




Classifying Image Sequences with the Markov Chain Structure and Matrix Normal Distributions

Ewaryst Rafajłowicz^(✉) 

Faculty of Electronics, Wrocław University of Science and Technology,
Wrocław, Poland
ewaryst.rafajlowicz@pwr.edu.pl

Abstract. We consider the problem of classifying image sequences to several classes. Such problems arise in numerous applications, e.g., when a task to be completed requires that all sub-tasks are properly executed. In order to derive realistic classifiers for such complicated problems, we assume that images in the sequence form a Markov chain, while the conditional probability density function of transitions has the matrix normal distribution, i.e., it has the covariance matrix being the Kronecker product of inter-rows and inter-columns covariance matrices. Under these assumptions we derive the Bayes classifier for image sequences and its empirical version that is based on applying the plug-in rule. We also provide interpretable versions of such classifiers at the expense of additional assumptions. The proposed classifier is tested on the sequence of images from the laboratory experiments of detecting stages of an additive manufacturing process. Finally, we state conclusions and (partial) explanations on why the problem of classifying sequences of images is (much) more difficult than that of classifying individual images.

Keywords: Matrix normal distribution · Bayesian classifier · Classification of image sequences

1 Introduction

Our aim is discuss a way to develop a classifier for image sequences. Each sequence is considered as a whole entity that can be a member of a certain class and our aim is to build an appropriate classifier. In other words, a classifier obtains an ordered set of images as one input.

This task only seemingly reduces to known classification problems by vectorization, because then it is extremely difficult to take into account stochastic dependencies between images and their covariance structures.

A large number of examples can be pointed out when we need (or it is desirable) to classify whole image sequences. In particular, they include the following cases.

- Quality control of a manufacturing process when at each stage we have images of properly and improperly produced items. Then, we can classify an item as conforming only when all the sequence of images is similar to the proper sequence. This class of examples is our main focus (see Sect. 6).
- Learning and teaching of complicated tasks to be performed requiring high precision of movements. Examples include: laparoscopic surgery (see [26]), training professional sportsmen and women and autonomous parking (see, e.g., [13]).
- Collecting, e.g., cytological images of the same patient (see [2]) along time and comparing them with image sequences of other patients.
- Subsequent histological sections of the same tissue (see [5, 6]), but recognized as one entity in the same spirit as in CT and in MRI images.
- When states of a dynamic systems are described as matrices or images (see, e.g. [20]), then the ability of classifying their sequences are of importance to decide at which state of the evolution the system is, e.g., whether it is still in transient states or near the equilibria states.
- Recognition of untidy hand written words by splitting them into letters, but considering them as one entity and testing to which word they are mostly similar.

The ability of classifying whole image sequences can also be useful for image understanding, but this topic is far outside the scope of this paper. We refer the reader to [24] for more detailed discussion on image understanding and the bibliography.

Clearly, it is rather impossible to construct a universal classifier for image sequences. We impose the following constraints on the class of considered classification tasks (see the next section for details):

- we confine ourselves to images represented by grey levels,
- images in a given sequence have the Markov property of the first order (a generalization to a higher order Markov chains is not difficult),
- conditional densities of the Markov chain have matrix normal distributions (MND) – see Appendix for basic properties of MND.

The last assumption is made for pragmatic reasons, otherwise we usually do not have enough observations in order to estimate the full covariance matrix of large images. An alternative approach, when we do not have enough observations, is proposed in [23].

The paper is organized as follows:

- in the following section we provide a short review of the works that have common points with this paper,
- then, in Sect. 3, we provide the problem statement and preliminary results on the Bayesian classifiers for image sequences,
- these topics are continued in the next section, in which special cases are discussed,
- in Sect. 5 we provide the empirical version of the Bayes MND classifier for image sequences, while

– a laboratory example is discussed in Sect. 5.

The paper ends by concluding remarks, including a discussion on the following question: why is the classification of an image sequence such a difficult problem?

2 Previous Work

In this section we provide a short survey of papers on classifiers that arise in cases when the assumption that class densities have the MND distribution holds. Then, we briefly discuss recent works on classifying image sequences.

