the cardinality of the kernel and $\mathcal{N}[k,l] = \{d[k+k',l+l']\}_{k',l' \in \mathcal{L}(\mathcal{G})}$ is the convolution region of input data $d[k,l] \in \mathcal{A}_d$ where k and l are spatial indexes in both dimensions of the image space. The input data are assumed to be binary ($\mathcal{A}_d = \{0,1\}$). The output of the channel $q[k,l]$ represents data interference within the region $\mathcal{N}[k,l]$ and it depends on the kernel size and type. Under the assumption of discrete input data, $q[k,l] \in \mathcal{A}_q$ is also discrete in values. All possible values that may result from the convolution create an alphabet $\mathcal{A}_q(\mathcal{G})$.

2.2 Cellular Models of ISI Channels

The convolution (1) is discrete operation due to the discrete inputs and it can be realized by a two-dimensional network of mutually connected elements. The network forms the finite state machine (2D FSM). These elements are shown in Fig. 3a. Let us assume that they have a general set of inputs and outputs: $\mathcal{V}_{I|O}[\ell] \in \{\mathcal{V}^{(i)}_{I|O}[\ell]\}_{i \in \mathcal{L}(\mathcal{V}_{I|O}[\ell])}$ with a discrete range of values $\mathcal{V}^{(i)}_{I|O}[\ell]$ and cardinality $\mathcal{L}(\mathcal{V}_{I|O}[\ell])$. Values of all inputs and outputs of one element create a set

$$\mathcal{N}_{IO} = \bigcup_{\ell} \mathcal{V}_{I|O}[\ell] \tag{2}$$

and input values of one element are denoted as

$$\mathcal{N} = \bigcup_{\ell} \mathcal{V}_{I}[\ell]. \tag{3}$$

The output values of the element

$$\{\mathcal{V}_O[\ell]\}_\ell = f(\mathcal{G},\mathcal{N}) \tag{4}$$

are a function of the kernel $\mathcal{G}$ and the inputs $\mathcal{N}$.

Firstly, the elements of the network can be understood as associated combinatorial logic that forms a Mealy type machine. Therefore, Fig. 3a contains more output ports although the output of the ISI channel (1) is only one. These "extra" outputs can be seen as state variables. They also represent input states of the neighboring elements. Secondly, the elements create a combinatorial logic where the state variables are not present. Then the element will have one output and the cardinality $\mathrm{card}(\mathcal{N}_{IO}) = \mathrm{card}(\mathcal{N}) + 1$.

There are three basic network topologies that are able to perform the convolution (1). These topologies are called: fixed topology, variable topology and layered topology. The fixed topology (FT) is based on a connection of combination elements in horizontal and vertical direction [11]. An extended version (EFT) has also diagonal connections [1]. In both cases, the topology does

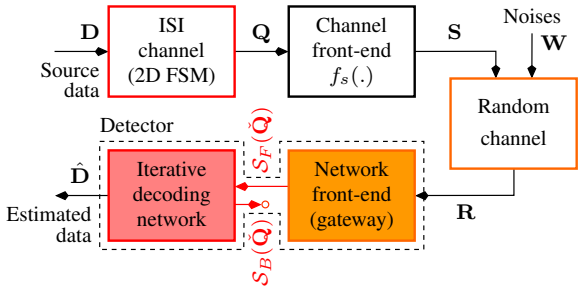
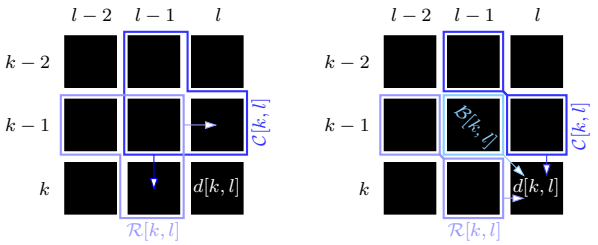


Fig. 1. Block diagram of a general two-dimensional transmission system.



(a) Fixed topology. (b) Extended fixed topology.

Fig. 2. The shapes of state variables $\mathcal{R}[k,l]$, $\mathcal{C}[k,l]$ and $\mathcal{B}[k,l]$ for a general reduced ISI channel $\mathcal{G}^{(\mathrm{R})}_{3\times3}$.

not depend on the shape of the kernel $\mathcal{G}$ and $\mathcal{N}_{IO}[k,l] = \{\mathcal{R}[k,l], \mathcal{C}[k,l], \mathcal{B}[k,l], d[k,l], \mathcal{R}[k,l+1], \mathcal{C}[k+1,l], \mathcal{B}[k+1,l+1], q[k,l]\}$. The first half of the set is given by the inputs and the second half are the outputs. The elements are exchanging state variables $\mathcal{R}$, $\mathcal{C}$, and $\mathcal{B}$ in vertical, horizontal and diagonal direction. The state variables contain a part of the convolution region $\mathcal{N}[k,l]$ and may be mutually disjoint (but it is not required). The initial states $\mathcal{R}[k,l]$, $\mathcal{C}[k,l]$, and $\mathcal{B}[k,l]$ together with the data $d[k,l]$ create the whole convolution region $\mathcal{N}[k,l]$. The input data are situated in the right down corner of the region. The final states $\mathcal{R}[k,l+1], \mathcal{C}[k+1,l], \mathcal{B}[k+1,l+1] \subset \mathcal{N}[k,l]$, transmitted to the neighboring elements, are given by a shift of information in the indicated direction (Fig. 2a). The state variables have to be placed and shifted in such a way that the network meets the condition of causality. Fig. 2b shows an example of the state variables for the reduced 3×3 kernel (i.e. a kernel with zero values in the corners).

The variable topology (VT) [11] represents a different approach to the solution of (1). The VT network is dependent on the shape of the kernel. It contains two types of combination elements (so-called broadcasters and receivers) and generally higher number of interconnections. The processing is carried out in two levels. On the lower level, the input data are copied into the state variables $c[k,l+l',\ell] = d[k,l]$ by broadcasters. The broadcasters (with the input-output set $\mathcal{N}_{IO}^{(\mathrm{B})}[k,l] = d[k,l] \cup \{c[k,l,\ell]\}$) create all mutually overlapped convolution regions $\mathcal{N}[k,l] = \{c[k+k',l+l',\ell]\}$. Consequently, these regions are combined into the outputs using elements on the higher level (receivers). The set of input-output variables of the receivers is given as $\mathcal{N}_{IO}[k,l] = \mathcal{N}[k,l] \cup q[k,l]$.

The FT and VT topologies are able to model the convolution kernel of an arbitrary shape. On the contrary, the layered topology can be applied only if the convolution (1) is separable into two orthogonal directions. The following condition must hold true for this type of kernels $\mathcal{G} = \mathcal{G}^{(\mathrm{V})} \times \mathcal{G}^{(\mathrm{H})}$, where $\times$ denotes the Cartesian product. Separable kernel always has a square or rectangular shape and the convolution (1) can be decomposed into a pair of one-dimensional convolutions $c[k,l] = f_H(\mathcal{G}^{(\mathrm{H})}, \mathcal{N}^{(\mathrm{H})}[k,l]) = \sum_{l' \in \mathcal{L}(\mathcal{G}^{(\mathrm{H})})} g_H[l'] d[k,l+l']$ and $q[k,l] = f_V(\mathcal{G}^{(\mathrm{V})}, \mathcal{N}^{(\mathrm{V})}[k,l]) = \sum_{k' \in \mathcal{L}(\mathcal{G}^{(\mathrm{V})})} g_V[k'] c[k+k',l]$, where $\mathcal{N}^{(\mathrm{H})}[k,l] = \{d[k,l+l']\}_{l' \in \mathcal{L}(\mathcal{G}^{(\mathrm{H})})}$ and $\mathcal{N}^{(\mathrm{V})}[k,l] = \{c[k+k',l]\}_{k' \in \mathcal{L}(\mathcal{G}^{(\mathrm{V})})}$ are one-dimensional convolution regions. Although the applicability of the layered topology is limited, there are several advantages, which we will discuss later in the paper.

Both layers of the model can be implemented using two layers with variable topology or two layers with fixed topology. In the latter case, inputs and outputs of the lower layer create a set $\mathcal{N}_{IO}^{(\mathrm{H})}[k,l] = \{\mathcal{R}[k,l], d[k,l], \mathcal{R}[k,l+1], c[k,l]\}$ and for the upper layer $\mathcal{N}_{IO}^{(\mathrm{V})}[k,l] = \{\mathcal{C}[k,l], c[k,l], \mathcal{C}[k+1,l], q[k,l]\}$.

2.3 Random IECS-ML Channel

For the signal $\mathbf{S}$ transmitted through a memory-less (ML) channel with independent eliminated channel states (IECS- ML), we may write

$$\mathbf{R} = \mathbf{S} + \mathbf{W} \tag{5}$$

where $\mathbf{W}$ represents an additive noise. Values of noise are assumed to be spatially invariant in the space $[k,l]$. The input-output relation of the random channel may be rewritten in the following scalar form

$$r[k,l] = s[k,l] + w[k,l] \tag{6}$$

where each noise value $w[k,l]$ has the same probability distribution $p_w(\xi)$. The probability distribution is considered to be the Gaussian with non-zero mean value.

