BMC Bioinformatics

Research article

Open Access

A multiresolution approach to automated classification of protein subcellular location images

Amina Chebira*[1], Yann Barbotin[2,4], Charles Jackson[1], Thomas Merryman[2], Gowri Srinivasa[1], Robert F Murphy[1,3] and Jelena Kovačević[1,2]

Address: [1]Center for Bioimage Informatics and Dept. of Biomedical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, [2]Dept. of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, [3]Depts. of Biological Sciences and Machine Learning, Carnegie Mellon University, Pittsburgh, PA, USA and [4]Dept. of Communication Systems, Swiss Federal Institute of Technology, Lausanne, Switzerland

Email: Amina Chebira* - amina@cmu.edu; Yann Barbotin - yann.barbotin@gmail.com; Charles Jackson - charlesjackson@cmu.edu; Thomas Merryman - tmerryman103@yahoo.com; Gowri Srinivasa - gowris@andrew.cmu.edu; Robert F Murphy - murphy@cmu.edu; Jelena Kovačević - jelenak@cmu.edu

* Corresponding author

Published: 19 June 2007

BMC Bioinformatics 2007, **8**:210 doi:10.1186/1471-2105-8-210

Received: 1 February 2007
Accepted: 19 June 2007

This article is available from: http://www.biomedcentral.com/1471-2105/8/210

Abstract

Background: Fluorescence microscopy is widely used to determine the subcellular location of proteins. Efforts to determine location on a proteome-wide basis create a need for automated methods to analyze the resulting images. Over the past ten years, the feasibility of using machine learning methods to recognize all major subcellular location patterns has been convincingly demonstrated, using diverse feature sets and classifiers. On a well-studied data set of 2D HeLa single-cell images, the best performance to date, 91.5%, was obtained by including a set of multiresolution features. This demonstrates the value of multiresolution approaches to this important problem.

Results: We report here a novel approach for the classification of subcellular location patterns by classifying in multiresolution subspaces. Our system is able to work with any feature set and any classifier. It consists of multiresolution (MR) decomposition, followed by feature computation and classification in each MR subspace, yielding local decisions that are then combined into a global decision. With 26 texture features alone and a neural network classifier, we obtained an increase in accuracy on the 2D HeLa data set to 95.3%.

Conclusion: We demonstrate that the space-frequency localized information in the multiresolution subspaces adds significantly to the discriminative power of the system. Moreover, we show that a vastly reduced set of features is sufficient, consisting of our novel modified Haralick texture features. Our proposed system is general, allowing for any combinations of sets of features and any combination of classifiers.

Background

Automated interpretation of protein subcellular location

Among the most important goals in biological sciences today is to understand the function of all proteins. One of the critical characteristics of a protein is its subcellular location, that is, its spatial distribution within the cell. Knowledge of the location of all proteins will be essential for building accurate models that capture and simulate cell behavior, and eventually can be expected to be useful for early diagnosis of disease and/or monitoring of therapeutic effectiveness. The most widely used method for determining protein subcellular location is fluorescence microscopy. Given that mammalian cells are believed to express tens of thousands of proteins, comprehensive analysis of protein location will require acquisition of numbers of images that are beyond our ability to analyze visually.

Fortunately, the feasibility of automated interpretation of subcellular patterns in fluorescence microscope images has been clearly demonstrated over the past ten years, initially by our group [1-3] and then by others [4-6]. The result is systems that can classify protein location patterns with well-characterized reliability and better sensitivity than human observers (for reviews, please see [7,8]). The heart of such systems is a set of numerical features – Subcellular Location Features (SLFs) – to describe the spatial distribution of proteins in each cell image. The SLFs include Haralick texture features, morphological features, and Zernike moments. Of particular relevance to the work described here is that the addition of simple multiresolution features resulted in a significant improvement of classification accuracy, to the highest reported accuracy of 91.5% for the 2D HeLa data set. This dataset contains images of all major subcellular patterns and is a well-established testbed for evaluating subcellular pattern analysis approaches. Note that with the aid of a parallel DNA channel, that accuracy climbed to 92.0%. It is important to have methods that work well when DNA images are available and also when they are not. We focus here on analysis of patterns without parallel DNA images and on improving performance relative to the best previous results.

Multiresolution techniques

As the introduction of the simplest *multiresolution (MR)* features produced a statistically significant jump in classification accuracy, our aim is to explore more sophisticated multiresolution techniques. In particular, the following are the three characteristics of multiresolution [9,10] we wish to explore:

(a) Localization: Fluorescence microscope images have highly localized structures both in space and frequency. This leads us to MR tools, as they have been found to be the most appropriate tools for computing and isolating such localized structures [11].