The role of multivariate normal distributions with the Kronecker product structure of the covariance matrix for deriving classifiers was appreciated in [10], where earlier results are cited. In this paper the motivation for assuming the Kronecker product structure comes from repeated observations of the same object to be classified. The topic of classifying repeated measurements was further developed in [11], where repeated observations are stacked into a matrix according to their ordering along the time axis. In [11] the test for verifying the hypothesis on the Kronecker product structure of the covariance matrix was developed. The classifier based on the MND's assumption occurred to be useful for classifying images (see [17, 18], where it was applied to classifying images of flames from a gas burner). In [19] it was documented – by extensive simulations – that such classifiers are relatively robust against the class imbalance.

As far as we know, classifiers that are based on MND's for recognizing image sequences, considered as entities, were not considered in the literature and this is the main topic of this paper.

The above does not mean that the topic of classifying image sequence was not considered. It was, but using other assumptions and approaches. It is worth distinguishing the following cases.

1. A rough classification of videos according to their type (comedy, drama etc.). The stream of literature on these topics is the largest. It is completely outside the scope of this paper. The closest paper in this stream is [8], in which the classification of sporting disciplines by convolutional neural networks (CNN) is discussed.
2. Detecting changes in a video stream, e.g., for safety monitoring. Here, one can distinguish two problem statements, namely,
 - the so-called novelty detection, when a proper state is known, but the type of changes is unspecified (see [21], [15])
 - a directional change detection, when the class of possible changes is a priori known. One can meet such tasks in monitoring of production processes. They are similar in spirit to pattern recognition problems (see [16] for an example).
3. The classification of (an) object(s) that are visible on several subsequent frames (see [9] and bibliography therein).
4. The classification of image sequences, where each sequence is considered as one entity. This is our main topic.

The differences between group 3 and group 4 are, in some cases, subtle. For example, consider a camera mounted over a road and two cars (say, a truck and a small car behind it). If one is interested in classifying cars into small and large ones (and possibly in classifying their types), then we are faced with case 3. However, if the small car overtakes the large one, we can ask whether the overtaking maneuver was done properly or not. This task illustrates the one from group 4, since we have to recognize all stages of this maneuver.

3 Problem Statement and Preliminary Results

By \mathbb{X} we denote a sequence of ordered images \mathbf{X}_k , $k = 1, 2, \dots, K$, represented by $m \times n$ matrices of grey levels that are considered to be real-valued variables. In practice, grey levels are represented by integers from the range 0 to 255, but – at this level of generality – it seems reasonable to consider them as real numbers, without imposing constraints on their range.

Sequence \mathbb{X} can be classified to one of $J > 1$ classes, labeled as $j = 1, 2, \dots, J$. The following assumptions apply to all J classes, but we avoid indexing them by class labels, unless necessary.

As (1) \mathbb{X} is a random tensor, having a probability density function (p.d.f.), denoted further by $f(\mathbb{X})$ or, equivalently, by $f(\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K)$. Slightly abusing the notation, we shall write $f(\mathbf{X}_{L_1}, \dots, \mathbf{X}_{L_2})$ for p.d.f.'s of subsequences of \mathbb{X} , where $1 \leq L_1 < L_2 \leq K$.

As (2) Elements of \mathbb{X} form a Markov chain in the following sense:

$$f(\mathbf{X}_k | \mathbf{X}_{k-1}, \dots, \mathbf{X}_1) = f_k(\mathbf{X}_k | \mathbf{X}_{k-1}), \text{ for } k = 2, \dots, K, \quad (1)$$

where $f_k(\mathbf{X}_k | \mathbf{X}_{k-1})$ is the conditional p.d.f. of \mathbf{X}_k when \mathbf{X}_{k-1} is given. $f_k(\mathbf{X}_k | \mathbf{X}_{k-1})$ is known as the transition p.d.f. of moving from \mathbf{X}_{k-1} to \mathbf{X}_k , for every $k > 1$.

For $k = 1$ we assume that $f_1(\mathbf{X}_1 | \mathbf{X}_0) = f_1(\mathbf{X}_1)$, i.e., f_1 is the unconditional p.d.f. of random matrix \mathbf{X}_1 .

As (3) We assume that $\mathbf{X}_1 \sim \mathcal{N}_{n,m}(\mathbf{M}_1, U_1, V_1)$, i.e., $f_1(\mathbf{X}_1)$ is the MND with the expectation matrix \mathbf{M}_1 and U_1 as $n \times n$ inter-rows covariance matrix and V_1 as $m \times m$ covariance matrix between columns (see Appendix).

