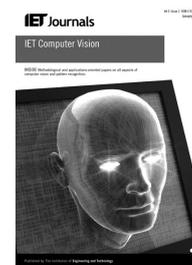


Published in IET Computer Vision
 Received on 16th December 2013
 Revised on 19th June 2014
 Accepted on 24th July 2014
 doi: 10.1049/iet-cvi.2014.0067



ISSN 1751-9632

Spatial codification of label predictions in multi-scale stacked sequential learning: a case study on multi-class medical volume segmentation

Frederic Sampedro¹, Sergio Escalera^{2,3}

¹Faculty of Medicine, Autonomous University of Barcelona, 08193 Barcelona, Spain

²Faculty of Mathematics, University of Barcelona, Gran Via de les Corts Catalanes 585, 08007 Barcelona, Spain

³Computer Vision Center, Campus UAB, Edifici O, 08193 Bellaterra, Barcelona, Spain

E-mail: fredsampedro@gmail.com

Abstract: In this study, the authors propose the spatial codification of label predictions within the multi-scale stacked sequential learning (MSSL) framework, a successful learning scheme to deal with non-independent identically distributed data entries. After providing a motivation for this objective, they describe its theoretical framework based on the introduction of the blurred shape model as a smart descriptor to codify the spatial distribution of the predicted labels and define the new extended feature set for the second stacked classifier. They then particularise this scheme to be applied in volume segmentation applications. Finally, they test the implementation of the proposed framework in two medical volume segmentation datasets, obtaining significant performance improvements (with a 95% of confidence) in comparison to standard Adaboost classifier and classical MSSL approaches.

1 Introduction

One of the most widely used assumptions in supervised learning is that data is independent and identically distributed (iid). However, there are many real world applications in which this assumption does not necessarily hold. Consider the case of object recognition in image or volume segmentation. It is clear that if one pixel or voxel belongs to a certain object category, it is very likely that its spatial neighbours also belong to the same object.

Sequential learning [1] breaks the iid assumption and assumes that samples are not independently drawn from a joint distribution of the data samples X and their labels G . In sequential learning, the training data consists of sequences of pairs (x, g) so that neighbouring examples exhibit some kind of correlation. In the literature, sequential learning has been addressed from different perspectives [2–6]. One of the most successful and flexible approaches in this scenario, especially in image segmentation applications, has been the multi-scale stacked sequential learning (MSSL) [7] and its multi-class extension [8].

In short, the MSSL approach in image segmentation proceeds in a two-level learning scheme. First, a base classifier outputs a label prediction for each image pixel. Then, the original feature set of each pixel is augmented (stacked) with the predicted label densities in several neighbouring scales around that pixel as new features. Finally, training a second classifier in this new training set outputs the final pixel classifications. Note that this rather simple framework has shown high performance results without assuming any specific correlation properties within

the neighbouring pixels, that is, by letting the second classifier learn these particular dependencies on any given dataset.

Although the MSSL model covers some flexibility in choosing the neighbouring lattices (cubical, spherical, Gaussian etc.) and scales, it relies only on the predicted labels density information for each scale in the design of the new features to be stacked. Specifically, these features do not encode the spatial distribution of the predicted labels, only its presence. While in many scenarios this approach may meet the performance criteria, we claim that the inclusion of the spatial distribution information of the predicted labels could enhance the performance results in many images and volume segmentation applications.

Therefore in this work, we propose a smart codification of the spatial distribution of the predicted labels to be stacked as new features within the MSSL framework. To accomplish this task, we focus on grid techniques to describe the spatial information using linear computation time and in particular in the use of the blurred shape model (BSM) [9] to obtain a robust descriptor that accounts for small shape variation independence (in contrast with the Zoning model [10]). The BSM has been successfully applied in several two-dimensional (2D) image analysis and classification tasks [9, 11].

Our contribution consists, on the one hand, in proposing the spatial codification of the predicted labels in multi-class MSSL and defining the corresponding mathematical scheme. On the other hand, we describe the adaptation of the proposed framework in multi-class volume segmentation problems, and show its performance results in two medical

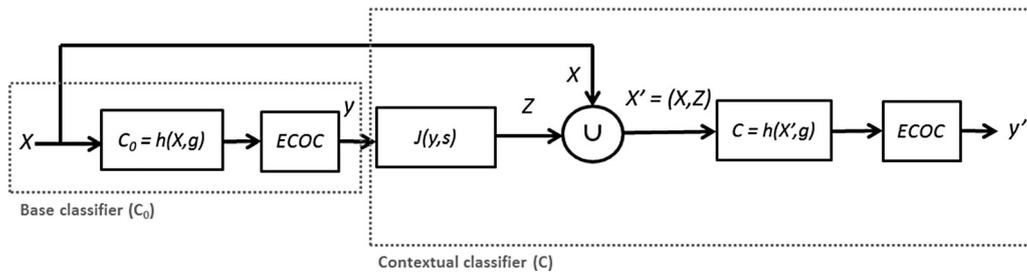


Fig. 1 Multiclass MSSL using ECOCs

volume datasets within a comparative analysis with respect to a direct AdaBoost [12, 13, 14] implementation and the original MSSL framework. This field of application is particularly relevant for different reasons. First, quantitative analysis derived from the segmentation of medical image or volume data provides essential clinical information in a number of diagnostic scenarios [13]. Second, carrying out the segmentation procedure manually or semi-automatically (especially in 3D datasets) by a trained specialist is sometimes highly time-inefficient in the clinical setting and suffers from inter- and intra-observer variabilities. Finally, any machine learning framework working in this domain should include a powerful contextual learning model to obtain accurate segmentations of the regions of interest, since in some applications even a small increase in accuracy may be very convenient for subsequent processing stages such as the computation of clinical indicators. To the best of our knowledge, very few works have addressed the use of MSSL in 3D medical data [15] and this is the first proposal to consider a spatial coherent descriptor of label predictions within the MSSL framework.