3. 2D MAP Detection

The two-dimensional maximum a posteriori (MAP) detection is in principle similar as in the one-dimensional case. More about the one-dimensional MAP detection can be found in [11, 12, 13]. We will focus on a transition of the problem from the single-stage (optimal) MAP detection to the iterative (suboptimal) MAP detection, which can be realized in practice.

3.1 Single-Stage (Optimal) MAP Detection

The optimal 2D MAP detector is based on the following MAP criterion

$$\hat{d}[k,l] = \arg \operatorname*{Ⓜ}_{\check{d}[k,l]} \left(\operatorname*{Ⓜ}_{\check{\mathbf{D}}:\check{d}[k,l]} \mathcal{S}(\mathbf{R}, \check{\mathbf{D}}) \right) \tag{7}$$

where $\hat{d}[k,l]$ is the output estimate of the reconstructed signal at the $[k,l]$ position, $\check{d}[k,l]$ denotes a testing estimator (a possible value of reconstructed signal), $\check{\mathbf{D}} : \check{d}[k,l]$ is a set of all possible realizations, which contains at the given position $\check{d}[k,l]$ and Ⓜ as well as Ⓜ represents general marginalization operators. Marginalization process runs through joint soft measures of the detector $\mathcal{S}(\mathbf{R}, \check{\mathbf{D}})$ and quantifies the degree of truth of the statement $\check{\mathbf{D}} = \mathbf{D}$ when a realization $\mathbf{R}$ was received. The amount of metrics $\mathcal{S}(\mathbf{R}, \check{\mathbf{D}})$ grows exponentially with the size of $\check{\mathbf{D}}$. The basis of this growth is the cardinality of the input alphabet. Therefore, the MAP criterion can not be implemented directly. In order to make the problem feasible, it is necessary to factorize the criterion into smaller parts. It is possible due to the assumption that the noise values $w[k,l]$ in the matrix $\mathbf{W}$ are not correlated. It stands that $p_{\mathbf{W}}(\Xi) = \prod_{k,l} p_w(\xi)$ and the metric of the detector

$$\begin{aligned} \mathcal{S}(\mathbf{R}, \check{\mathbf{D}}) &= \left(\operatorname*{©}_{k,l} \mathcal{S}_F(r[k,l] | \check{\mathcal{N}}[k,l]) \right) © \left(\operatorname*{©}_{k,l} \mathcal{S}_B(\check{d}[k,l]) \right) \\ &= \mathcal{S}_F(\mathbf{R} | \check{\mathbf{D}}) © \mathcal{S}_B(\check{\mathbf{D}}) \end{aligned} \tag{8}$$

can be decomposed into individual, overlapping, convolution regions $\check{\mathcal{N}}[k,l]$, where $\mathcal{S}_F(r[k,l] | \check{\mathcal{N}}[k,l])$ are a posteriori

metrics and $\mathcal{S}_B(\check{d}[k,l])$ are a priori metrics of data. We assume that the data are spatially invariant $[k,l]$. If the data has uniform distribution, the component $\mathcal{S}_B(\check{\mathbf{D}})$ in (8) can be neglected and the MAP criterion becomes the Maximum Likelihood (ML) criterion. All factorized metrics in (8) are combined into the joint metric of the detector using a general combination operator ©. Combination and marginalization operators can be divided according to an operation domain and a detection technique. It is the symbol detection (SyD) or the sequential detection (PgD). The symbol detection tries to minimize the error detection of a symbol $d[k,l]$. The sequential (or also page) detection tries to minimize the error detection of all data $\mathbf{D}$ as a whole. Tab. 1 shows an overview of the operators where $\mathcal{M}(.) = -\ln \mathcal{P}(.)$ and $\min^*(x,y) = \min(x,y) - \ln(1+e^{-|x-y|})$. Md-PgD version is numerically the most effective because it has the lowest numerical complexity of the operators.

Domain	Detect.	Ⓜ	Ⓜ	©	©	$©^{-1}$
Probability	Page	max	max	Π	$\times$	$\div$
Probability	Symbol	max	Σ	Π	$\times$	$\div$
Metric	Page	min	min	Σ	+	$-$
Metric	Symbol	min	$\min^*$	Σ	+	$-$

Tab. 1. Summary of combination and marginalization operators.

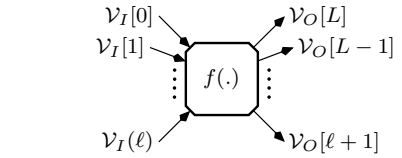
3.2 Iterative (Suboptimal) MAP Detection

It has been shown that the MAP criterion can be for ML-IECS channel decomposed into individual convolution regions $\check{\mathcal{N}}[k,l]$. This leads to the iterative detection, which basically reduces the complexity of the single-stage MAP criterion from the size of the input data $\mathbf{D}$ to the size of the kernel $\mathcal{G}$. The iterative detector (decoding network) is a system of mutually connected elements within one of the topologies described in Sec. 2.2. Each element of the network decodes the particular convolution region. The minimum number of network elements is equal to the size of the input data $\mathbf{D}$. Combination-marginalization elements are shown in Fig. 3b. Hereafter, we will refer to them as SISO (Soft-In Soft-Out) modules. In essence, they represent statistical soft inversion in relation to the elements of the ISI channel model (Fig. 3a). Each port of the SISO module is bidirectional and carries the whole discrete soft metric (probability density) quantifying the reliability of all possible values of the variable (not only one value of the variable as in the case of the ISI channel model).

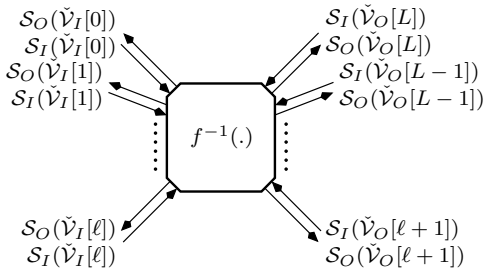
During the detection process, each module initially take over the input metrics $\mathcal{S}_I$ from its neighbors and combine them into partial joint a posteriori metrics $\mathcal{S}(\check{\mathcal{N}}_{lO})$. The joint a posteriori metrics $\mathcal{S}(\check{\mathcal{N}}_{lO})$ denote reliabilities of all possible arrangements (realizations) of the particular convolution region. Subsequently, the SISO module marginalize the set of joint metrics into outputs. This operation is called an activation of the SISO module. SISO modules are activated progressively according to the so-called activation schedule. The activation schedule has to have a logic that corresponds to the information flow (state and auxiliary variables) within the ISI channel model. If the activation schedule is selected inappropriately (against the flow of the state variables), the time of the convergence to the correct solution (given in the number of iterations) increases. The term iteration will be understood as a time interval in which each SISO module is activated at least once. Each iteration is concluded with the exchange of metrics

$$\begin{aligned} \mathcal{S}_O(\check{\mathcal{V}}_{l|O}[\ell]) &\rightarrow \mathcal{S}_I(\check{\mathcal{V}}_{l|O}[\ell]) \\ \mathcal{S}_I(\check{\mathcal{V}}_{l|O}[\ell]) &\leftarrow \mathcal{S}_O(\check{\mathcal{V}}_{l|O}[\ell]) \end{aligned} \tag{9}$$

between SISO modules.



(a) General processing element.



(b) Soft-In Soft-Out module.

Fig. 3. A general processing element and its soft inversion (SISO module).

At the beginning of the detection process, the input metrics $\mathcal{S}_I$ are set according to the a priori knowledge about the variables which they represent. If the a priori knowledge is unknown, these metrics are set uniformly (each possible value of the testing estimator has the same probability). In this case, only forward (input) metrics $\mathcal{S}_F$ from the front-end of the iterative detector will be non-uniform.

The iterative detection is suboptimal. The final estimate can be regarded as optimal after infinite number of iterations. In most cases, however, the convergence of the network is fast (in order of units of iterations). When the amount of information exchanged between SISO modules stabilizes, it does not change much with the following iterations. The system reaches a state when the metrics are "locked" in the feedback (iterative loops). The detection process is then terminated and the final hard decision is made

$$\hat{\mathcal{V}}_{l|O}[\ell] = \arg \underset{\check{\mathcal{V}}_{l|O}[\ell]}{Ⓜ} \left(\mathcal{S}_I(\check{\mathcal{V}}_{l|O}[\ell]) © \mathcal{S}_O(\check{\mathcal{V}}_{l|O}[\ell]) \right) \tag{10}$$

by a combination of the input and output metrics on the particular port of the SISO module (10).

4. Iterative Decoding Networks

In Sec. 3.2, we have described the base of the iterative detection, cellular structure of the two-dimensional iterative MAP detector and its functional blocks. The activation of the SISO module will be further discussed in detail together with the integration of the SISO module within different topologies (described in Sec. 2.2).