As (4) For $k > 1$ the transition p.d.f.'s $f_k(\mathbf{X}_k | \mathbf{X}_{k-1})$ are also assumed to have the MND's of the following form:

$$\frac{\alpha}{c} \exp \left[-\frac{1}{2} \text{tr} [U^{-1}(\mathbf{X}_k(\alpha) - \mathbf{M}_k) V^{-1}(\mathbf{X}_k(\alpha) - \mathbf{M}_k)^T] \right], \quad (2)$$

where c is the normalization constant which is given by:

$$c \stackrel{def}{=} (2\pi)^{0.5nm} \det[U]^{0.5n} \det[V]^{0.5m}, \quad (3)$$

while $n \times m$ matrix $\mathbf{X}_k(\alpha)$ is defined as follows: for $0 \leq \alpha \leq 1$

$$\mathbf{X}_k(\alpha) = \alpha \mathbf{X}_k + (1 - \alpha) \mathbf{X}_{k-1}. \quad (4)$$

In the above, \mathbf{M}_k plays the role of the mean matrix of the image sequence (video frame) at k -th step.

Several remarks are in order, concerning the above assumptions.

Remark 1 – *By selecting $0 \leq \alpha \leq 1$, one can control the influence of the previous image on the p.d.f. of the present one. The choice is case dependent. For example, when a small object is slowly moving over almost the same background, the influence of the previous frame is large, suggesting smaller values of α .*

– *For $\alpha = 1$ we obtain the independence between \mathbf{X}_k and \mathbf{X}_{k-1} . This case can happen, e.g., when images are taken from a very fast-moving train.*

Proposition 1. *Let As (1)–As (4) hold. Tentatively, we additionally assume:*

$$U_1 = U \quad \text{and} \quad V_1 = V. \tag{5}$$

Then, each $\mathbf{X}_k, k = 2, \dots, K$ has the matrix normal distribution with the expectation matrix, denoted as $\mathbf{M}_k(\alpha)$, of the following form:

$$\mathbf{M}_k(\alpha) = \alpha^{-1} [\mathbf{M}_k - (1 - \alpha)\mathbf{M}_{k-1}(\alpha)], \quad k = 2, 3, \dots, K, \tag{6}$$

where $\mathbf{M}_1(\alpha) \stackrel{\text{def}}{=} \mathbf{M}_1$.

The covariance matrices of \mathbf{X}_k 's are of the form:

$$C^{k-1}(\alpha)U_1, \quad C^{k-1}(\alpha)V_1, \quad k = 2, 3, \dots, K, \tag{7}$$

where

$$C(\alpha) \stackrel{\text{def}}{=} (1 + (1 - \alpha)^2)/\alpha^2. \tag{8}$$

Notice that $\mathbf{M}_k(\alpha) \rightarrow \mathbf{M}_k$ and $C(\alpha) \rightarrow 1$ as $\alpha \rightarrow 1$.

Proof. For $k = 2$ it suffices to integrate $f_2(\mathbf{X}_2|\mathbf{X}_1) f_1(\mathbf{X}_1)$ with respect to \mathbf{X}_1 . The rest of the proof goes by the induction, since – after this integration – we again obtain MND with the expectation (6) and the covariances (7), when $k = 2$ is substituted. •

Notice the growth of the variances in (7). For this reason, it is advisable to use $\alpha < 1$, but close to 1 and to apply the Markov scheme, proposed in As (4), to rather short image sequences.

Under As (1) and As (2) it is easy to derive the following expression for the natural logarithm of f

$$\log f(\mathbb{X}) = \sum_{k=2}^K \log f_k(\mathbf{X}_k|\mathbf{X}_{k-1}) + \log f_1(\mathbf{X}_1). \tag{9}$$

If, additionally, As (3) and As (4) hold, then for minus $\log f(\mathbb{X})$ we obtain:

$$\begin{aligned} LLF(\mathbb{X}, \mathbb{M}, U, V) &\stackrel{\text{def}}{=} -\log f(\mathbb{X}) = \log(c/\alpha) \\ &+ \frac{1}{2} \sum_{k=2}^K \text{tr}[U^{-1}(\mathbf{X}_k(\alpha) - \mathbf{M}_k) V^{-1}(\mathbf{X}_k(\alpha) - \mathbf{M}_k)^T \\ &\quad + \text{tr}[U_1^{-1}(\mathbf{X}_1 - \mathbf{M}_1) V_1^{-1}(\mathbf{X}_1 - \mathbf{M}_1)^T], \end{aligned} \tag{10}$$

where \mathbb{M} consists of $\mathbf{M}_k, k = 1, 2, \dots, K$. The *LLF* also depends on K, α, m, n , but we omit displaying them as arguments, since – in a given application – they remain the same for each class.