(b) Adaptivity: Given that we are designing a system to distinguish between classes of proteins, it is clear that an ideal solution is to use adaptive transforms, a property provided by MR techniques. The reasoning is that if there is a different MR transform for each different class, then the transform itself would help in distinguishing the class.

(c) Fast and Efficient Computation: It is well known that MR techniques such as wavelets have a computational cost of the order $O(N)$, where N is the input size, as opposed to $O(N \log N)$ typical for other linear transforms including the FFT. This is one of the major reasons for the phenomenal success of MR techniques in real applications and one of the reasons to incorporate MR features into the system.

MR transforms are many; we now give a brief overview. The basic idea behind MR is that we can look at a signal at different scales or resolutions and at different points in time. This should give us information about hidden structures in the signal, with a particular behavior across scales.

The main building block of any MR transform is a *filter* bank [10]; it is a device that splits the signal into *MR subspaces* (also called subbands, wavelet coefficients or transform coefficients), where each MR subspace contains one part of the signal's spectrum. As an example, in the top part of Figure 1, a two-channel filter bank is given, operating on a 1D signal. For images, on both rows and columns (horizontal and vertical directions), the signal is filtered, followed by downsampling by two (discarding every other sample, allowed because there is filtering beforehand). In the simplest case, this produces four subbands; one extracting lowpass information in both directions, one extracting highpass information in both direction and the remaining two extracting lowpass information in one direction and the highpass information in the other. The example in the lower part of the figure shows such a filter bank applied to each subband again (two levels), as we will use in this paper.

Adaptivity of MR transforms manifests itself in many guises, including a number of popular transforms: (a) Growing a full tree to L levels with specific filters of the same length as the downsampling factor yields the Discrete Fourier Transform (DFT) of size 2^L. (b) Growing a full tree to L levels but allowing the filters to be longer, leads to the Short-Time Fourier Transform, or, the Gabor Transform. (c) Growing the tree only on the lowpass branch to L levels leads to the L-level Discrete Wavelet Transform (DWT). (d) Growing an arbitrary tree leads to Wavelet Packets (WP). (e) Splitting the signal into more

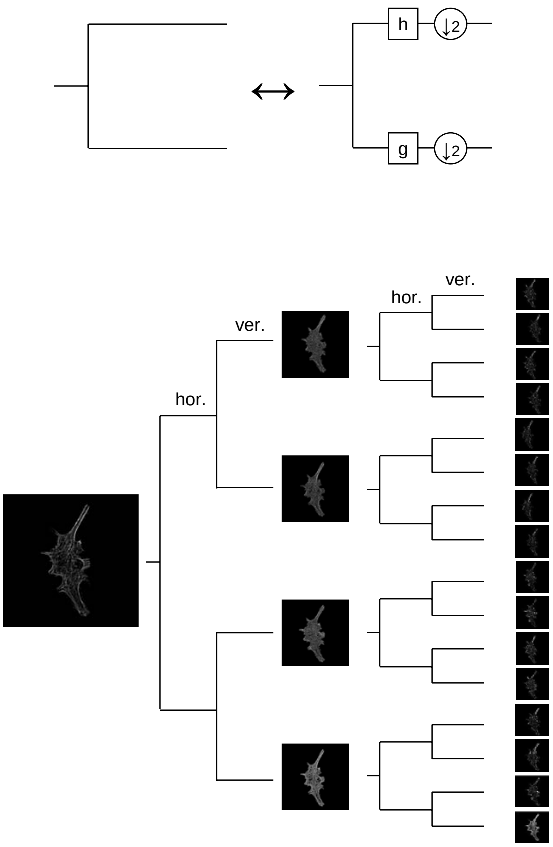

Figure 1
Basic multiresolution block. Top: Two-channel analysis filter bank. The filter *h* is a highpass filter and *g* is a lowpass filter. Bottom: A 2-level filter bank decomposition of actin. If the original image is of size *N* × *N*, the ones in the middle are of sizes *N*/2 × *N*/2 and the ones on the right are of sizes *N*/4 × *N*/4. Each branch has either the lowpass filter *g* or the highpass filter *h* followed by downsampling by 2 as in the top figure. Filtering and sampling are performed along the horizontal direction (rows) followed by the same operations along the vertical direction (columns).

than two channels, allowing filters in the above transforms to be orthogonal and/or linear phase, allowing for true multidimensional filters and/or samplers, etc., leads to even more degrees of freedom.