4.1 Network Elements - SISO Modules

The activation of the SISO module is composed of two steps - combination and marginalization. The combination merges input (a priori and a posteriori) soft metrics $\{\mathcal{S}_I(\check{\mathcal{V}}_{I|O}[\ell])\}_\ell$ of all input-output variables $\mathcal{V}_{I|O}[\ell]$ into a joint a posteriori metric $\mathcal{S}(\check{\mathcal{N}}_{IO})$ using a particular combination operator ©. This operation is given as

$$\mathcal{S}(\check{\mathcal{N}}_{IO}) = \mathop{\copyright}_{\check{\mathcal{V}}_{I|O}[\ell]\in\check{\mathcal{N}}_{IO}} \mathcal{S}_I(\check{\mathcal{V}}_{I|O}[\ell]) \tag{11}$$

where the condition $\check{\mathcal{V}}_{I|O}[\ell] \in \check{\mathcal{N}}_{IO}$ arises from the combination table of the SISO module. A posteriori metrics $\mathcal{S}(\check{\mathcal{N}}_{IO}) \in \{\mathcal{S}(\mathcal{N}_{IO}^{(i)})\}_i$ correspond to the likelihood of individual realizations of the set of inputs and outputs $\mathcal{N}_{IO}^{(i)}$. The alphabet of all realizations of the set $\mathcal{N}_{IO}$ is denoted as $\mathcal{A}_{IO} = \{\mathcal{N}_{IO}^{(i)}\}_i$. For example, a 3×3 channel with a binary input has $\mathrm{card}(\mathcal{A}_{IO}) = 512$. Therefore, a set of 512 joint metrics is obtained in the first step of the activation of the SISO module.

$i \mapsto \mathcal{A}_{IO}$	$i_{c[0]} \mapsto \mathcal{A}_c$	$i_{c[2]}$	$i_{c[1]}$	$i_q \mapsto \mathcal{A}_q$
0	0	0	0	0
1	1	0	0	1
2	0	1	0	3
3	1	1	0	4
4	0	0	1	1
5	1	0	1	2
6	0	1	1	4
7	1	1	1	5

Tab. 2. Table of combinations controlling the example simple SISO module.

Joint metrics are marginalized into the output (a posteriori) metrics $\{\mathcal{S}_O(\check{\mathcal{V}}_{I|O}[\ell])\}_\ell$ by the marginalization operator Ⓜ. This second step of the activation is defined as

$$\mathcal{S}_O(\check{\mathcal{V}}_{I|O}[\ell]) = \left(\mathop{\text{Ⓜ}}_{\check{\mathcal{N}}_{IO}:\check{\mathcal{V}}_{I|O}[\ell]} \mathcal{S}(\check{\mathcal{N}}_{IO})\right) \copyright^{-1} \mathcal{S}_I(\check{\mathcal{V}}_{I|O}[\ell]) \tag{12}$$

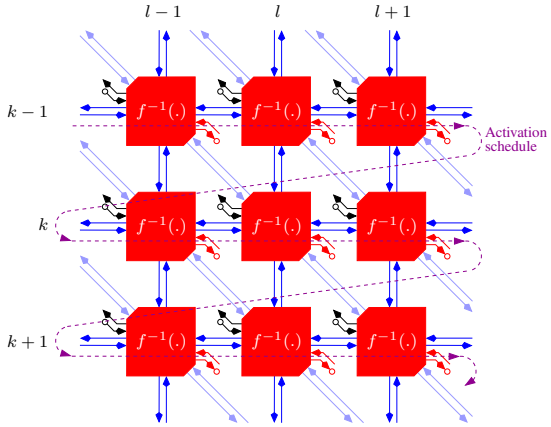
where $\check{\mathcal{N}}_{IO} : \check{\mathcal{V}}_{I|O}[\ell] \subset \mathcal{A}_{IO}$ denotes a set of all inputs and outputs that contain the testing estimator $\check{\mathcal{V}}_{I|O}[\ell]$. Subsequently, the SISO module sends the output metrics into the neighboring modules.

The mechanism is explained in the following example. We assume a simple one-dimensional kernel $\mathcal{G}^{(H)} = \{\ g'\quad g\quad g'\ \}$. The output of such a channel is $q = g'(c[0]+c[1])+gc[2]$, where $c[0]$, $c[1]$, and $c[2]$ are binary inputs. For testing estimators of inputs, outputs, and input-output sets, we may write $\check{c}[\ell] \in \mathcal{A}_c = \{0,1\}$, $\check{q} \in \mathcal{A}_q = \{0, g', 2g', g, g+g', g+2g'\}$ and $\check{\mathcal{N}}_{IO} = \{\check{c}[0], \check{c}[2], \check{c}[1], \check{q}\}$. Individual variables are associated into combinations (Tab. 2) where i are indexes of mappings into alphabet of the given variable. Each out of eight table rows represents one realization $\mathcal{N}_{IO}^{(i)}$ of the set $\check{\mathcal{N}}_{IO}$. Four densities are the input of the combination. Three two-element $\mathcal{S}_I(\check{c}[\ell])$ and one six-element $\mathcal{S}_I(\check{q})$. The combination of input metrics gives eight joint metrics $\mathcal{S}(\check{\mathcal{N}}_{IO})$. The joint metrics are marginalized into three two-element metrics $\mathcal{S}_O(\check{c}[\ell])$ and one six-element metric $\mathcal{S}_O(\check{q})$. An example of the first step of the activation process is calculation of the joint metric $\mathcal{S}(\check{\mathcal{N}}_{IO} = \{1,1,0,g+g'\}) = \mathcal{S}_I(\check{c}[0]=1)©\mathcal{S}_I(\check{c}[2]=1)©\mathcal{S}_I(\check{c}[1]=0)©\mathcal{S}_I(\check{q}=g+g')$. This combination is marked by blue squares in the table (Tab. 2). An example of the second step of the activation process can be the calculation of the output metric $\mathcal{S}_O(\check{c}[0]=0) = (\mathcal{S}(\check{\mathcal{N}}_{IO}=\{0,0,0,0\})Ⓜ\mathcal{S}(\check{\mathcal{N}}_{IO}=\{0,1,0,g\})Ⓜ\mathcal{S}(\check{\mathcal{N}}_{IO}=\{0,0,1,g'\})Ⓜ\mathcal{S}(\check{\mathcal{N}}_{IO}=\{0,1,1,g+g'\}))©^{-1}\mathcal{S}_I(\check{c}[0]=0)$. This marginalization is marked by red squares. Each other SISO module uses the same principle. A difference is only in input-output variables, their cardinality and combination table that controls the combination-marginalization process.

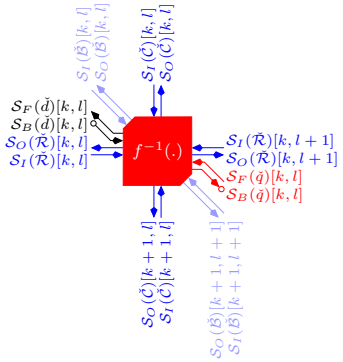
4.2 Fixed Topologies

Fig. 4a. shows the iterative decoding network with fixed topology. The topology does not depend on the shape of the kernel and is composed of the SISO modules depicted in Fig. 4b. Basic module connections are horizontal and vertical [11]. An extended version has also diagonal connections [1]. The activation schedule of the detector is in Fig. 4a (the purple dashed line). The activation can be performed row by row or column by column or in "zig-zag" manner. All these variants progress in the same direction as the flow of the state variables in the ISI channel model. Networks with fixed topology are also called as networks with the marginalization at the symbol block level. The convolution region is separated into blocks. The advantage of the fixed topology is high efficiency due to state variables that often have a significant overlap. Large amout of information has to be transmitted and stored between the modules.

Optimal separation of the convolution region into the state variables is difficult for some kernel types. Sometimes it is not even possible. In that case, the state variables have a higher cardinality and the memory requirements are raising. The extended variant with diagonal connections tries to suppress these problems. The state variable then allows higher variability for the convolution region decomposition. There are three state variables, so they can have lower cardinality and the memory requirements are reduced in exchange for slight degradation of the performance.



(a) The IDN topology with marked activation schedule.



(b) The IDN cell in the node $[k,l]$.

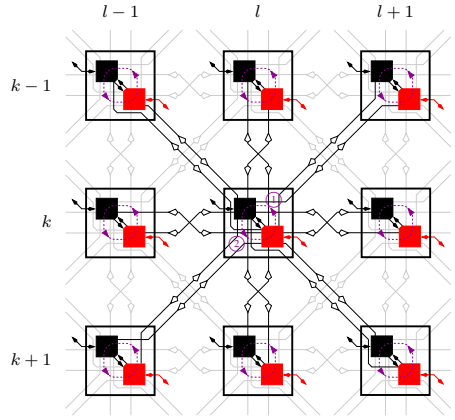
Fig. 4. Iterative decoding network marginalizing at the symbol block level.

4.3 Variable Topologies

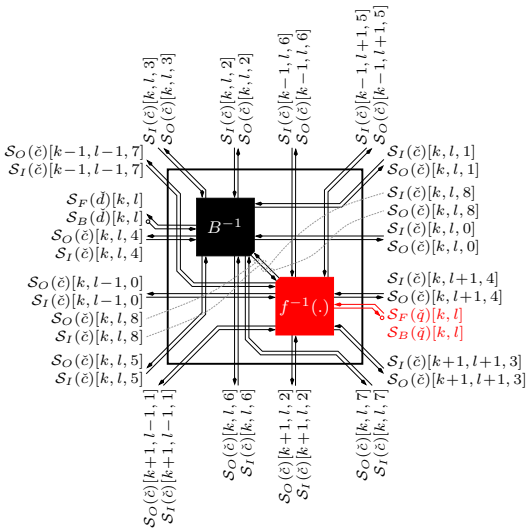
The variable topology can not be depicted in a general form because it depends on the shape of the kernel. We will explain the principle on an example of a general 3×3 kernel $\mathcal{G}_{3\times3}$. Fig. 5a shows the topology for this type of kernel. Each element of the structure (Fig. 5b) contains two SISO modules [3]. SISO modules (in red) perform the soft inversion of the convolution itself. It is followed by the soft inversion of broadcasters (in black). The broadcasters are in the reference channel model responsible for input data branching and preparation of individual convolution regions. Data branching is described in a combination table Tab. 3, which controls the soft inversions of broadcasters in the decoding network.