The rest of this paper is organised as follows. Section 2 introduces the mathematical framework of our contribution, Section 3 describes its performance results on two multi-class medical volume segmentation applications and Section 4 concludes the paper and points out some future work.

2 Methods

In this section, we describe the building blocks of our proposed learning framework and its adaptation to volume segmentation scenarios.

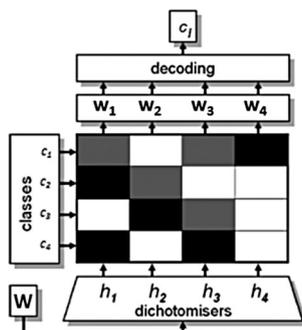
2.1 General framework

First, we revisit the principal building blocks of the multi-class MSSL framework (Fig. 1). As it has been

mentioned, the main idea of this model is to first train a base classifier (C_0) from the original training set X and their ground truth labels g . In a multi-class setting of L classes, a convenient extra layer to select the class output (y) should be added. In this work, we focus on error correcting output codes (ECOCs) to extend the MSSL framework to the multi-class case because of its demonstrated performance [16, 17]. A short description of this methodology is introduced as follows.

Given that most state-of-the-art learning strategies are defined to deal with two-class problems, extension to multi-class problems requires an ensemble of binary classifiers. In this sense, the ECOC approach has shown to be a powerful framework for the combination of classifiers to deal with multi-class data. In short, given an L -class problem, a coding matrix $M \in \{-1, 0, +1\}^{L \times n}$ is designed, where each of the n columns represents a binary classifier or dichotomiser (h_i), the L rows are defined as the codewords codifying each class c_i , $i \in [1, \dots, L]$, and $+1$, -1 identifies the class membership for a binary classifier, being 0 if the class is not considered by the classifier. Then, for any given test instance x , its codeword W is computed applying all binary classifiers, and its classification prediction is defined by the class with codeword at minimum distance given a distance metric. An ECOC design example is shown in Fig. 2.

Given the label predictions y of the base classifier, for each entry (e.g. pixel/voxel) we inspect its neighbourhood (with a given lattice such as cubical, spherical or Gaussian) at different scales s and compute a descriptor $J(y, s)$ that seeks to model the distribution of the predicted labels for each neighbourhood scale. Then, the original feature set is augmented with the new descriptors (Z) obtained from $J(y, s)$ and therefore all the entries of the training matrix X are extended with this new features obtaining an augmented training set X' . Finally, a second classifier C is now trained with this computed contextual information (X') and outputs the final classification label y' .



ECOC coding design for a 4-class problem. White, black, and grey positions correspond to the symbols $+1$, -1 , and 0 , respectively. Once the four binary problems (h_i), at the decoding step a new test sample X is tested by the n classifiers. Then, the new codeword $W = \{w_1, \dots, w_n\}$ is compared with the class code-words $\{c_1, \dots, c_n\}$, classifying the new sample by the class c_i , which codeword minimizes the decoding measure.

Fig. 2 ECOC design example

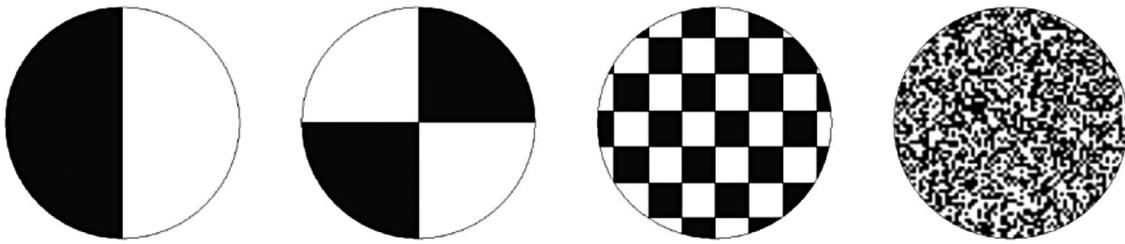


Fig. 3 Different spatial distributions for the same label density codification

The objective of this work is to discuss the role of the descriptor function $J(y, s)$. As it has been mentioned, the original definition of MSSL assumes that $J(y, s)$ encodes only the density of the predicted labels; that is, for each neighbourhood s , the proportion of $y=y_i, \forall i \in [1, \dots, L]$ within the neighbourhood entries is recorded. Although computationally convenient and performance efficient in many applications, we claim that codifying not only the predicted label densities but also its spatial distribution within the neighbourhood scale may prove useful in some applications by increasing the overall performance results. Consider the illustrative example in Fig. 3, where clearly all the drawings have the same label density information but a substantially different spatial distribution. If these drawings were the neighbour label predictions for the central pixel of a circular lattice for a given scale, it is clear that by encoding the spatial distribution as new features, the contextual classifier could derive a much more accurate re-classification rule for the pixel under consideration than when only providing the label density information.