Towards multiresolution classification

We now summarize our initial MR classification effort [12,13]. We started with a simple classification system consisting of Haralick texture feature computation followed by a maximum-likelihood classifier, and demonstrated that, by adding an MR block in front, we were able to raise the classification accuracy by roughly 10% (from 71.8% to 82.2%) as compared to the system with no MR. This fits within our generic framework shown in Figure 2, where the feature computation block uses Haralick texture features and the classification block is maximum likelihood. We concluded that selecting features in MR subspaces allows us to custom-build discriminative feature sets. However, although the multiresolution block substantially increased classification accuracy, the accuracy of the overall system was still not high enough, and thus, in this work, we reexamined each step of the system: the features used, the classifier, and the weighting process.

Results and discussion

Problem statement and philosophy

The problem we are addressing is that of classifying the spatial distribution patterns of selected proteins within the cell. Assume that the images are of size $N \times N$ and let $\mathbb{R}$ denote the set of intensities covered by all the images in the given dataset, compactly represented as an image belonging to $\mathbb{R}^{N \times N}$. Then, the problem can be formulated as designing a map from the *signal space* of protein localization images $\mathcal{X} \subset \mathbb{R}^{N \times N}$, to a *response space* $\mathcal{Y} \subseteq \{1, 2,..., C\}$ of class labels. Thus, decision d is the map, d: $\mathcal{X} \mapsto \mathcal{Y}$ that associates an input image with a class label [14]. To reduce the dimensionality of the problem, one sets up a feature space $\mathcal{F} \subset \mathbb{R}^f, f \leq N^2$, between the input space and the response space. The feature extractor θ is the map $\theta: \mathcal{X} \mapsto \mathcal{F}$, and the classifier ψ is the map $\psi: \mathcal{F} \mapsto \mathcal{Y}$. The goal is to find a (θ, ψ) pair that maximizes the classification accuracy.

To evaluate MR approaches, we use the well-characterized 2D HeLa set described previously [3]. The proteins in the data set were labeled using immunofluorescence, and thus, we know the ground truth, that is, which protein was labeled in each cell and subsequently imaged. This is useful for algorithm development as we can test the accuracy of classification schemes.

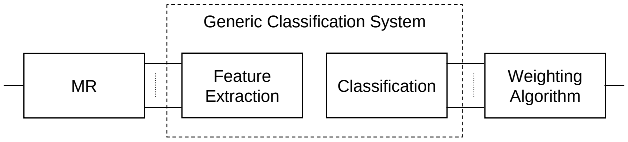

Figure 2
Multiresolution (MR) classification system. The generic classification system (GCS) consists of feature extraction followed by classification (inside the dashed box). We add an MR block in front of GCS and compute features in MR subspaces (subbands). Classification is then performed on each of the subbands yielding local decisions which are then weighed and combined to give a final decision.

The challenge in this data set is that images from the same class may look different while those from different classes may look very similar (see Figure 2 in [13]). Based on the above discussion, we would like to extract discriminative features within space-frequency localized subspaces. These are obtained by MR decomposition; that is, instead of adding MR features as in [15], we compute features in the MR-decomposed subspaces. Thus, our system is a generic system with an MR decomposition block in front (see Figure 2), followed by feature computation and classification in each of the subspaces. These are then combined through a weighting process. The hypothesis we test here is that adaptive classification in MR subspaces will improve the classification accuracy.

Base system (nMR)

We denote as *no MR* (*nMR*) the system consisting of the feature computation and the classifier blocks (see inside the dashed box in Figure 2). In our previous MR work, we used a maximum likelihood classifier that assumed the data to be well-separated Gaussian distributions, an assumption we found not to fit the data well. Instead, due to their simplicity and generality, we decided to use a two-layer neural network classifier. The first layer contains a node for each of the input features, each node using the Tan-Sigmoid transfer function. The second layer contains a node for each output and uses a linear transfer function (no hidden layers are used). We then train the neural network using a one-hot design, that is, since each output from the second layer corresponds to a class, when training, each training image will have an output of 1 for the class of which it is a member and a 0 for all other classes. To maximize the use of our data, our training process of the neural network block uses five-fold cross validation.