$i \mapsto \mathcal{A}_{IO}^{(\mathrm{B})}$	$i_{c[0]} \mapsto \mathcal{A}_d$	$\cdots$	$i_{c[8]}$	$i_d \mapsto \mathcal{A}_d$
0	0	$\cdots$	0	0
1	1	$\cdots$	1	1

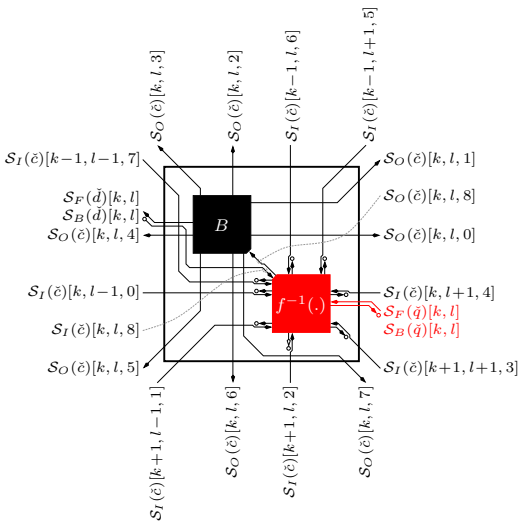
Tab. 3. Table of combinations controlling the soft inversion of binary broadcaster.



(a) The IDN topology with marked activation schedule.



(b) The IDN cell in the node $[k,l]$.



(c) The simplified IDN cell in the node $[k,l]$.

Fig. 5. Iterative decoding network marginalizing at the symbol level for a general ISI channel $\mathcal{G}_{3\times3}$.

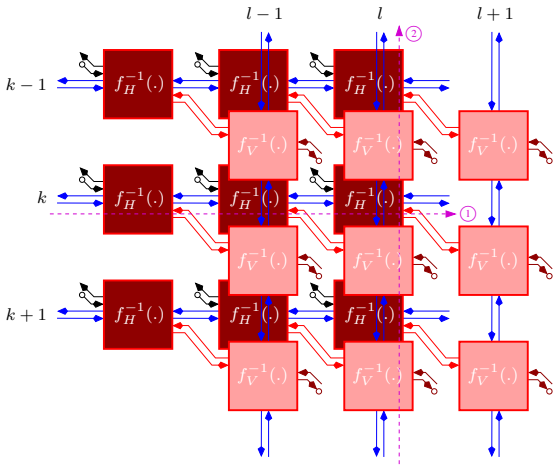
The activation schedule is parallel. It appears in purple in Fig. 5a. The activation takes place in all elements at once and consists of two steps. Firstly, metrics collected from the neighboring elements activate the soft inversion of the combinational logic. Secondly, soft inversions of broadcasters are performed followed by the distribution of output metrics into the neighboring elements. The whole process is repeated in the next iteration.

The convolution is separated on the lowest possible level (symbol level). The network is flexible and can be well adjusted even for kernels with uncommon shapes. The disadvantage is lower performance caused by maximal decomposition of the convolution region and a large amount of connections. On the other hand, memory requirements are low because cardinality of the metrics of the state variables is the smallest possible. The variable topology can be implemented in a simplified version. It significantly reduces computational demands. A simplified element [11] is shown in Fig. 5c. The simplified implementation is based on the assumption that the center of the kernel has the dominant coefficient. The output of the reference ISI channel model is then mostly influenced by the center of the convolution region. Therefore, the element contains only one SISO module, which performs one marginalization. Only the state variable in the center is marginalized. The output is one metric and it is further distributed into the neighboring elements by a common broadcaster. Connections in the topology are the same, but compared to the layout in Fig. 5a the distribution of metrics is performed only in one direction (away from broadcaster). In the simplified version, only one marginalization is carried out (instead of nine). It hugely reduces computational complexity. Additionally, the soft inversion of the broadcaster is not necessary. The amount of connections is half that because the flow of information is only one-way. Memory requirements are also lower, but the performance deteriorates and the network can not be used to solve a more complex task (as for instance de-mosaicing, which is explained in the next chapter).

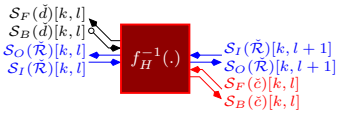
4.4 Layered Topologies

The layered topology represents a combination of the two previously mentioned topologies. It can be used only if it is possible to decompose the convolution kernel into two orthogonal directions (as mentioned in 2.2). If the two-dimensional convolution can be replaced by two consecutive one-dimensional convolutions, the reference channel model will have two layers. So as the iterative network has two layers. Each of these layers can be implemented using fixed or variable topology. The utilization of fixed topology leads to a detector where the lower layer is concatenated horizontally and the upper layer vertically (Fig. 6a). The SISO modules of the upper and lower level are shown in Fig. 6b and Fig. 6c, respectively.

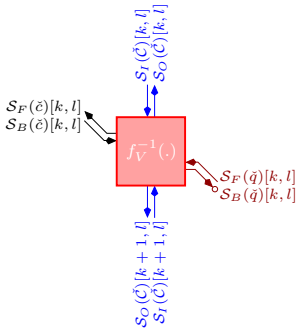
One iteration of the system is given by activation of the upper layer in a column by column manner. It is followed



(a) The IDN topology with marked activation schedule.



(b) The IDN cell in the node $[k,l]$ on the bottom layer.



(c) The IDN cell in the node $[k,l]$ on the top layer.

Fig. 6. Layered iterative decoding network marginalizing at the symbol block level.

by parallel activation of the lower level, which is done in a row by row manner. During the iteration process, the state metrics $\mathcal{S}_{I|O}(\check{\mathcal{R}})$ and $\mathcal{S}_{I|O}(\check{\mathcal{C}})$ on the individual layers and the "inter-layers" metrics $\mathcal{S}_{F|B}(\check{c})$ are precised. If the variable topology is used, we obtain the structure in Fig. 7 (for the deconvolution of a 3×3 channel). The iteration process is then analogical. The activation runs from the upper layer and it is followed by the activation of the lower layer.

The biggest advantage of the layered detector is the reduction of computational demands. Let us assume a decomposed kernel $\mathcal{G}_{3\times3}^{(\mathrm{HV})}$. In the case of the standard (non-simplified) variable topology and fixed topology, the cardinality of the SISO module combination table is $\mathrm{card}(\mathcal{A}_{IO}) = 2^9 = 512$. For the layered topology, the cardinality of the lower layer is $\mathrm{card}(\mathcal{A}_{IO}^{(\mathrm{H})}) = 2^3 = 8$ and on the upper layer $\mathrm{card}(\mathcal{A}_{IO}^{(\mathrm{V})}) = 6^3 = 216$, which is 224 in total. The computational demand is decreased but as a consequence the performance is degraded. Basically, the performance decreases

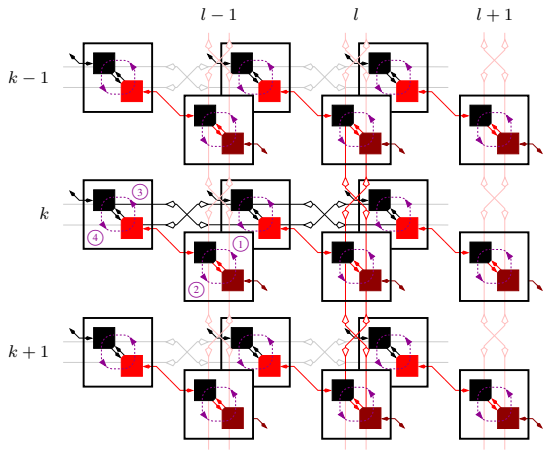


Fig. 7. Layered iterative decoding network marginalizing at the symbol level for a general decomposition-able ISI channel $\mathcal{G}_{3\times3}^{(\mathrm{HV})}$.

with the increasing number of marginalization operations. The layered topology contains one extra marginalization that produces "inter-layers" metrics.

5. Application in Image Restoration

The performance and capabilities of iterative detection will be demonstrated on a restoration of binary images obtained from a camera with CCD sensor. Thresholding that leads to a binary image is often an important first step of image analysis [19]. In many applications, images are in essence binary. They are blurred by the optical system and some noise is added during the process of image acquisition. The thresholding then can be seen as a reconstruction of the ideal image. The iterative decoding network reconstructs the desired image due to its deconvolution and noise suppression properties. The algorithm decides for each pixel whether its value will be black or white by maximizing the corresponding a posteriori probability that the pixel contains the useful signal (object in the foreground) or not (background noise etc.). We also demonstrate that if the image is undersampled, the iterative decoding network is able to calculate the missing information. It promises interesting applications in image interpolation.