To further illustrate the point, consider the simplified MSSL scenario in Fig. 4. Fig. 4a shows a ground truth segmentation of three objects (blue, red and green) in a given sample image. Suppose that the base classifier predictions are the ones in Fig. 4b. Now, without loss of generality, assume a single-scale stacked sequential learning scheme where a square lattice of a given size is used (the orange square in Fig. 4c). Consider the $J(y, s)$ computation of the pixel located at the centre of the orange square.

Using the density descriptor for $J(y, s)$, the only information provided in the new features for this pixel is that there is a certain proportion of green predictions and blue predictions around him (optionally including the background proportion). Notice that there is no apparent reason to think that the contextual classifier would learn that the pixel under consideration should change its label, since it seems coherent that a green pixel may be nearby a blue object (as there are other examples in the bottom region).

Now consider instead a possible $J(y, s)$ that encodes the spatial distribution of the predicted labels within the neighbourhood scale. For instance, consider the simplest case of selecting as new features the number of predictions of each label in each quadrant in a 2×2 grid decomposition of the neighbourhood as shown in Fig. 4c. Now the contextual classifier will have more information and may learn that it is suspicious that a green pixel has in its right a blue object, since this is not observed anywhere else in the image, and consider refining its prediction to red in the final classification.

In this work, we propose the codification of the spatial distribution of the predicted labels $J(y, s)$ in each scale through the BSM [9, 11]. This model is a particular grid technique (with linear complexity for each scale) that is also robust to small shape variations. Its general properties are described as follows. In short, the BSM provides a simple and fast descriptor that defines a probability density function of the shape of an object by encoding its probabilistic spatial variability. This model has proven useful in multi-class 2D symbol categorisation problems (within an ECOC framework) because of its robustness to symbol's rigid and elastic deformations, successfully recognising clefs in music scores, and multiple object categories in public image categorisation datasets [9, 11].

2.2 MSSL–BSM framework adaptation to volume segmentation applications

We now particularise the general framework to be applied in volume segmentation scenarios. In this context, the neighbourhood lattices are defined in the 3D space. Examples include cubical, spherical or Gaussian lattices. Therefore the BSM needs to be adapted to be applied in a 3D context. Taking into account the general high computational demands in volume processing scenarios, our proposal modification of the BSM descriptor computation is presented in Fig. 5. Note that for simplicity, we assume the

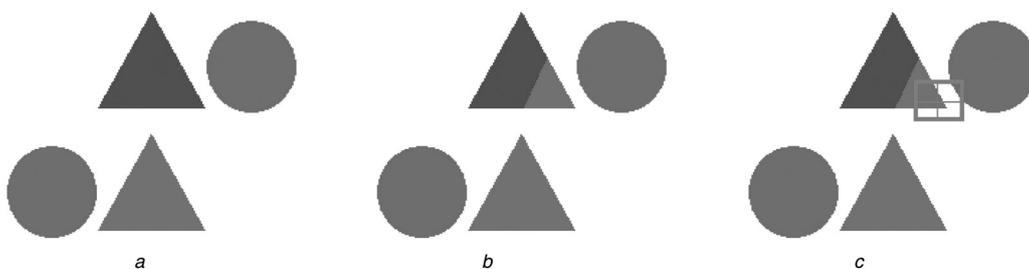


Fig. 4 Simple example showing the relevance of encoding the spatial structure of the predicted labels

a Ground truth image

b Base classifier prediction

c Neighbourhood around the relevant misclassified pixel

Given a voxel's neighborhood volume V in scale s , and a given label y :

1. Obtain the set A of voxels predicted with label y within V .
2. Divide V in *Height* \times *Width* \times *Depth* equal m -size sub-regions $R = \{r_1, \dots, r_{m^3}\}$ with α_i being the central point in region r_i , $i \in [1, \dots, m^3]$.
3. Let $N(r_i)$ be the neighbor regions of region r_i , defined as $N(r_i) = \{r_k | r_k \in R, \|\alpha_k - \alpha_i\| < 2|\beta|\}$, where β is the cell size.
4. Let r_i^x be the region which contains the point x .
5. Initialise the probability vector v as $v(r_i) = 0, \forall i \in [1, \dots, m^3]$
6. **For** each point $x \in A$,
 - $D = 0$
 - For** each $r_i \in N(r_i^x)$,
 - $d_i = d(x, r_i) = \|x - \alpha_i\|^2$
 - $D = D + \frac{1}{d_i}$
 - End_for**
 - Update the probability vector v as $v(r_i) = v(r_i) + \frac{1}{D \cdot d_i}$
 - End_for**
7. Obtain the BSM descriptor as $d_{BSM}(r_i) = \frac{v(r_i)}{\sum_{j=1}^{m^3} v(r_j)} \forall i \in [1, \dots, m^3]$

Fig. 5 Adaptation of the BSM descriptor computation algorithm in the 3D setting for a particular label y

same discretisation scale on the three dimensions (height, width and depth), but this decision could be modified depending on the application.