Each class has its own p.d.f., denoted further by $f_j(\mathbb{X})$ and the corresponding minus log-likelihood function: $LLF(\mathbb{X}, \mathbb{M}^{(j)}, U^{(j)}, V^{(j)})$, where $\mathbb{M}^{(j)}$ is the sequence of means for j -th class, while $U^{(j)}, V^{(j)}$ are the corresponding covariance matrices, $j = 1, 2, \dots, J$. We assume that for each class there exists a priori probability $p_j > 0$ that sequence \mathbb{X} was drawn from this class. Clearly $\sum_{j=1}^J p_j = 1$.

It is well known (see, e.g., [4]) that for the 0-1 loss function the Bayes risk of classifying \mathbb{X} is minimized by the following classification rule:

$$j^* = \arg \max_{1 \leq j \leq J} p_j f^{(j)}(\mathbb{X}), \tag{11}$$

where $f^{(j)}$ is the p.d.f. of sequences \mathbb{X} from j -th class.

Under all the above assumptions As (1)–As (4), our aim in this paper is the following:

1. having learning sequences of mutually independent $\mathbb{X}_n^{(j)}$'s from j -th class, $n = 1, 2, \dots, N_j, j = 1, 2, \dots, J$
2. and assuming proper classifications to one of the classes
3. to construct an empirical classifier that mimics (11) decision rule in the plug-in way

and to test this rule on real data. Notice that each $\mathbb{X}_n^{(j)}$ is a sequence itself. Its elements will further be denoted as $\mathbf{X}_{k,n}^{(j)}, k = 1, 2, \dots, K$.

4 Some Properties of the Bayes Classifier for Sequences

From (11) we obtain that the **Bayesian classifier for sequence \mathbb{X}** is the form:

$$j^* = \arg \min_{1 \leq j \leq J} \left[-\log(p_j) + LLF(\mathbb{X}, \mathbb{M}^{(j)}, U^{(j)}, V^{(j)}) \right] \tag{12}$$

or – in the full form:

\mathbb{X} is classified to class j^* , for which the following expression is minimal with respect to j :

$$\left\{ \frac{1}{2} \sum_{k=2}^K \text{tr}[(U^{(j)})^{-1} (\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)}) (V^{(j)})^{-1} (\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)})^T \right. \tag{13}$$

$$\left. + \text{tr}[U_1^{(j)}]^{-1} (\mathbf{X}_1 - \mathbf{M}_1) (V_1^{(j)})^{-1} (\mathbf{X}_1 - \mathbf{M}_1)^T + \log(c^{(j)}) \right\} - \log(p_j).$$

Above and further on the summand $\log(1/\alpha^2)$ is omitted, since it does not depend on j .

In order to reveal the interpretation of the optimal classifier (13), it is expedient to consider the following special cases.

Corollary 1. *Let As (1)–As (4) hold and, additionally, the a priori class probabilities are equi-distributed, i.e., $p_i = 1/J$. Then, the Bayes risk is minimized by this j for which the sum of the Mahalanobis distances between $\mathbf{X}_k(\alpha)$ and $\mathbf{M}_k^{(j)}$ is minimized.*

Proof. It suffices to observe that

$$\begin{aligned} & \text{tr}[(U^{(j)})^{-1} (\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)}) (V^{(j)})^{-1} (\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)})^T] \\ &= \text{vec}^T(\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)}) \Sigma_j^{-1} \text{vec}(\mathbf{X}_k(\alpha) - \mathbf{M}_k^{(j)}), \end{aligned} \tag{14}$$

where $\Sigma_j \stackrel{\text{def}}{=} U_j \otimes V_j$, while \otimes is the Kronecker product of matrices. •

Corollary 2. *If – in addition to the assumptions made in Corollary 1 – there are no correlations between rows and between columns (U_j 's and V_j 's are the identity matrices) and there are no correlations between images ($\alpha = 0$), then sequence \mathbb{X} is classified to this class j for which*

$$\sum_{k=1}^K \|\text{vec}(\mathbf{X}_k - \mathbf{M}_k^{(j)})\|^2 \tag{15}$$

is minimal, where $\|\cdot\|$ is the Euclidean norm of a vector. Thus, (15) is the nearest mean classifier in the generalized sense, i.e., the distance of all the sequence \mathbb{X} is compared to the sequences of all mean matrices $\mathbb{M}^{(j)}$, $j = 1, 2, \dots, J$ and the closest one is selected.