5.1 System Model

The ISI channel in this application is the objective of the camera. The diffraction of light prevents exact convergence of the light rays to a single point at the image plane. A sharp point on the object is blurred into a finite-sized spot in the image which is described by the point spread function (PSF) of the objective. The diffraction limited PSF of an ideal lens with a circular aperture can be described by the Airy disk [21]. The PSF of a real lens of a common camera can be well approximated by Gaussian model [22],

$$\mathrm{PSF}_\Delta(x,y) = \frac{\Delta}{\pi} e^{-\Delta(x^2+y^2)} \tag{13}$$

where Δ determines the lobe width. Model does not consider the loss of light in the lens. Therefore $\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\mathrm{PSF}_\Delta(x,y)dxdy = 1$. Cells of the imaging sensor are assumed to be square-shaped, with no gaps between them and with a normalized length equal to 1. The amount of light that impacts a cell of the sensor is given by integration of the PSF over a particular square-shaped area. We obtain the Gaussian kernel

$$\{\mathcal{G}_{L\times L,\Delta}^{(\mathrm{Gauss})}\}_{k,l} = \int_l^{l+1}\int_k^{k+1} \mathrm{PSF}_\Delta\left(x-\frac{1}{2}, y-\frac{1}{2}\right) dxdy \tag{14}$$

where L is the size of the kernel, which determines a number of neighboring pixels affecting each other by blurring. Between the suppression of the main beam and lobe width of the PSF, there is a relation $\{\mathcal{G}_{L\times L,\Delta}^{(\mathrm{Gauss})}\}_{0,0} = \mathrm{erf}(\sqrt{\Delta}/2)^2$. The kernel size L and the lobe width Δ reflect the properties of the camera objective.

The camera sensor introduces random additive noise to the signal. The basic camera model takes into account three noise sources: thermal noise $\mathbf{W}_T$, readout noise $\mathbf{W}_R$ and quantization noise $\mathbf{W}_C$ of the A/D converter [14]. The signal at the output of the imaging sensor can be expressed as

$$\begin{aligned}\mathbf{R}_C &= \frac{N_e^{(\mathrm{ET})}\mathbf{Q} + \mathbf{W}_T + \mathbf{W}_R}{\Delta_C} \\ &= \mathbf{S} + \frac{\mathbf{W}_T + \mathbf{W}_R}{\Delta_C}\end{aligned} \tag{15}$$

where

$$\Delta_C = \left\lfloor 2^{-N_B} N_e^{(\mathrm{FWC})} \right\rfloor \tag{16}$$

denotes the number of electrons per quantization step of the converter and $N_e^{(\mathrm{ET})}$ is the number of electrons generated in the potential well at the maximum irradiation of the sensor ($\mathbf{Q} = \mathbf{1}$) during the exposure time T_E. We expect an exposure time that does not cause the saturation of the sensor so that $N_e^{(\mathrm{ET})} < N_e^{(\mathrm{FWC})}$ where $N_e^{(\mathrm{FWC})}$ is the full well capacity (FWC) of the sensor. The signal is quantized at the output of the sensor,

$$\begin{aligned}\mathbf{R} &= \mathbf{R}_C + \mathbf{W}_C \\ &= \lceil \mathbf{R}_C \rfloor\end{aligned} \tag{17}$$

using rounding $\lceil . \rfloor$ to the nearest integer value.

Gateway of detector processes the signal $\mathbf{R}$ using the knowledge of statistical properties of the noise. Noise distributions are known. Under standard conditions, the Poisson distribution of the thermal noise can be approximated by normal distribution with the mean value μ_T and standard deviation $\sqrt{\mu_T}$. The approximation holds true for values $\mu_T \geqq 40$ [14]. The combination of these two noise sources leads to the Gaussian additive noise with the probability density $p_w(\xi,\mu_w,\sigma_w) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-(\xi-\mu_w)^2/2\sigma_w^2)$. We integrate this continuous distribution over different quantization

steps from 0 up to the full well capacity. So we get discrete distribution,

$$\Pr_w^{(\mathrm{Cut})}(\mu,\sigma)[n,N] = \begin{cases} \frac{1}{2}\mathrm{erfc}\left(\frac{2\mu-1}{\sqrt{8}\sigma}\right), & n=0 \\ \frac{1}{2}\left(\mathrm{erf}\left(\frac{2(n-\mu)+1}{\sqrt{8}\sigma}\right)-\mathrm{erf}\left(\frac{2(n-\mu)-1}{\sqrt{8}\sigma}\right)\right), & 0<n<N \\ \frac{1}{2}\mathrm{erfc}\left(\frac{2(N-\mu)-1}{\sqrt{8}\sigma}\right), & n=N \end{cases} \quad (18)$$

where N will be substituted for the highest quantization level. Thus, we obtain the transformation function of the gateway,

$$\{\mathcal{P}_F(q^{(i)})[k,l]\}_i = \left\{ \Pr_w^{(\mathrm{Cut})}\left(\frac{\hat{\mu}_T + N_e^{(\mathrm{ET})} q^{(i)}}{\Delta_C}, \frac{\sqrt{\hat{\mu}_T}+\hat{\sigma}_R}{\Delta_C} \right) [r[k,l], 2^{N_B}-1] \right\}_i \quad (19)$$

where $\hat{\mu}_T$ is an estimate of the mean value of the thermal noise, $\hat{\sigma}_R$ is an estimate of the standard deviation of the readout noise, N_B is the number of bits of the A/D converter and $q^{(i)}$ are individual values from the alphabet $\mathcal{A}_q$.

5.2 Examples of Dichromatic Image Restoration with Perfect CSI Knowledge

Image binarization will be demonstrated using two types of convolution kernels. The first one is a 3×3 kernel with the transmission of the main ray 0.3.

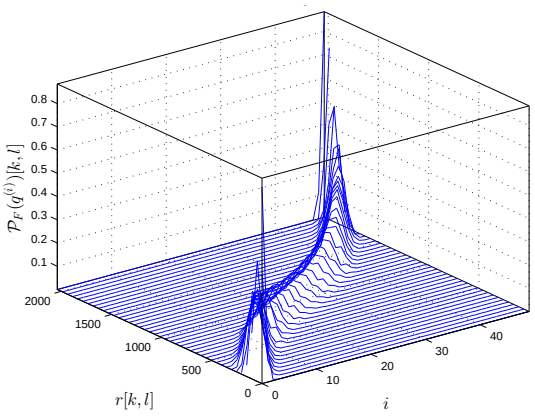
$$\mathcal{G}_{3\times3,1.13}^{(\mathrm{Gauss})} = \begin{Bmatrix} 0.046 & 0.117 & 0.046 \\ 0.117 & 0.3000 & 0.117 \\ 0.046 & 0.117 & 0.046 \end{Bmatrix} \quad (20)$$

In the second case, a 5×5 reduced kernel is utilized with suppressed side beams and the transmission of the main ray 0.2,

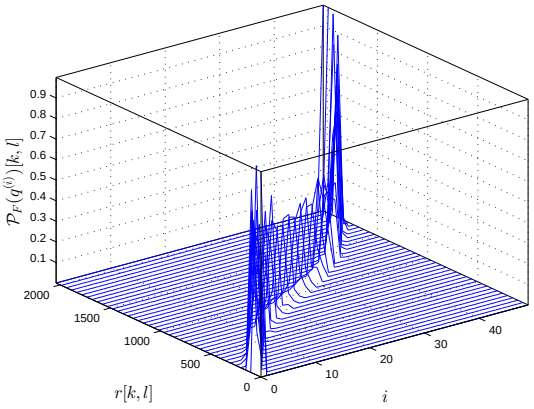
$$\mathcal{G}_{5\times5,0.705}^{(\mathrm{Gauss,R})} = \begin{Bmatrix} & & 0.016 & & \\ & 0.057 & 0.107 & 0.057 & \\ 0.016 & 0.107 & 0.2000 & 0.107 & 0.016 \\ & 0.057 & 0.107 & 0.057 & \\ & & 0.016 & & \end{Bmatrix}. \quad (21)$$

Let us assume that the properties of the optical system on the detection part are known and that the detector works with perfect channel state information (CSI) about the values of the kernel. In this simulation, we model the image acquisition by the sensor iXon3 885 (Andor Technology). Real properties of this sensor were used for the following parameters $N_e^{(\mathrm{FWC})} = 30\times10^3$ [e], $\Delta_C = 14$ [e], $N_B = 11$. For the chosen value $N_e^{(\mathrm{ET})} = 29\times10^3$ [e], the sensor is excited up to 96.5 %. We leave $\hat{\sigma}_R$ as an independent variable, which determines the standard deviation $\hat{\sigma} = \sqrt{\hat{\mu}_T} + \hat{\sigma}_R$ of the resulting Gaussian noise. In the simulations, we used $\hat{\mu}_T = 60$ [e].

The transfer function of the gateway corresponds to the properties of the sensor in combination with the first model of the convolution kernel (20) (Fig. 8). For the second model of the convolution kernel, the transfer function is similar with typical "s-shaped deflection".



(a) $\hat{\sigma} = 10^3$ [e].



(b) $\hat{\sigma} = 316$ [e].

Fig. 8. The examples of transfer functions of network gateway for Gaussian kernel 3×3, $\Delta = 1.13$ and various $\hat{\sigma}$.