We also revisit the ECOC framework within a volume segmentation scenario. Basically, in this context the data entries x are the voxel's features (with W its associated codeword) and the possible classification labels correspond to the voxel's belonging to the different segmentation regions. In [18], the authors shown that using the Hamming distance (HD) metric [17] as the ECOC decoding measure has proven successful results in volume segmentation scenarios (Fig. 6).

The final BSM-based spatial descriptor J to be stacked within the MSSL framework is therefore obtained by a vectorisation of the BSM descriptors of each label y_i at each scale s_i , with a total size $L \cdot N_s$ (N_s being the number of scales). Note that this new feature set can become very large and the curse of dimensionality must be taken into consideration. However, most volume datasets contain a large number of voxels and therefore in general this

increase in the number of features should not be an issue. Fig. 7 summarises the global proposed scheme in an illustrative manner.

Finally, a unifying picture of the proposed methodology is described. The main point is that any machine learning framework working in medical volume segmentation problems needs to model appropriately the context of the particular voxel to be classified. Unfortunately, given the high heterogeneity and irregular morphology of most regions of interest to be segmented in these applications, modelling the context properties analytically is too restrictive.

Therefore an interesting approach is to actually learn the context properties through the predicted labels of the learning algorithm. The MSSL framework is a successful setting designed to accomplish this task by augmenting the original voxel feature set with the proportions of predicted labels by the learning algorithm for each class and in several context scales, and then training a contextual classifier on this extended feature set. Our contribution relies on proposing a new methodology where we improve

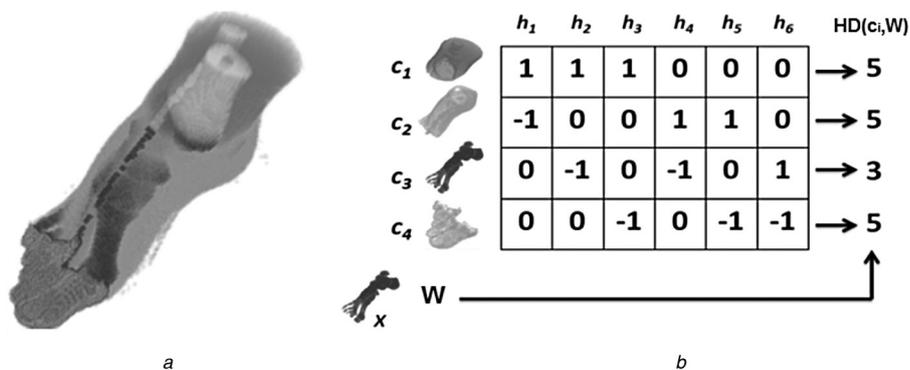


Fig. 6 ECOC decoding measure has proven successful results in volume segmentation scenarios

a Input volume

b One-vs-one ECOC coding matrix of $L(L-1)/2$ classifiers and HD decoding for a four-class problem

For each column, 1 and -1 indicate the two classes learned by h_i and 0 corresponds to the classes not considered for training by its correspondent classifier h_i . Given a new voxel sample x and its associated codeword W , the HD computation is obtained counting the number of different entries between W and each c_i . Therefore in this case, the classification label for sample x would be 3, corresponding to the codeword of the class at the minimum HD

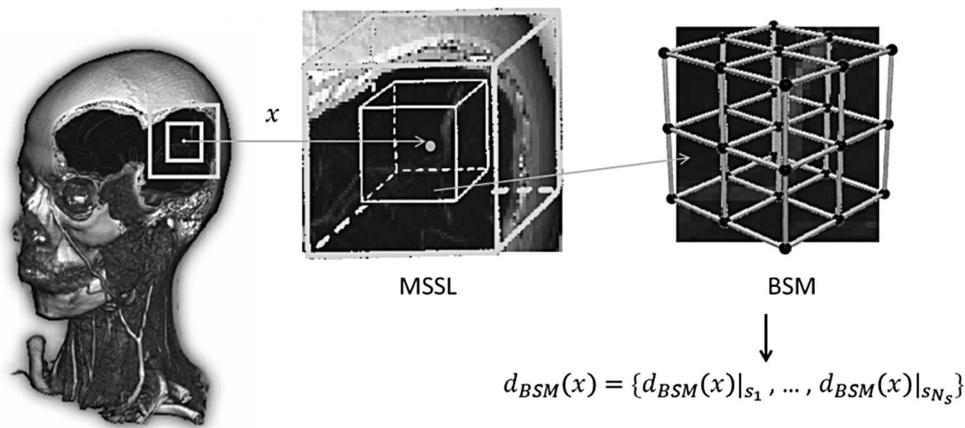


Fig. 7 Illustration of a voxel processing instance within the proposed MSSL–BSM framework

Neighbourhood of a voxel x is analysed at different scales (s_1, \dots, s_{N_s}), and in each one of them a BSM descriptor is computed to model the spatial distribution of the predicted labels

Final voxel descriptor $d_{BSM}(v)$ is obtained by the concatenation of the descriptors in each scale

the context description of the MSSL framework by introducing a smart spatial codification (based on the BSM) of the predicted labels.