Corollary 2 is intuitively pleasing, but it is a very special case of (13).

Corollary 3. *For $J = 2$, if $U_1^{(1)} = U_1^{(2)}$, $V_1^{(1)} = V_1^{(2)}$ and $U_2^{(1)} = U_2^{(2)}$, $V_2^{(1)} = V_2^{(2)}$, then the classifier (13) is linear with respect to $\text{vec}(\mathbf{X}_k(\alpha))$, $k = 1, 2, \dots, K$.*

Proof. Follows directly from the right hand side of the equality in (14), since – under our assumptions – we have $\Sigma_1 = \Sigma_2$ and the quadratic terms vanish. •

5 An Empirical Bayes, Plug-In Classifier for Sequences of Matrices (images)

Having learning sequences of $\mathbb{X}_n^{(j)}$, $n = 1, 2, \dots, N_j$, for each class $j - j = 1, 2, \dots, J -$ at our disposal, we construct the empirical Bayes classifier, using the classical plug-in approach. Its derivation relays the assumptions As (1)–As (4), but – as we shall see – we can formally try to use it without imposing the MND structure of the observations. Clearly, if the observations do not follow MND, information contained in the full covariance matrix is partially lost, since we use only inter-rows and inter-columns covariances. On the other hand, however, we obtain a classifier, which is able to classify image sequences of a moderate size.

A Classifier for MND Sequences (CMNDS)

The learning phase. Firstly, p_j 's are estimated as $\hat{p}_j = N_j/N$, where $N = \sum_{j=1}^J N_j$. The means $\mathbb{M}^{(j)}$ are estimated as the empirical means of $\mathbb{X}_n^{(j)}$, $n = 1, 2, \dots, N_j$, but for large images and large K (long sequences) this is not a trivial computational task. These empirical means are denoted as $\hat{\mathbb{M}}^{(j)}$'s. Notice that, for practical reasons, we propose to estimate $\mathbb{M}^{(j)}$ as if $\mathbb{X}_n^{(j)}$, $n = 1, 2, \dots, N_j$ were mutually independent, i.e., for $\alpha = 1$. We introduce $\alpha < 1$ in the testing phase only when it leads to the reduction of the classification error.

The estimation of $U^{(j)}$'s and $V^{(j)}$'s is done in a non-classic way. Details are provided in the Appendix. The resulting estimates are denoted as $\hat{U}^{(j)}$'s and $\hat{V}^{(j)}$'s.

The recognition phase. When new sequence \mathbb{X} is to be classified we use the empirical version of (12) rule, i.e., it is classified to class \hat{j} such that

$$\hat{j} = \arg \min_{1 \leq j \leq J} \left[-\log(\hat{p}_j) + LLF(\mathbb{X}, \hat{\mathbb{M}}^{(j)}, \hat{U}^{(j)}, \hat{V}^{(j)}) \right]. \quad (16)$$

The constant c that is present in LLF also depends on j , but our experiments indicate that in some cases it is better to consider it as a constant and to neglect it (as done in the example presented in the next section).

The assessment of the quality of learning can be done by the classic approach, namely, by the cross-validation. Notice, however, that we have to estimate two covariance matrices for each class, which may be difficult, even for small images, due to the lack of sufficiently long learning sequences. The second difficulty is the possibility that $\hat{U}^{(j)}$ and/or $\hat{V}^{(j)}$ are ill-conditioned. Even if we replace the calculations of their inversions by solving the corresponding sets of linear matrix equations, a kind of the regularization may be necessary.

6 A Laboratory Example

In order to test the CMNDS, we use the same example as in [17], but this time we consider triples of subsequent images as one sequence to be classified. These images were taken during the monitoring of a laser based additive manufacturing process of constructing a thin wall, described in more detail in [22].