We ran the classifier with selected combinations of the three feature types used previously [15] (we did not use wavelet and Gabor features since they are inherently MR). These are: morphological features (M), Haralick texture features (T_1) and Zernike moment features (Z). The results obtained are given in the first row of Table 1. We can see that the most powerful features on their own are texture features T_1, yielding 85.49% classification accuracy. Because of that, we looked into other texture feature sets, such as the second Haralick set from [16] we termed T_2, which produced a slightly better result of 85.76% (second row of Table 1). By examining these feature sets more closely, we developed a novel version of Haralick texture features, termed T_3 (details are given in **Methods**). With this set, the classification accuracy of the nMR system jumped to 87.46% (third row of Table 1). We also ran the experiment for all possible combinations of feature sets, as shown in the first three rows of Table 1. We found that, while the addition of other feature sets to T_3 did not increase the accuracy significantly, it did increase the number of features and thus computational complexity. This "flat" trend will turn out be general, as we will show later.

MR Bases (MRB)

We now implement our main idea of adding an MR block in front of feature computation and classification, as in Figure 2. We start with the MR decomposition being a basis expansion (details are given in **Methods**). We grow a full tree to two levels with Haar filters (see the bottom part of Figure 1). We then test the system with all feature combinations, a neural network classifier as well as two versions of the weighting algorithm (open-form and closed-form, details are given in **Methods**).

Table 1: Classification accuracy per class. *Z*, *M* and *T* stand for Zernike, morphological and texture features.

System	*T*	Weight.	Classification accuracy [%]						
			M	*T*	*Z*	*T, M*	*M, Z*	*T, Z*	All
nMR	T_1	NW	66.12	85.49	51.20	85.76	72.48	85.06	85.04
	T_2	NW	66.12	85.76	51.20	86.64	72.48	85.78	86.24
	T_3	NW	66.12	**87.46**	51.20	87.38	72.48	87.12	86.86
MRB	T_3	OF	81.62	91.82	65.42	92.04	83.38	91.66	92.36
	T_3	CF	81.48	**92.32**	65.84	92.62	83.58	92.34	92.54
MRF	T_3	OF	84.92	94.72	65.82	94.64	86.80	94.74	94.52
	T_3	CF	85.16	**95.26**	65.24	**95.40**	85.88	95.26	**95.38**

T_1 are the original Haralick texture features, T_2 are modified Haralick texture features from [16] and T_3 are our improved texture features. nMR denotes the base system with no MR, MRB denotes MR basis classification and MRF denotes MR frame classification. OF denotes open-form weighting algorithm while CF denotes closed-form weighting algorithm. NW denotes no weighting as there is no MR block in front. Each entry is a number denoting the classification accuracy mean over a number of trials (different orderings of the images) for a given combination of feature sets. Note that the accuracy of nMR with features *M* is the same across the rows T1, T2, T3 since texture features are not involved in the classification when morphological features alone are used (similarly with *Z*, and *M*, *Z*). A subset of these results is shown pictorially in Figure 3. These results should be compared to the best previously obtained result (on the same data set) of 91.5% [15].

The classifier is evaluated using nested cross validations (five-fold cross validation in the neural networks block and ten-fold during the weighting process). One problem with this technique is that the initial ordering of the images determines which images are grouped together for training and testing in each fold of the cross validation. A different original ordering of the images would result in different groupings, which would be equivalent to presenting different data sets to the classifier, and would thus result in a different overall result. We solve this problem by running multiple trials, each with a random initial ordering of the images. The mean result of these trials is taken as our true classification accuracy. In our experiments, we perform ten-fold cross validation on the weight calculation.

We note the following trends: (a) For all feature combinations, MRB significantly outperforms nMR, thus demonstrating that classifying in MR subspaces indeed improves classification accuracy. (b) For the two versions of the weighting algorithm, open form and closed form, the closed-form algorithm slightly outperforms the open-form one for all feature combinations except for *M* alone (fourth and fifth rows of Table 1). In particular, for texture features T_3, the accuracy rose slightly, from 91.82% to 92.32%. (c) While a slightly higher classification accuracy is obtained by using all three feature sets (92.54%) as well as both *T* and *M* (92.62%), the larger number of features and additional complexity of using *M* and *Z* features do not justify the slight improvement in accuracy (texture features T_3 alone achieve 92.32%). As for nMR, this "flat" trend is good news as we can use a significantly reduced feature set and still obtain a fairly high classification accuracy.

While we were satisfied that our hypothesis seems to be true, that is, classifying in MR subspaces increases classification accuracy significantly, we decided to look more closely into how we can improve the system even more. A known issue with MRB is that they are not shift invariant (rather, they are periodically shift invariant). This is due to downsampling used and can create problems as shifted versions of data can lead to different features in MR subspaces.