At the implementation level, the process is summarised as follows: at the training stage, given the original voxel training set, a base classifier is trained. Then, the original training set is augmented by appending as new features the BSM descriptors of the predicted labels of the base classifier at different context scales (having the entry voxel as the centre voxel of the context neighbourhood lattices). The final contextual classifier is then obtained by learning the augmented training set. At the test level, first all the test voxel entries are classified using the learned base classifier. Then, for each test entry, its augmented feature vector is computed by combining the original features with the BSM descriptors obtained from the predicted labels at different context scales (with the same protocol as in the training phase). Finally, the learned contextual classifier is applied to this augmented entry and the final voxel classification label is obtained.

3 Results and discussion

In this section, we present and discuss the results of applying the proposed methodology in two medical volume segmentation datasets. The first one aims to segment a brain volume model of size $181 \times 217 \times 181$ (<http://www.slicer.org/archives>) in three compartments (right hemisphere, left hemisphere and cerebellum) providing as input only a small ground truth

subset of voxels of each of the four classes. The second one is analogous to the first but using a thorax volume model of size $128 \times 128 \times 128$ (http://www.fredsampedro.com/other/thorax_volume_model_8labs.dat), which we are interested to divide in eight structures (spine, bones, heart and circulatory system, lungs, pancreas, liver, kidneys and intestine) providing again as a training set only a small subset of labelled voxels of each of the eight classes (Fig. 8). For both cases, the voxel feature set includes the voxel 3D coordinates, intensity, gradient magnitude and gradient along each of the three dimensions, defining an initial feature set for each voxel of eight features. These datasets have been used, among others, to test classification performance of different state-of-the-art classifiers (such as AdaBoost [12]) in volume segmentation scenarios [18].

The settings defined to test our system are defined as follows. Without loss of generality, we chose AdaBoost [12] (with 200 decision stumps) as the learning algorithm for both the base and the contextual classifier and use cubical lattice neighbourhoods with BSM grids of $4 \times 4 \times 4$ voxels. The multi-class classification is then performed by the one-versus-one ECOC framework [19]. The main goal of this section is to compare the performance results on different settings applying a standard AdaBoost classifier, the standard MSSL framework using density descriptors and our proposed MSSL framework with the spatial codification of the predicted features using BSM.

We first conducted a single-scale sequential learning experiment on the brain dataset. For computational reasons, we chose a cubical neighbourhood lattice of $18 \times 18 \times 18$ voxels. We also chose a sampled version of the original volume to conduct the experiments in a reasonable amount of time. Accuracy results are shown in Table 1 for a

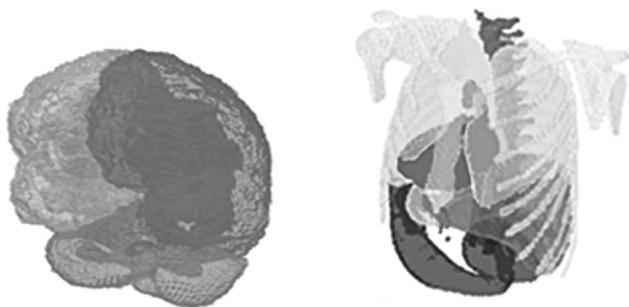


Fig. 8 Brain dataset (three labels) and thorax dataset (eight labels)

Table 1 Accuracy segmentation results of each label (y_1, y_2, y_3) in the brain volume dataset using a single-scale framework

	y_1	y_2	y_3
base classifier	0.981811	0.972827	0.973386
contextual classifier – density codification	0.984056	0.972469	0.980048
contextual classifier – BSM codification	0.988299	0.977877	0.985226

Table 2 Accuracy of each label (y_1, y_2, y_3) in the brain volume dataset using a two-scale framework

	y_1	y_2	y_3
base classifier	0.981811	0.972827	0.973386
contextual classifier – Density codification	0.988066	0.972603	0.982736
contextual classifier – BSM codification	0.990090	0.976581	0.984343

training set of randomly chosen voxels of size 0.5% for each label from the total dataset.

Note that as it has been mentioned, the standard single-scale sequential learning framework provides substantial performance improvements with respect to a pure AdaBoost classifier. Also, as claimed, a global overall performance improvement is observed using the BSM codification.

The situation is analogous when a multi-scale approach is used. In this case, we chose two scales of sizes 4 and 12 within the same model and the same ground truth sample. Table 2 shows the results of this configuration. Note that in this case we observe both the performance improvement of the multi-scale against the single-scale scheme and the BSM against density codification. Fig. 9 shows the visual results in this scenario. However, most part of the

improvement is not easily appreciated visually since it is mainly focused on correcting a small number of voxels from very specific boundaries, as discussed in Section 2.1.

Finally, using the thorax dataset which models an 8-class classification problem, using two scales of size 4 and 12 and a 10% random sample as training set (to account for the substantial increase in the number of features), similar results are obtained (Table 3). Fig. 10 shows the associated visual results for some of the volume regions.