The classification (and then decision) problem that arises during monitoring of this process is to determine whether the laser head is above the main body of the wall (Class 1) or near one of its ends (Class 2). This task cannot be solved just by gauging positions of the laser head, since near the ends the wall it becomes thicker and thicker as construction of the wall is progressing. Additionally, these thicker parts occupy larger and larger of the wall. Precisely this unwanted behavior is to be prevented by: firstly, recognizing that a thicker end begins and then by reducing the laser power appropriately (see [22] for details concerning the reduction of the laser power). Here, we concentrate on the recognition phase only.

Original images were down-sampled by 10 to the size 12×24 . Then, they were averaged (each class separately). The resulting images are shown in Fig. 1, where the left hand side image corresponds to Class 1 and the second one is typical for Class 2.

Three element sequences, typical for Class 1, consists of:

- (a) either three images as the one on the l.h.s. of Fig. 1 or
- (b) two such images and the one similar to that on the r.h.s. of this figure.

Analogously, the triples typical for Class 2 contain:

- (c) either three images like the one on the r.h.s. or
- (d) two of this kind and one similar to the l.h.s. sub-image.

For learning and testing purposes we had 300 such triples, but classes are not well balanced, since the laser head spends much more time in the middle of the wall than near its ends.

Remark 2. Notice that ordering of images in these two kinds of sequences is not artificial – it is natural for this process, since the laser head moves back and forth along the wall. However, the presence of the sequences like those described as (b) and such as mentioned in (d) may lead to large classification errors.



Fig. 1. Averaged images typical for Class 1 (left panel) and for Class 2 (right panel)

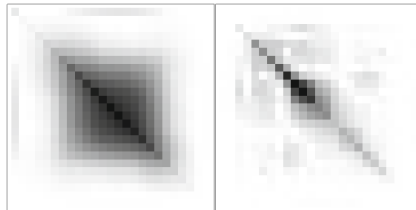


Fig. 2. Estimated V matrices for Class 1 (left panel) and Class 2 (right panel)

Matrices U and V for both classes were estimated by the method that is described in the learning phase of CMNDS and in the Appendix. The results are shown in Fig. 2 for V -type matrices and in Fig. 3 for U -type matrices. As one can observe, both U -type and V -type matrices are essentially different between

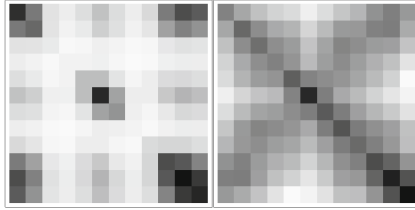


Fig. 3. Estimated U matrices for Class 1 (left panel) and Class 2 (right panel)

classes. Thus, we cannot use a linear classifier and therefore the full version of the quadratic classifier (16) was used in our example.

The following cross-validation methodology was used for testing CMNDS (see [1] for the survey of the test error estimation of classifiers). The whole sequence of triple sequences was split at random into the learning sequence of the length 125 the testing sequence of the length 175. Then, the matrices of means and covariances were estimated and plugged-in into the classifier, which was tested on the remaining 175 triples. The classification error was stored and the whole cycle of random drawing, learning and testing was repeated 1000 times. The averaged classification error (for $\alpha = 0.9$) was the following: 32% with minor fluctuations between all 1000 runs.

This result is rather disappointing, since for almost the same MND classifier, but applied to individual images, we obtained 4% of the averaged classification errors, using the same sequence of 900 images and the same methodology of testing the classifier.

One of possible reasons is that we have a relatively small number of learning and testing examples, namely, 900 images provide only 300 of triple image sequences. As a remedy in this example one may try to extend the data artificially, in a way similar to those that are used in imputation techniques, e.g., as it is proposed in [7], but this is outside the scope of this paper.

The reasons of a high recognition errors can be case-dependent (see Remark 2), but – in general – they indicate that the problem of classifying image sequences is much more difficult in practice than one might expect. Notice, however, that we do not apply any feature selection techniques, i.e., raw image triples were fed as inputs both in the learning and the testing phase. Applying a dedicated feature selection technique, e.g., a modified version of the method proposed in [3], one may expect much better results.

7 Concluding Remarks

Under several restricting, but interpretable and partly removable, assumptions the method of classifying image sequences (considered as entities) is proposed. It was extensively tested on image sequences from laboratory experiments, concerning the monitoring of the additive manufacturing, laser based, process. The results of testing indicate that the method works properly, but the percentage

of correct classifications (68%) is lower than 94% obtainable under the MND assumptions, i.e., when images are considered separately. This conclusion is in agreement with the results reported in [8] that classifying individual images may sometimes lead to a better correct classification rates than classifying whole sequences. These facts indicate that problems of classifying image sequences is much more difficult than classifying individual images. It requires further research on deciding which problem statement is more appropriate in a given application.