Our hypothesis is that shifts in the testing set produce reduced classification accuracy. We test this hypothesis by running the algorithm with T_3 features alone and with shifts of $t = 0, 1, 2, 3$ horizontally and vertically in the testing set (these shifts are chosen because we use 2 levels of the MR transform, so it is shift invariant to shifts of 2^2t, but not to shifts of $2^2t + 1$, $2^2t + 2$, $2^2t + 3$). As expected, the classification accuracy drops by 0.22%.

This experiment strongly indicates the use of MR techniques which are shift invariant (or almost shift invariant). These are called *frames* and we examine them next.

MR Frames (MRF)

The simplest MR frame which is completely shift-invariant is called à trous [10] and is obtained by removing downsamplers (which introduce shift variance) from the scheme. This leads to redundancy but avoids the problem of shift variance. The results of the experiments with MR frames (MRF) are given in the last two rows of Table 1 (for the two versions of the weighting algorithm again).

As for MRB, the three trends are similar: (a) MRF outperforms MRB (the only set showing no improvement is *Z* alone). (b) The closed-form algorithm outperforms the open-form one whenever T_3 set is involved. (c) Again, the "flat" trend continues: the difference between using T_3 only as opposed to T_3, *M* or all feature sets is so minor that the added number of features is not worth the complexity. We highlight these trends pictorially in Figure 3.

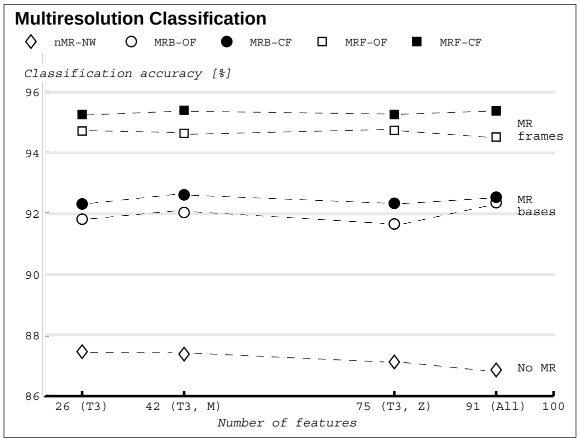

Figure 3

Pictorial representation of classification accuracy results. The diagram shows results from Table 1 for those sets involving T_3, namely (T_3), (T_3, *M*) and (T_3, *M*, *Z*). Diamond markers represent the nMR system (no MR block), circles represent the MRB system (MR bases, no redundancy) and squares represent the MRF system (MR frames, redundancy). Filled markers denote the closed-form weighting algorithm (CF), while empty ones denote the open-form weighting algorithm (OF). The following trends are noteworthy: (a) Introducing MR (both MRB and MRF) significantly outperforms nMR, thus demonstrating that classifying in MR subspaces indeed improves classification accuracy. (b) MRF outperform MRB. (b) For the two versions of the weighting algorithm, open form and closed form, the closed-form algorithm slightly outperforms the open-form one. (d) The trend in each case is almost flat across various feature set combinations, indicating that the texture set T_3 alone (26 features) is sufficient for high classification accuracy.

Discussion and future work

Classification of protein subcellular images was indeed significantly improved by classifying in MR subspaces. One reason for this improved performance over the system using the inherently MR features is that those features are simply energies in the subbands, while here, the features can be any suitable set, leading to a more general space of solutions. A reason for the improved performance of the MR systems over the nMR one could be intuitively understood if we assumed that this data set is highly "texture"-like. For example, it is possible for two different textures to have the same set of Haralick texture features (they have the same co-occurrence matrices), while when decomposed, even at the first level, their co-occurrence matrices would be different, leading to different Haralick texture features, and thus discriminative power. An example of this is given in the compendium to the paper (see Additional file 1 and [17]).

We plan on exploring a number of issues in our future work. (a) For example, our system effectively builds an adapted MR decomposition (via subband weights) for the whole data set; we want to adapt that decomposition to each class, arguing that a different MR decomposition for a different class would be a discriminative feature in itself. We are currently working on this by adapting the closed-form algorithm. (b) We would also like to explore whether improved performance can be obtained by incorporating feature selection methods during classifier training for each subband, as was done in the original work in [15]. (c) It will also be of interest to explore how and whether to include information from parallel DNA images, since this information improved nMR-based classification accuracy in [15] from 91.5% to 92.0%. This improvement is because the parallel DNA image provides a frame of reference for distinguishing proteins that are inside or near the nucleus from those with similar patterns that are not. (d) Lastly, we would like to find a cost function that would allow us to explicitly build wavelet packets. While we implicitly do this now using weights, it would lead to improved computational efficiency if we had a method for building a subtree as opposed to using all the subbands.