A statistical analysis is carried out in order to justify the significance of these improvements. Note that the results shown above are for a particular experiment after randomly selecting a small percentage of voxels as ground truth. Logically, a different random selection of the ground truth voxels will lead to slightly different accuracy results in absolute value, since the voxels located near the region boundaries are more informative than the others. However, in this setting we are interested on the relative performance improvement of the BSM contextual classifier with respect to the density one. Therefore, regarding the relative improvement of both approaches, after running a number of experiments ($N=10$ because of computational reasons), the maximum empirical standard deviation recorded was inferior to 0.001%. If we take that value as a worst-case scenario we can conduct several statistical tests to show the significance of the obtained results.

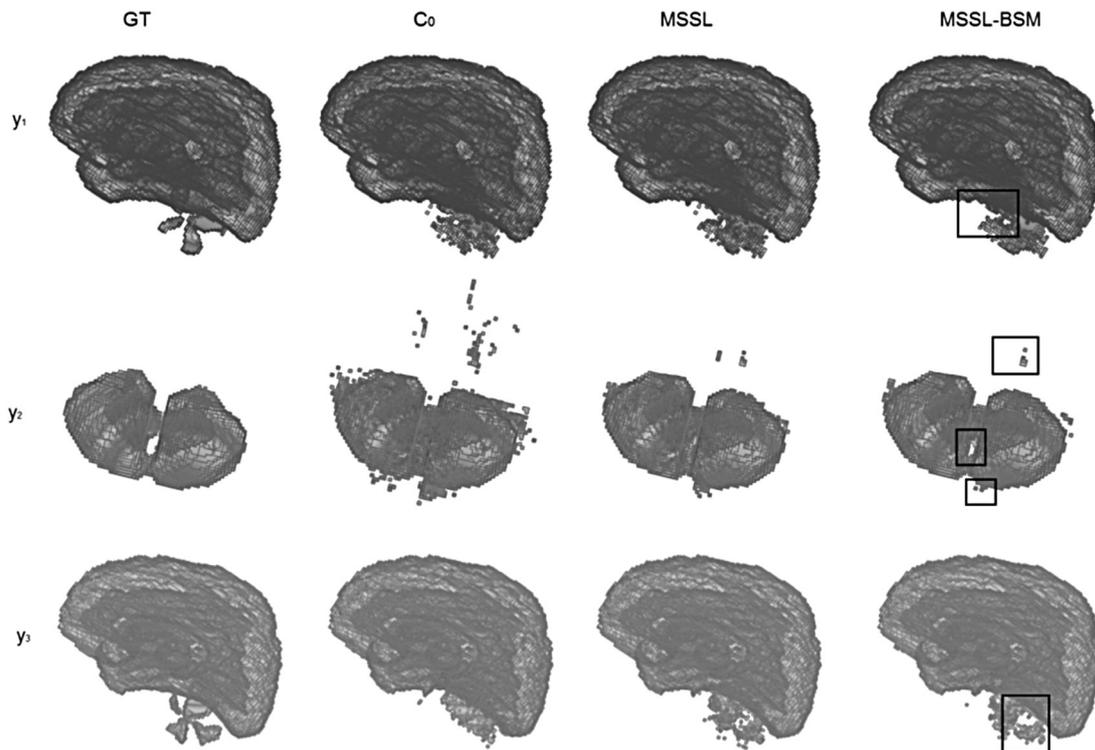


Fig. 9 Visual results on a sampled version of the brain dataset
Regions of highest improvement as marked with rectangles in the MSSL–BSM column

Table 3 Accuracy of each label (y_1, \dots, y_8) in the thorax volume dataset using a two-scale framework

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
base classifier	0.991308	0.986269	0.990308	0.997527	0.995772	0.995192	0.990414	0.992561
MSSL– density codification	0.990456	0.992298	0.990650	0.997332	0.995772	0.995277	0.990180	0.993834
MSSL–BSM codification	0.996194	0.995947	0.990906	0.997420	0.995257	0.995379	0.991349	0.993962