Acknowledgements. Special thanks are addressed to Professor J. Reiner and to MSc. P. Jurewicz from the Faculty of Mechanical Engineering, Wroclaw University of Technology for common research on laser power control for additive manufacturing.

Appendix: MND and Its Estimation

The matrix normal distribution (MND) has the probability density function of the form (see, e.g., [14]):

$$f(\mathbf{X}) = \frac{1}{c} \exp \left[-\frac{1}{2} \text{tr}[U^{-1}(\mathbf{X} - \mathbf{M}) V^{-1} (\mathbf{X} - \mathbf{M})^T] \right], \tag{17}$$

where c is the normalization constant, which is given by:

$$c \stackrel{def}{=} (2\pi)^{0.5nm} \det[U]^{0.5n} \det[V]^{0.5m}, \tag{18}$$

where $n \times m$ matrix M denotes the mean.

Concerning the covariance structure of MND densities:

1. $n \times n$ matrix U denotes the covariance matrix between rows of an image,
2. $m \times m$ matrix V stands for the covariance matrix between columns, we assume that $\det[U] > 0, \det[V] > 0$. We use the notation: $\mathbf{X} \sim \mathcal{N}_{n,m}(\mathbf{M}, U, V)$. The MND is a special case of a general class of Gaussian p.d.f.'s, since $\text{vec}(\mathbf{X}) \sim \mathcal{N}_{nm}(\text{vec}(\mathbf{M}), \Sigma)$, where $\text{vec}(\mathbf{X})$ is the operation of stacking columns of matrix \mathbf{X} , while Σ is an $nm \times nm$ covariance matrix, which is the Kronecker product of U and V .

We assume that we have the sequence of observations: $\mathbf{X}_i, i = 1, 2, \dots, N$. Conditions for estimating properly the covariance matrices can be found in [12]. The maximum likelihood estimates (MLE) of the covariance matrices fulfil the following set of equations (see [12,25]):

$$\hat{U} = \frac{1}{Nm} \sum_{i=1}^N (\mathbf{X}_i - \hat{\mathbf{M}}) \hat{V}^{-1} (\mathbf{X}_i - \hat{\mathbf{M}})^T, \tag{19}$$

$$\hat{V} = \frac{1}{Nn} \sum_{i=1}^N (\mathbf{X}_i - \hat{\mathbf{M}})^T \hat{U}^{-1} (\mathbf{X}_i - \hat{\mathbf{M}}). \tag{20}$$

Equations (19) and (20) can be solved by the flip-flop method. It was proved in [25] that one iteration is sufficient to obtain the efficient estimators of U_j and V_j .