Conclusion

This paper addresses automated and robust classification of major protein subcellular location patterns. With the introduction of a multiresolution approach, we are able to obtain a high classification accuracy of 95.26% with only 26 texture features, proving that adaptive MR techniques improve the classification of the 2D HeLa data set.

Methods

Data set

We used the collection of 2D images of HeLa cells described previously [3] and publicly available [18]. It contains approximately 90 single-cell images of size 512 × 512, in each of $C = 10$ classes. The 10 classes of subcellular location patterns were obtained by labeling an endoplasmic reticulum protein, two Golgi proteins (giantin and gpp130), a lysosomal protein, a mitochondrial protein, a nucleolar protein, two cytoskeletal proteins (actin and tubulin), an endosomal protein, and DNA. The best previously described overall classification accuracy on this data set, without the use of the parallel DNA channel, is 91.5% [15].

Base system (nMR)

Feature Sets

As in [15], we start with Haralick texture features (set T_1, 13 features), morphological (set M, 16 features) and Zernike moments (set Z, 49 features). Unlike in [15], we do not use wavelet/Gabor features because the MR advantage given by these will be achieved by our MR decomposition. Therefore, our total number of features is 78, as opposed to 174 in [15].

MR classification

We argued at the beginning of the paper that the nature of our data sets requires tools which offer localization in space and frequency as well as adaptivity, and we further argued that those tools are MR in nature. Thus, the novelty here is classifying in MR subspaces as opposed on the original image itself. The idea is that certain features will react well at a certain scale but not at another. Thus, we add an *MR* block in front of the standard feature extraction and classification blocks, as in Figure 2.

MR block

The basic MR block is the so-called *two-channel filter bank* (see top part of Figure 1). It, and its extensions, can be used to build decompositions custom-tailored to the image at hand. This is done by using this filter bank in a tree, iterating on any of the two-channels and its children. Moreover, the filter bank can have more than two channels, and can have more channels than the sampling factor (leading to redundant representations), etc.

Amongst the possible trees that one can use to analyze an image, the wavelet packets mentioned previously [19] adapt themselves to the signal at hand. However, this is possible only if a suitable *cost function* is available. That is, in order to adaptively build the tree, we need to find a suitable "measure" that will indicate whether a subband (a node in the tree) contains useful information or not. If it does, then we keep the node, otherwise, we prune it.

Adaptive flavors of MR have been explored for their utility in classification in various domains [20]. These studies have used the transform domain coefficients themselves as features and so had a natural cost function in selecting the tree most adapted to the signal. In [21], we used wavelet packets for fingerprint identification and verification with remarkable results.

To get a fairly general set of possibilities MR toolbox offers, we define the following matrix

$$MR_{l,b,m,n} = [\psi_{l,b,o,p}], \tag{1}$$

whose elements $\psi_{l,b,o,p}$ go from $o = 0, ..., m - 1$ and $p = 0, ..., n - 1$. In the above, l denotes the level at which the block is applied, b denotes the particular child branch to which it is applied, m denotes the number of channels and n the sampling factor.

If $m = n$ for every block, the above transforms would implement a basis (nonredundant) expansion. If at least for one block $m > n$, the resulting decomposition is a frame and is redundant. The standard discrete dyadic wavelet transform is obtained with $m = n = 2$ and the MR block being applied only to the first branch of every preceding block.

As an example, we will show how our MR basis with Haar filters is represented. We have $m = n = 2$, and assume that we have a 1D system (a 2D system is obtained by applying the 1D one on rows and then on columns) with 1 level. Thus, $l = 1$ and the block is applied to the input branch only (we number that branch 0). The corresponding matrix is:

$$MR_1 = MR_{1,0,2,2} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{2}$$

where the lowpass filter is given in the first column (roughly averaging) and the highpass is given in the second column (roughly differencing). This matrix describes the operation of the system on the input of length 2. Since the input sequence is in general infinite, the matrix describing the operation of the system on such a sequence is given by

$$MR = \begin{pmatrix} \ddots & \vdots & \vdots & \ddots \\ \cdots & MR_1 & 0 & \cdots \\ \cdots & 0 & MR_1 & \cdots \\ \ddots & \vdots & \vdots & \ddots \end{pmatrix}. \tag{3}$$

The effect of downsampling is seen in the movement of the block MR_1 by 2 each time (if the downsampling were not present, the blocks would be moving by 1).