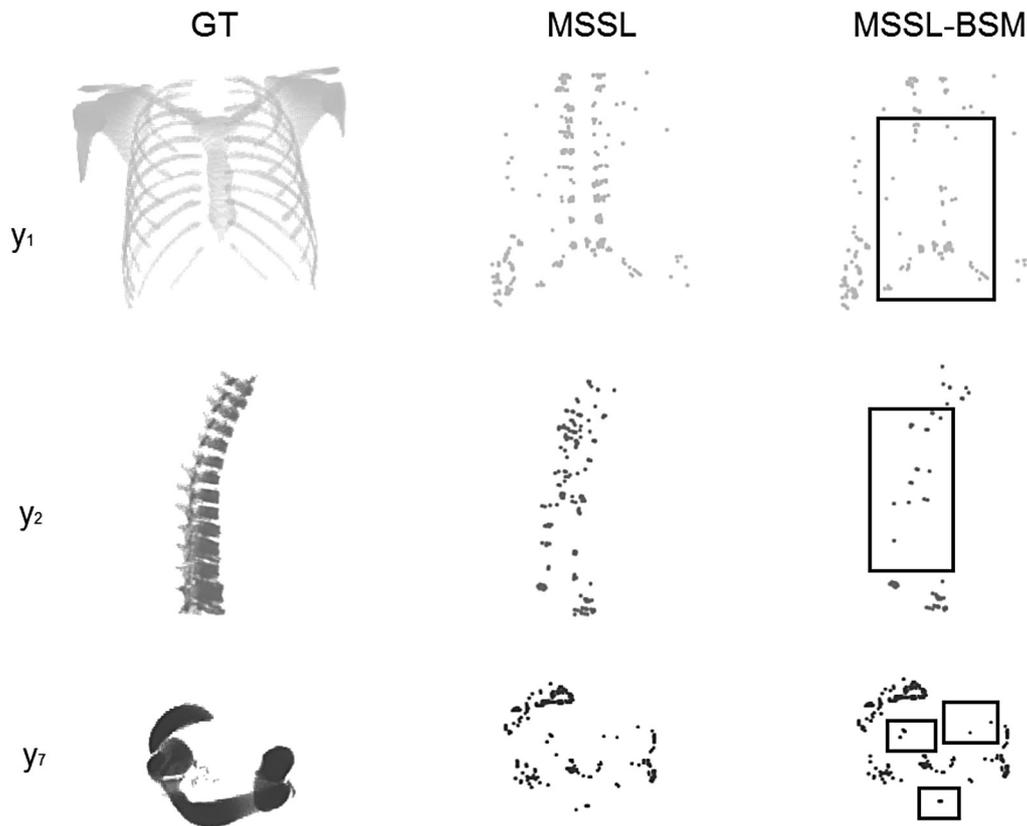


Fig. 10 Visual results on the thorax dataset

Note that in order to visually appreciate the difference, instead of the classified volumes by MSSL and MSSL-BSM only the misclassified voxels are shown for each method

Regions of highest improvement as marked with rectangles in the MSSL-BSM column

Only those labels where the performance improvement is easily appreciated visually are shown (y_1, y_2, y_7)

In particular, we applied the Friedman and Nemenyi statistics in order to look for statistical significance among the obtained performances [20]. In order to compare the performances obtained for each of the three methods on the two datasets, Table 4 shows the mean rank of each method considering the 14 different scores, corresponding to the different label performances. The rankings are obtained by estimating each particular ranking p_i^j for each method i and score j , and computing the mean ranking P for each result as $P_j = (1/Q) \sum_i p_i^j$, where Q is the total number of scores. One can see that, as expected, our method achieves the best rank position, followed by the MSSL approach, and finally by the base classifier results.

To reject the null hypothesis that the measured ranks do not differ from the mean rank, and therefore the ranks are affected by randomness in the results, we use the Friedman test. The Friedman statistic value is computed as follows

$$X_F^2 = \frac{12Q}{k(k+1)} \left[\sum_j P_j^2 - \frac{k(k+1)^2}{4} \right]$$

In our case, for $k=3$ methods to compare, the result for X_F^2 is

Table 4 Mean rank for the three methods on the performed experiments

	Base AdaBoost	MSSL	MSSL-BSM
rank	2.5714	2.2857	1.1429

15.99. Since this value is undesirable conservative, Iman and Davenport proposed a corrected statistic [21]

$$F_F = \frac{(Q-1)X_F^2}{Q(k-1) - X_F^2}$$

Applying this correction we obtain 17.32 as the value of F_F . With 3 methods and 14 scores F_F is distributed according to the F distribution with $(k-1, (k-1)(Q-1)) = (2, 26)$ degrees of freedom. The critical value of $F(2, 26)$ for a 95% of confidence is 3.36. Since the obtained value for F_F is higher than the one obtained by $F(2, 26)$ we can reject the null hypothesis.

Once we have checked for the non-randomness of the results, we can perform a post hoc test to check if one of the methods can be statistically singled out. For this purpose we use the Nemenyi test. The Nemenyi statistic is obtained as follows

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6Q}}$$

In our case, the CD value with a 95% of confidence corresponding to a q_α is 0.88. In order to illustrate the intersections of the CD interval with the method ranks, Fig. 11 plots the corresponding values.

One can see that the rank of our proposal (MSSL-BSM) does not intersect with the other methods rank for the computed CD value. Thus, we can guarantee that our

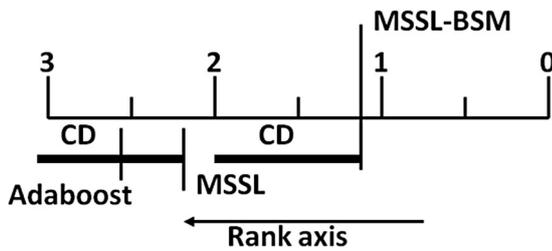


Fig. 11 Nemenyi confidence intervals for comparative results (AdaBoost, MSSL and MSSL-BSM)

achieved results are statistically significant for the performed experiments. On the other hand, the ranks of Adaboost and MSSL approaches intersect for the computed CD value. This means that although MSSL achieves a better rank value (and better performance than Adaboost for most of the test labels), we cannot guarantee that there exists a statistical significance difference between Adaboost and MSSL on the performed experiments. Therefore, we have provided empirical evidence for the motivation of the spatial codification of the predicted labels within the MSSL framework.