References

1. Anguita, D., Ghelardoni, L., Ghio, A., Ridella, S.: A survey of old and new results for the test error estimation of a classifier. *J. Artif. Intell. Soft Comput. Res.* **3**(4), 229–242 (2013)
2. Bruzdziński, T., Krzyżak, A., Fevens, T., Jeleń, L.: Web-based framework for breast cancer classification. *J. Artif. Intell. Soft Comput. Res.* **4**(2), 149–162 (2014)
3. Chang, O., Constante, P., Gordon, A., Singana, M.: A novel deep neural network that uses space-time features for tracking and recognizing a moving object. *J. Artif. Intell. Soft Comput. Res.* **7**(2), 125–136 (2017)
4. Devroye, L., Györfi, L., Lugosi, G.: *A Probabilistic Theory of Pattern Recognition*. Springer, Berlin (2013). <https://doi.org/10.1007/978-1-4612-0711-5>
5. Górniak, A., Skubalska-Rafajłowicz, E.: Registration and sequencing of vessels section images at macroscopic levels. In: Saeed, K., Homenda, W. (eds.) *CISIM 2015*. LNCS, vol. 9339, pp. 399–410. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24369-6_33
6. Górniak, A., Skubalska-Rafajłowicz, E.: Tissue recognition on microscopic images of histological sections using sequences of Zernike moments. In: Saeed, K., Homenda, W. (eds.) *CISIM 2018*. LNCS, vol. 11127, pp. 16–26. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99954-8_2
7. Jordanov, I., Petrov, N., Petrozziello, A.: Classifiers accuracy improvement based on missing data imputation. *J. Artif. Intell. Soft Comput. Res.* **8**(1), 31–48 (2018)
8. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1725–1732 (2014)
9. Kafai, M., Bhanu, B.: Dynamic Bayesian networks for vehicle classification in video. *IEEE Trans. Industr. Inf.* **8**(1), 100–109 (2012)
10. Krzyśko, M., Skorzybut, M.: Discriminant analysis of multivariate repeated measures data with a Kronecker product structured covariance matrices. *Stat. Pap.* **50**(4), 817–835 (2009)
11. Krzysko, M., Skorzybut, M., Wolynski, W.: Classifiers for doubly multivariate data. *Discussiones Mathematicae: Probability & Statistics*, p. 31 (2011)
12. Manceur, A.M., Dutilleul, P.: Maximum likelihood estimation for the tensor normal distribution: algorithm, minimum sample size, and empirical bias and dispersion. *J. Comput. Appl. Math.* **239**, 37–49 (2013)
13. Notomista, G., Botsch, M.: A machine learning approach for the segmentation of driving maneuvers and its application in autonomous parking. *J. Artif. Intell. Soft Comput. Res.* **7**(4), 243–255 (2017)
14. Ohlson, M., Ahmad, M.R., Von Rosen, D.: The multilinear normal distribution: introduction and some basic properties. *J. Multivariate Anal.* **113**, 37–47 (2013)
15. Prause, A., Steland, A.: Sequential detection of three-dimensional signals under dependent noise. *Sequential Anal.* **36**(2), 151–178 (2017)
16. Rafajłowicz, E., Steland, A.: The Hotelling-like T^2 control chart modified for detecting changes in images having the matrix normal distribution. In: *Stochastic Models, Statistics and Their Applications*. Springer, Cham (2019, accepted)
17. Rafajłowicz, E.: Data structures for pattern and image recognition with application to quality control *Acta Polytechnica Hungarica. Informatics* **15**(4), 233–262 (2018)

18. Rafajłowicz, E.: Classifiers for matrix normal images: derivation and testing. In: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J.M. (eds.) ICAISC 2018. LNCS (LNAI), vol. 10841, pp. 668–679. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91253-0_62
19. Rafajłowicz, E.: Robustness of raw images classifiers against the class imbalance – a case study. In: Saeed, K., Homenda, W. (eds.) CISIM 2018. LNCS, vol. 11127, pp. 154–165. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99954-8_14
20. Rafajłowicz, E., Rafajłowicz, W.: Linear matrix-state systems and their use for image-driven control. In: 10th International Workshop on Multidimensional (nD) Systems (nDS), 13–15 September 2017, Zielona Góra, Poland, Danvers, pp. 1–6. IEEE (2017)
21. Rafajłowicz, E.: Detection of essential changes in spatio-temporal processes with applications to camera based quality control. In: Steland, A., Rafajłowicz, E., Szajowski, K. (eds.) Stochastic Models, Statistics and Their Applications, pp. 433–440. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-13881-7_48
22. Rafajłowicz, W., Jurewicz, P., Reiner, J., Rafajłowicz, E.: Iterative learning of optimal control for nonlinear processes with applications to laser additive manufacturing. *IEEE Trans. Control Syst. Technol.* (2018, accepted, available on-line)
23. Skubalska-Rafajłowicz, E.: Sparse random projections of camera images for monitoring of a combustion process in a gas burner. In: Saeed, K., Homenda, W., Chaki, R. (eds.) CISIM 2017. LNCS, vol. 10244, pp. 447–456. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59105-6_38
24. Szczepaniak, P., Tadeusiewicz, R.: The role of artificial intelligence, knowledge and wisdom in automatic image understanding. *J. Appl. Comput. Sci.* **18**(1), 75–85 (2010)
25. Werner, K., Jansson, M., Stoica, P.: On estimation of covariance matrices with Kronecker product structure. *IEEE Trans. Signal Process.* **56**(2), 478–491 (2008)
26. Wytyczak-Partyka, A., Nikodem, J., Klempous, R., Rozenblit, J., Klempous, R., Rudas, I.: Safety oriented laparoscopic surgery training system. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) EUROCAST 2009. LNCS, vol. 5717, pp. 889–896. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04772-5_114