For $l = 2$ levels, and using the same transform MR_1 at each branch and at each level, we obtain the matrix MR_2:

$$MR_2 = MR_1 \otimes MR_2 = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}, \tag{4}$$

where $\otimes$ denotes the Kronecker product. The matrix describing the operation of the whole system is now an infinite block-diagonal matrix with MR_2 on the diagonal (similarly to (3)).

Feature extraction block

Instead of combining all features into a single probability vector, we allow each feature set its own probability vector per subband. For example, for 2 levels as in (4), we have a total of 21 subbands (original image + 4 subbands at the first level + 16 subbands at the second level), effectively bringing the number of subbands to $3 \cdot S = 3 \cdot 21 = 63$ if all three feature sets are used, where S is the number of subbands per level. Note that although we have decreased the number of features significantly, we have also increased the number of classifiers, because we now have one classifier per subband. Evaluating this computational tradeoff is a task for future work.

New texture feature set T_3

As the Haralick texture features seem to possess the most discriminative power, we looked more closely into these. Haralick texture features are calculated using four cooccurrence matrices [16]: 1) P_H (horizontal nearest neighbors), 2) P_V (vertical nearest neighbors), 3) P_{LD} (left diagonal nearest neighbors), and 4) P_{RD} (right diagonal nearest neighbors). We now calculate 13 measures on each of these four matrices, as defined by Haralick. For example, the first two features on P_H are:

$$f_{H,1} = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g}\left(\frac{P_H(i,j)}{R_H}\right)^2, \tag{5}$$

$$f_{H,2} = \sum_{k=0}^{N_g - 1} k^2 \sum_{|i-j|=k} \frac{P_H(i,j)}{R_H}, \tag{6}$$

where N_g is the number of gray levels in the image and R_H is a normalizing constant equal to the sum of all the elements in P_H.

The other measures follow in similar fashion, giving us four sets of 13 measures: $f_{(H,\ 1-13)}, f_{(V,\ 1-13)}, f_{(LD,\ 1-13)}$ and $f_{(RD,\ 1-13)}$. Haralick's original method reduces these to a single set of 13 by calculating the mean of each measure across the four sets (feature set T_1):

$$f_i^{(T_1)} = \frac{f_{H,i} + f_{V,i} + f_{LD,i} + f_{RD,i}}{4},$$

for $i = 1, \ldots, 13$. An alternative method [16] that we have used previously [22], is to use both the mean and the range of the 13 measures, thus resulting in two sets of 13 features (26 features overall, feature set T_2).

We can significantly improve upon these results by making a change in the way that we combine our initial four sets of features. We note that P_H and P_V are fundamentally different from P_{LD} and P_{HD} because adjacent neighboring pixels are spatially closer than diagonal neighboring pixels. Therefore, instead of averaging the features from all four sets, we create our first set of 13 features by averaging $f_{(H,i)}$ and $f_{(V,i)}$, and a second set of 13 features by averaging $f_{(LD,i)}$ and $f_{(RD,i)}$. Thus, we end up with two sets of 13 features, which are concatenated into one feature set (T_3) of 26 features:

$$f_i^{(T_3)} = \frac{f_{H,i} + f_{V,i}}{2}, \quad f_{i+13}^{(T_3)} = \frac{f_{LD,i} + f_{RD,i}}{2},$$

for $i = 1, \ldots, 13$.

Weighting algorithms

Figure 2 shows a graphical representation of a generic MR classification system, including the process of combining all of the subband decisions into one. We use weights for each subband to adjust the importance that a particular subband has on the overall decision made by the classification system. If the weights are chosen such that the no decomposition weight is equal to 1, and all other weights are 0, we will achieve the same output vector as we would have without using the adaptive MR system. Therefore, we know that there exists a weight combination that will do at least as well as the generic classifier (when no MR is involved) in the training phase. Our goal is to decide how to find the weight vector that achieves the highest overall classification accuracy on a given data set. We developed two versions of the weighting algorithm: open-form and closed-form.

The difference between the open- and closed-form algorithms is that in the open-form version we optimize classification accuracy on the training set as opposed to the closed-form where we look for the least-squares solution. The open-form algorithm for the training and the testing phases are given in [13] under Algorithms 1 and 2, respectively.

The neural network block outputs a series of decision vectors for each subband of each training image. Each decision vector $d_s^{(r)}$ contains $C = 10$ numbers (because we have 10 classes) that correspond to the "local" decisions made by the subband s for a specific image r.