4 Conclusion

MSSL has proven to be a successful framework to deal with non-independent identically distributed dataset entries. In this work, we proposed the spatial codification of the label predictions within the MSSL framework as a method for improving its overall performance, in contrast with its original density based codification. In particular, we introduced an adaptation the BSM that provides a smart codification of the predicted labels' spatial distribution, which is used as a descriptor for the new stacked features. In comparison with Adaboost and standard MSSL, our method showed significant performance improvement (with a 95% of confidence) at classifying multiple label structures on two public medical volume segmentation problems. Future work include reducing the running time of our approach by designing appropriate parallel computation schemes optimised for this scenario as well as building more volume segmentation datasets to test the proposed methodology.

5 Acknowledgments

The work of Frederic Sampedro is supported by the Spanish government FPU (Formación del Profesorado Universitario) doctoral grant.

6 References

- 1 Dietterich, T.G.: 'Machine learning for sequential data: a review'. Proc. of the Joint IAPR Int. Workshop on Structural, Syntactic, and Statistical Pattern Recognition, Springer-Verlag, London, UK, 2002, pp. 15–30
- 2 Aas, K., Eikvil, L., Huseby, R.B.: 'Applications of hidden Markov chains in image analysis', *Pattern Recogn.*, 1999, **32**, (4), pp. 703–713
- 3 Davidson, J.L., Cressie, N., Hua, X.: 'Texture synthesis and pattern recognition for partially ordered Markov models', *Pattern Recogn.*, 1999, **32**, (9), pp. 1475–1505
- 4 Cai, J.H., Liu, Z.Q.A.: 'Pattern recognition using Markov random field models', *Pattern Recogn.*, 2002, **35**, (3), pp. 725–733
- 5 Lafferty, J.D., McCallum, A., Pereira, F.: 'Conditional random fields: probabilistic models for segmenting and labeling sequence data'. Proc. of the Int. Conf. on Machine Learning, Williamstown, MA, USA, 2001, pp. 282–289
- 6 Bottou, L., Bengio, Y., Le Cun, Y.: 'Global training of document processing systems using graph transformer networks'. Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 1997, pp. 489–494
- 7 Gatta, C., Puertas, E., Pujol, O.: 'Multi-scale stacked sequential learning', *Pattern Recogn.*, 2011, **44**, (10–11), pp. 2414–2426
- 8 Puertas, E., Escalera, S., Pujol, O.: 'Generalized multi-scale stacked sequential learning for multi-class classification', *Pattern Anal. Appl.*, 2013, pp. 1–15, doi: 10.1007/s10044-013-0333-y
- 9 Escalera, S., Fornés, A., Pujol, O., Lladós, J., Radeva, P.: 'Circular blurred shape model for multiclass symbol recognition', *IEEE Trans. Syst. Man. Cybern. B*, 2011, **41**, (2), pp. 497–506
- 10 Ankerst, M., Kastenmiller, G., Kriegel, H.-P., Seidl, T.: '3d shape histograms for similarity search and classification in spatial databases'. Proc. of the Sixth Int. Symp. on Advances in Spatial Databases, 1999, vol. 28, pp. 207–226
- 11 Escalera, S., Fornés, A., Pujol, O., Radeva, P., Sánchez, G., Lladós, J.: 'Blurred shape model for binary and grey-level symbol recognition', *Pattern Recogn. Lett.*, 2009, **30**, (15), pp. 1424–1433, doi: 10.1016/j.patrec.2009.08.001
- 12 Freund, Y., Schapire, R.: 'A decision-theoretic generalization of on-line learning and an application to boosting', *J. Comput. Syst. Sci.*, 1997, **55**, pp. 119–139
- 13 Linninger, A.A.: 'Biomedical systems research – new perspectives opened by quantitative medical imaging', *Comput. Chem. Eng.*, 2012, **36**, pp. 1–9
- 14 Ribeiro, P.C., Moreno, P., Santos-Victor, J.: 'Introducing fuzzy decision stumps in boosting through the notion of neighbourhood', *IET Comput. Vis.*, 2012, **6**, (3), pp. 214–223
- 15 Sampedro, F., Escalera, S., Domenech, A., Carrió, I.: 'Automatic metabolic tumor volume segmentation in breast cancer whole-body PET/CT scans: a supervised learning approach', *J. Med. Imaging Health Inf.*, 2013. Under revision
- 16 Escalera, S., Tax, D., Pujol, O., Radeva, P., Duin, R.: 'Subclass problem-dependent design of error-correcting output codes', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2008, **30**, (6), pp. 1041–1054
- 17 Escalera, S., Pujol, O., Radeva, P.: 'On the decoding process in ternary error-correcting output codes', *Trans. Pattern Anal. Mach. Intell.*, 2010, **32**, (1), pp. 120–134
- 18 Escalera, S., Puig, A., Amorós, O., Salamó, M.: 'Intelligent GPGPU classification in volume visualization: a framework based on error-correcting output codes', *Comput. Graph. Forum J.*, 2011, **30**, (7), pp. 2107–2115
- 19 Dietterich, T., Bakiri, G.: 'Solving multi-class learning problems via error-correcting output codes', *J. Artif. Intell. Res.*, 1995, **2**, pp. 263–286
- 20 Demsar, J.: 'Statistical comparisons of classifiers over multiple data sets', *JMLR*, 2006, **7**, pp. 1–30
- 21 Iman, R., Davenport, J.: 'Approximations of the critical region of the Friedman statistic', *Commun. Stat.*, 1980, **9**, (6), pp. 571–595