Fig. 9 shows a simulated image of a QR code at the output of the A/D converter of a camera. Two cases are the outputs of a monochromatic camera and two are outputs from a camera with the Bayer mask (green filter pattern). Fig. 10 and Fig. 11 represent the reconstructed (binarized) output images together with their error rates after one ($I = 1$), three ($I = 3$) and five ($I = 5$) iterations. In the case of 3×3 kernel, the image was reconstructed by the decoding network with a marginalization at the pixel block level and SyD technique. We used marginalization at the pixel level and PgD technique for the 5×5 kernel. The results show that the error rate falls rapidly with an increasing iteration number. Already the 3rd iteration provides satisfactory results.

If the kernels are large, the level of noise should be lower or the detection may start to fail. It is caused by a relatively small bit-depth of the A/D converter. The quantization step Δ_C is then too coarse and the deconvolution task becomes ambiguous due to the larger cardinality of the al-

phabet $\mathcal{A}_q$. If the signal to noise ratio (SNR) is low and the convolution kernel large, it is good to use A/D converter with finer quantization (for instance 16 bit). If the detection task includes also de-mosaicing, each second metric $\mathcal{P}_F$ is unknown. The missing metrics are replaced by uniform metrics. It informs the detector that we have no information about the signal $r[k,l]$ in this place. The local SISO module relies fully on the information from its neighbors. Although it makes the detection highly ambiguous, the detector is able to reconstruct the signal but the noise level has to be lower compared to the sensing at full raster.

5.3 Performance Analyses

The objective performance evaluation was performed under MATLAB object oriented environment and carried out by measuring the bit error rate (BER) of the output data $\hat{\mathbf{D}}$. We have used the Monte Carlo method with a random input data $\mathbf{D}$ of size 64×64. All experiments were done with the 3×3 Gaussian convolution kernel with three different settings of the main coefficient $g_{0,0}$ (0.2, 0.3 and 0.5).

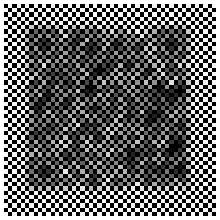
Each BER value is an average of 150 measurements. We utilized a logarithmic scale of the standard deviation of the Gaussian noises: 10000, 3162, 1000, 316, 100, 32, and 10. Fig. 12 depicts the separation of the state variables of the convolution region for the fixed topology network. The results of the analysis are shown in Fig. 13 and Fig. 14 (a monochromatic camera), Fig. 15 (a color camera with Bayer mask).

The fixed topology offers the best performance. It works with robust state variables, which usually are not disjunctive. Their disadvantage is higher computational demand than for other topologies and difficult applicability in a case of non-standard shaped kernels. The intersections of the state variables complicate the design of the SISO module implementation with tree-like combination-marginalization, which enables to decrease the computational complexity. The extended fixed topology with diagonal connections partially eliminates these disadvantages but it has lower performance compared to the simple fixed topology.

Variable topology provides low computational demands and high application flexibility. It can be well applied even for the reduced kernels where the fixed topology usually can not be used, because the data input $d[k,l]$ in the fixed topology is always situated in the right down corner of the convolution region. If the channel transfer is in this place equal to zero (and it is for the reduced kernel $\mathcal{G}^{(\mathrm{R})}_{3\times3}$), the input variable of the reference model does not affect the output $q[k,l]$, but only the following states $\mathcal{R}[k,l+1]$, $\mathcal{C}[k+1,l]$ a $\mathcal{B}[k+1,l+1]$. The output then depends only on the output states $\mathcal{R}[k,l]$, $\mathcal{C}[k,l]$ and $\mathcal{B}[k,l]$. Therefore, the cardinality of the combination table of the SISO modules is $\mathrm{card}(\mathcal{A}_{IO}) = 2^6 = 64$. It is twice as much as for the same kernel designed by the variable topology where the cardinality of the combination table is $\mathrm{card}(\mathcal{A}_{IO}) = 2^5 = 32$ due to the optimal coverage of the convolution region.

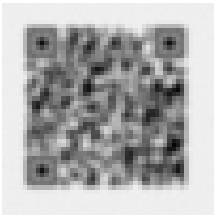
Original pattern **D**.

(a.1) Input pattern **R**. Monochromatic camera. Noise: $\hat{\sigma} = 1000$ [e].

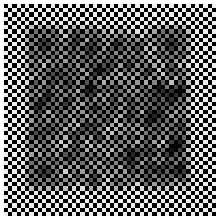


(a.2) Input pattern **R**. Color camera based on Bayer CFA. Noise: $\hat{\sigma} = 316$ [e].

(a) Lens model: Gaussian kernel 3×3, $\Delta = 1.13$.



(b.1) Input pattern **R**. Monochromatic camera. Noise: $\hat{\sigma} = 70$ [e].

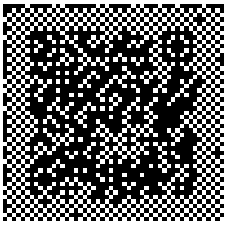


(b.2) Input pattern **R**. Color camera based on Bayer CFA. Noise: $\hat{\sigma} = 10$ [e].

(b)Lens model: reduced Gaussian kernel 5×5, $\Delta = 0.705$.

Fig. 9. The examples of pattern realizations.

(a.1) $I = 1$, BER $= 0.348$



(b.1) $I = 1$, BER $= 0.419$

(a.2) $I = 3$, BER $= 0.014$

(b.2) $I = 3$, BER $= 0.046$

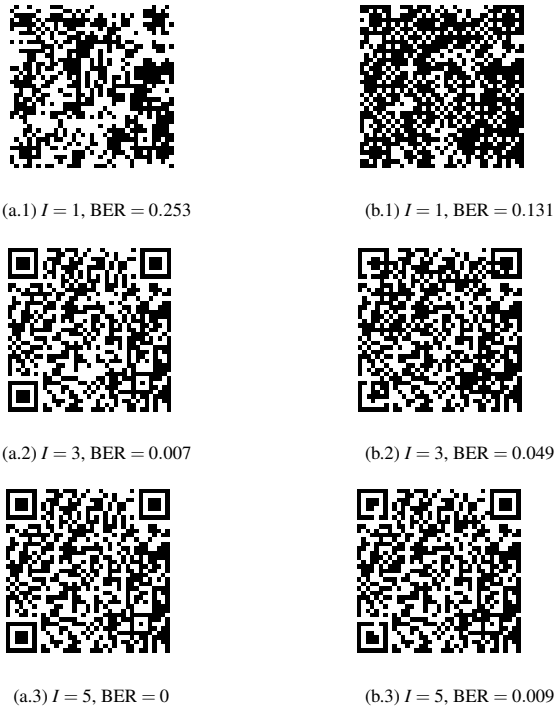
(a.3) $I = 5$, BER $= 0.006$

(b.3) $I = 5$, BER $= 0.007$

(a) Output patterns $\hat{\mathbf{D}}$. Monochromatic camera. Noise: $\hat{\sigma} = 1000$ [e].

(b) Output patterns $\hat{\mathbf{D}}$. Color camera based on Bayer CFA. Noise: $\hat{\sigma} = 316$ [e].

Fig. 10. The examples of QR code restoration by IDN marginalizing at the symbol block level and using Pd-SyD technique. Lens model: Gaussian kernel 3×3, $\Delta = 1.13$.



(a.1) $I = 1$, BER $= 0.253$ (b.1) $I = 1$, BER $= 0.131$

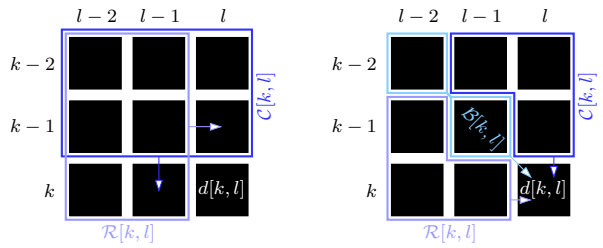
(a.2) $I = 3$, BER $= 0.007$ (b.2) $I = 3$, BER $= 0.049$

(a.3) $I = 5$, BER $= 0$ (b.3) $I = 5$, BER $= 0.009$

(a) Output patterns $\hat{\mathbf{D}}$. Monochromatic camera. Noise: $\hat{\sigma} = 70$ [e].

(b) Output patterns $\hat{\mathbf{D}}$. Color camera based on Bayer CFA. Noise: $\hat{\sigma} = 10$ [e].

Fig. 11. The examples of QR code restoration by IDN marginalizing at the symbol level and using Pd-PgD technique. Lens model: reduced Gaussian kernel 5×5, $\Delta = 0.705$.



(a) Fixed topology.

(b) Extended fixed topology.

Fig. 12. The shapes of state variables $\mathcal{R}[k,l]$, $\mathcal{C}[k,l]$ and $\mathcal{B}[k,l]$ for the ISI channel $\mathcal{G}^{(\text{Gauss})}_{3\times3,\Delta}$.

The layered topology is the most suitable if the key factor for a specific application is low computational complexity. The cardinality of the table of SISO modules is reduced at the upper level due to the "inter-level" marginalization. The computational demands are lower in exchange for slightly degraded performance. All discussed topologies can be applied to the input image with full grid. If the grid of the input image contains holes, all topologies that do not rely on the simplified IDN cell (Fig. 5c) can be used.

6. Conclusions and Future Work

The paper has shown the effectivity of the two-dimensional iterative detection when it is applied on image reconstruction problems. The simulations prove that the algorithm provides satisfactory results with low bit error rate already in a small number of iterations. Even if half of the input information is unknown (as in the de-mosaicing problem), the detection network is able to restore the data into the full resolution. Due to the deconvolution and noise suppression capabilities, the algorithm performs well even if the noise level is high.

The simulations show a promising potential of the two-dimensional iterative detection in the field of microscopy image analysis and particle detection.Various network topologies were described and tested. The fixed topology reached the best performance. Variable topology offers the best trade-off between the performance, computational demands and application flexibility. The layered topology has the lowest computational demands.

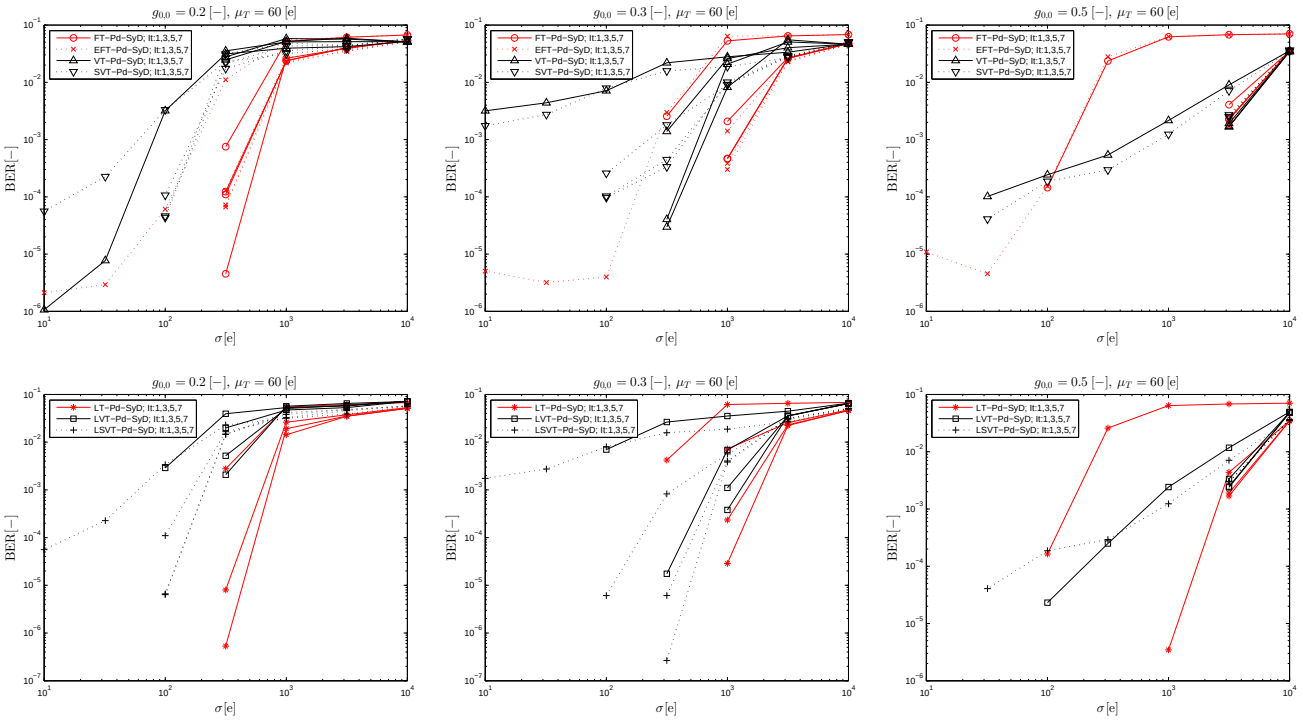
There are several open tasks for further improvement of the two-dimensional iterative MAP detection method. First of all, the reduction of complexity of the SISO modules. A possible solution can be a reduction of combination tables if the states with very low probabilities will not be taken into account. The cut of the transformation function along index i (Fig. 8) shows that values are equal to zero. All joint a posteriori probabilities that contains these values are then also equal to zero and it is not necessary to calculate them. It leads to significant complexity reduction. Another research task can be a synchronization of the network. In the real application, the kernel is not precisely known. Therefore, CSI at the detection site is not perfect. It can be measured during a calibration process with limited accuracy. For the optimal performance, the network should be synchronized within this degree of uncertainty. To solve the associated synchronization task, we probably could use a particular form of the expectation-maximization criteria or similar algorithm.

Acknowledgements

This work was partially supported by the COST IC1003 European Network on Quality of Experience in Multimedia Systems and Services - QUALINET, by the COST CZ LD12018 Modeling and verification of methods for Quality of Experience (QoE) assessment in multimedia systems - MOVERIQ and by the grant No. P102/10/1320 Research and modeling of advanced methods of image quality evaluation of the Czech Science Foundation.

References

[1] KEKRT, D., KLIMA, M., FLIEGEL, K. Restoration of dichromatic images gained from CCD/CMOS camera by iterative detection networks with fragmented marginalization at the symbol block level. In *Proceedings of SPIE 7444, Mathematics for Signal and Information Processing, 74441A*. San Diego (USA), 2009, p. 74441A-1 - 74441A-12.



(a) Main ray attenuation 0.2. (b) Main ray attenuation 0.3. (c) Main ray attenuation 0.5.

Fig. 13. The performance analyses of random patterns restoration by various IDNs using Pd-SyD technique. Patterns are captured with full resolution.

[2] KEKRT, D., KLIMA, M., FLIEGEL, K. The iterative detection network based suppression of the thermal noise and blurring due to object moving in black and white pictures shot by a camera with CCD/CMOS sensor. In *Proceedings of SPIE 7076, Optics and Photonics.* San Diego (USA), 2008, p. 70760M-1 - 70760M-9.

[3] KEKRT, D., KLIMA, M. PODGORNY, R. The iterative detection network suppression of defocusing and thermal noise in black and white pictures shot by a camera with CCD/CMOS sensor. In *Photonics Prague 2008, The 6th International Conference on Photonics, Devices and Systems, 71381Z.* Prague (Czech Republic), 2008, p. 71381Z-1 - 71381Z-7.

[4] KEKRT, D., KLIMA, M. The advanced noise model for an IDN based restoration of black and white pictures captured by a camera with CCD/CMOS sensor. In *Proceeding of the 42nd Annual IEEE International Carnahan Conference on Security Technology (ICCST).* Prague (Czech Republic), 2008, p. 126 - 130.

[5] KEKRT, D., KLIMA, M. A black & white picture reconstruction by iterative detection network in the image capturing system with CCD (CMOS) sensor. In *Proceedings of 50th International Symposium ELMAR*. Zadar (Croatia), 2008, p. 125 - 128.

[6] KEKRT, D., KLIMA, M. 2D iterative detection network based image restoration: Principles, applications and performance analysis. *Image Restoration - Recent Advances and Applications*. In Tech, 2012, chapter 14, p. 315 - 350. ISBN 978-953-51-0388-2.

[7] THIENNVIBOON, P., ORTEGA, A., CHUGG, K. Simplified grid message-passing algorithm with application to digital image halftoning. In *Proceedings of the International Conference on Image Processing, 2001*. Thessaloniki (Greece), 2001, vol. 2, p. 1061 - 1064.

[8] THIENNVIBOON, P., CHUGG, K. Model-based digital image halftoning using iterative reduced-complexity grid message-passing algorithm. In *Proceedings of SPIE 5008, Color Imaging: Processing, Hardcopy and Applications VIII.* Santa Clara (USA), 2003, p. 419 - 430.

[9] CHEN, X., CHUGG, K. Near-optimal page detection for two-dimensional ISI/AWGN channels using concatenated modeling and iterative detection. In *Proceedings of the International Conference on Communications.* Atlanta (USA), 1998, vol. 2, p. 952 - 956.

[10] POLAK, L., KRATOCHVIL, T. Analysis and simulation of the transmission distortions of the mobile digital television DVB-SH part 1: Terrestrial mode DVB-SH-A with OFDM. *Radioengineering*, vol. 21, no. 4, 2011, p. 952 - 960.

[11] CHUGG, K., ANASTASOPOULOS, A., CHEN, X. *Iterative Detection: Adaptivity, Complexity Reduction and Applications.* Kluwer Academic Publishers, 2001.

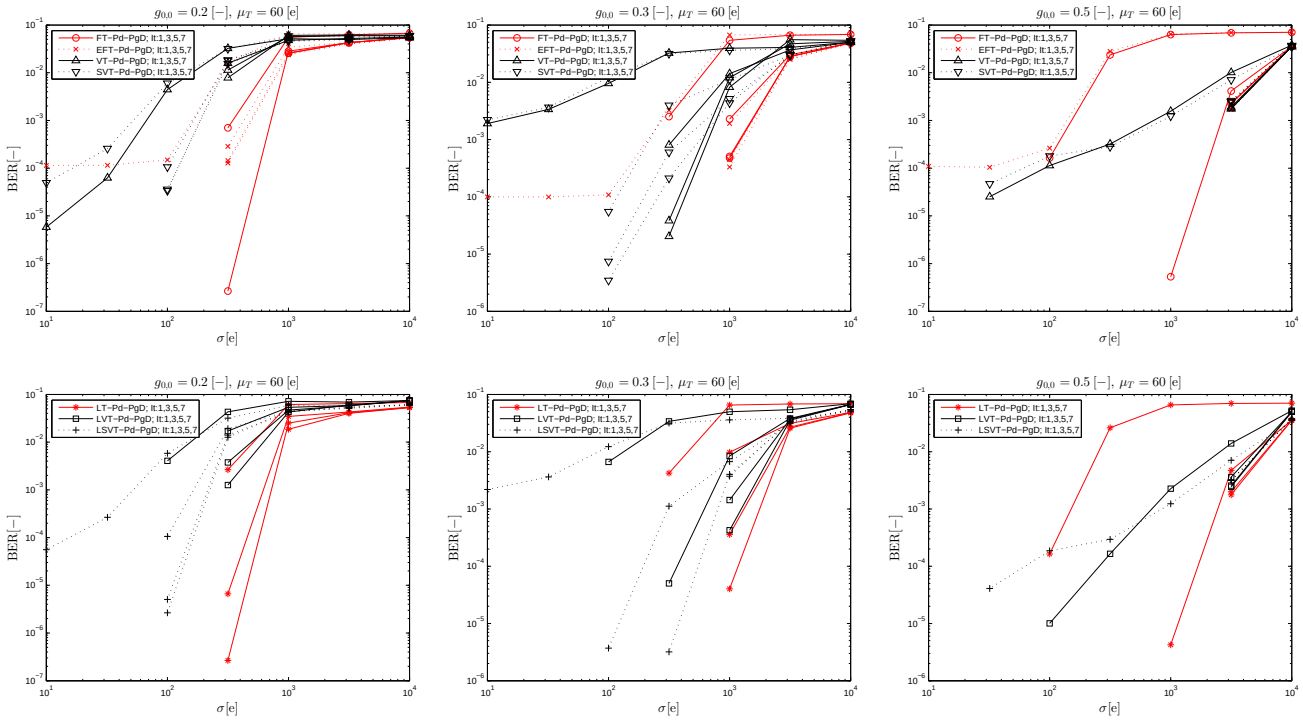
[12] VUCETIC, B., JUAN, J. *Space-Time Coding.* Kluwer Academic Publishers, 2003.

[13] VUCETIC, B., JUAN, J. *Turbo codes : Principles and Applications.* Kluwer Academic Publishers, 2000.

[14] VLIET, L. J., BODDEKE, F. R., SUDAR, D., YOUNG, I. T. Image detectors for digital image microscopy. In *Digital Image Analysis of Microbes; Imaging, Morphometry, Fluorometry and Motility Techniques and Applications, Modern Microbiological Methods.* Chichester (UK), 1998, p. 37 - 64.

[15] BERROU, C., GLAVIEUX, A., THITMAJSHIMA, P. Near Shannon limit errorcorrecting coding and decoding: Turbo-codes. In *Proceedings of the International Conference on Communications.* Geneva (Switzerland), 1993, p. 1064 - 1070.

[16] BAHL, L. R., COCKE., J., JELINEK, F., RAVIV, J. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Transactions on Information Theory*, 1974, vol. IT-20, p. 284 - 287.



(a) Main ray attenuation 0.2. (b) Main ray attenuation 0.3. (c) Main ray attenuation 0.5.

Fig. 14. The performance analyses of random patterns restoration by various IDNs using Pd-PgD technique. Patterns are captured with full resolution.

[17] HAGENAUER, J., HOEHER, P. A Viterbi algorithm with soft-decision outputs and its applications. In *Proceedings of IEEE Global Telecommunications Conference 1989.* Dallas (USA), 1989, p. 1680 - 1686.

[18] FORNEY, J. D. The Viterbi Algorithm. *Proceedings of the IEEE*, 1973, vol. 61, no. 3, p. 268 - 278.

[19] SANKUR, B., SEZGIN, M. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 2004, p. 146 - 165.

[20] PAVLOVIČOVÁ, J., ORAVEC, M., POLEC, J., KELEŠI, M., MOKOŠ, M. Error concealment using neural networks for block-based image coding. *Radioengineering*, vol. 15, no. 2, 2006, p. 30 - 36.

[21] GOODMAN, J. W. *Introduction to Fourier Optics.* New York (USA): McGraw-Hill, 1996.

[22] STALLINGA, S., RIEGER, B., Accuracy of the Gaussian point spread function model in 2D localization microscopy. *Opt. Express 18 (24)*, 2010, p. 24461 - 24476.

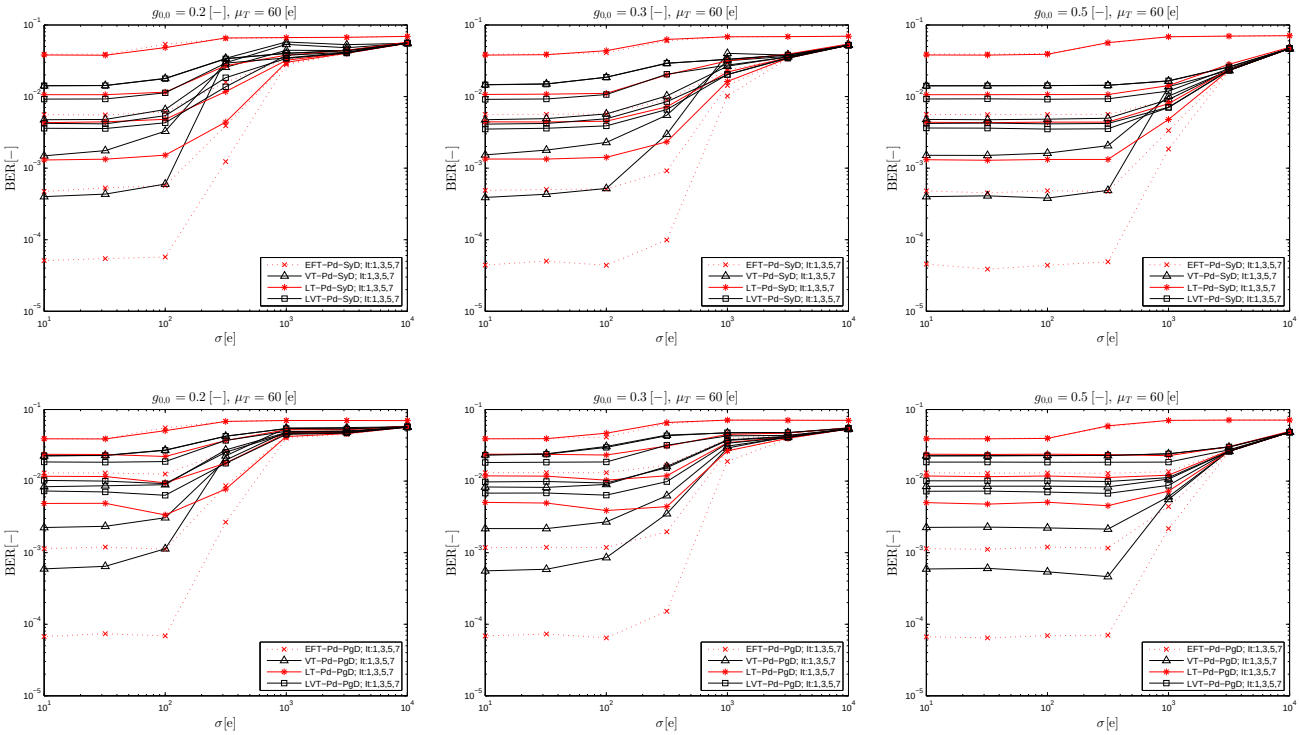
About Authors ...

Daniel KEKRT was born in Prague, Czech Republic, in 1981. He received the M.Sc. degree in electrical engineering from the Czech Technical University in Prague (CTU) in 2005. He is now working towards his Ph.D. at CTU. His research interests include image processing, iterative detection, 2D iterative decoding networks and joint iterative synchronization and detection.

Tomáš LUKEŠ was born in Prague, Czech Republic, in 1987. He received his M.Sc. degree in signal and image processing from Czech Technical University in Prague (CTU) in 2013. He is currently pursuing a Ph.D. at CTU and collaborates with the Biological Imaging Laboratory at the Institute of Cellular Biology and Pathology at the Charles University in Prague and Laboratoire d'Optique Biomedicale, EPFL, Switzerland. His research interests are image processing, microscopy image analysis and advanced optical microscopy methods.

Miloš KLÍMA was born in Prague, Czech Republic, in 1951. He graduated from the Czech Technical University in Prague (CTU) in 1974. Currently he is a Full Professor at the same university and head of the Multimedia Technology Group (MMTG). He is active in the field of photonics and image processing for multimedia and security technology. He is a member of IEEE, SPIE and Czech and Slovak Society for Photonics.

Karel FLIEGEL was born in Prague, Czech Republic, in 1979. He received his M.Sc. degree in electrical engineering from the Czech Technical University in Prague (CTU in Prague) in 2004 and Ph.D. from the CTU in Prague in 2011. Now he is an assistant professor with the Multimedia Technology Group (MMTG) at the CTU in Prague. His research interests include image processing, imaging systems, image and video compression.



(a) Main ray attenuation 0.2.

(b) Main ray attenuation 0.3.

(c) Main ray attenuation 0.5.

Fig. 15. The performance analyses of random patterns restoration by various IDNs using Pd-SyD and Pd-PgD technique. Patterns are captured with half resolution by the color camera based on the Bayer CFA.