Open-form algorithm

If using the open-form algorithm, we initialize all the weights (see Algorithm 1 in [13] for details), and a global decision vector is computed using a weighted sum of the local decisions. An initial class label will be given to an image using this global decision vector. If that class label is correct, we go to the next image. If it is incorrect, we look at the local decisions of each subband and adjust the weights of each subband s as follows:

$$w_s^{iter} = \begin{cases} w_s^{iter-1} \cdot (1+\varepsilon) & \text{if subband } s \text{ is correct,} \\ w_s^{iter-1} \cdot (1-\varepsilon) & \text{otherwise.} \end{cases}$$

where *iter* is the iteration number and ε is a small positive constant. This can be viewed as a reward/punishment method where the subbands taking the correct decisions will have their weights increased, and those taking wrong decisions will have their weights decreased. We continue cycling through the images until there is no increase in classification accuracy on the training set for a given number of iterations.

Closed-form algorithm

The closed-form solution does not use an iterative algorithm. Instead, it finds the weight vector by solving a minimization problem in the least-square sense. We now explain how this is accomplished.

Assume that we have R training images. For each training image $r = 1, \ldots, R$, the vector $d_s^{(r)} = (d_{s,1}^{(r)}, d_{s,2}^{(r)}, \ldots, d_{s,C}^{(r)})^T$, is the $C \times 1$ decision vector at the output of each subband classifier s, where $d_{s,c}^{(r)}$ indicates the confidence of subband s that the training image r belongs to class c. For each training image r, the weighting block takes as input the subband (local) decision vectors $d_s^{(r)}$ and combines them into a single output decision vector as follows:

$$\sum_{s=1}^{S} w_s d_s^{(r)}.$$

We can rewrite the above by, for each training image r, forming a matrix $D^{(r)}$ of size $C \times S$, where each element

$D_{c,s}^{(r)}$ is the value at position c of the decision vector $d_s^{(r)}$ of subband classifier s. We can now compute:

$$D^{(r)}w,$$

where $w = (w_1, . . ., w_S)^T$ is of size $S \times 1$. Thus, we want to find a weight vector w common to all training images $r = 1, . . ., R$. A possible solution for w is the one that minimizes the error in the least-square sense:

$$w_{win} = \arg\min_{w} \sum_{r=1}^{R} || d^{(r)} - D^{(r)} w ||^2, \qquad (7)$$

where $d^{(r)}$ is the desired target decision vector of size $C \times 1$. It has a 1 in the position of the true class, and 0 elsewhere.

We need to rewrite the above in a direct error-minimization form. We thus define a target output vector d of size $CR \times 1$, as a vector which concatenates all the target decision vectors $d^{(r)}$ as follows:

$$d = \left(\underbrace{d_1^{(1)},\, d_2^{(1)},\, \ldots,\, d_C^{(1)}}_{\text{training image 1}}, \ldots, \underbrace{d_1^{(R)},\, \ldots,\, d_C^{(R)}}_{\text{training image } R} \right)^T,$$

and a $CR \times S$ matrix D consisting of the all the decision matrices $D^{(r)}$ of all the training data stacked on top of each other:

$$D = \begin{pmatrix} D_{1,1}^{(1)} & \cdots & D_{1,S}^{(1)} \\ \vdots & \ddots & \vdots \\ D_{C,1}^{(1)} & \cdots & D_{C,S}^{(1)} \\ \vdots & \ddots & \vdots \\ D_{1,1}^{(R)} & \cdots & D_{1,S}^{(R)} \\ \vdots & \ddots & \vdots \\ D_{C,1}^{(R)} & \cdots & D_{C,S}^{(R)} \end{pmatrix}.$$

We can now rewrite (7) as:

$$w_{win} = \arg\min_{w} \| d - Dw \|,$$

which possesses a closed-form solution and can be computed efficiently.

Then, for a testing image t, we compute its decision vector $\delta = (\delta_1, \delta_2, . . ., \delta_C)$ as follows:

$$\delta = \sum_{s=1}^{S} w_{win,s} d_s^{(t)},$$

where $d_s^{(t)}$ are the local decision vectors for t. The classification decision is then made as

$$c_{win} = \arg\max_{c} \delta_c,$$

that is, the winning class corresponds to the index of the highest coefficient in δ.

MR Bases

Among all possible combinations given in (1), we now confine ourselves to those implementing bases, that is, the resulting decompositions are nonredundant. Thus, in each MR subblock, $m = n$.

We grow a full MR tree with 2 levels. The classification system uses all the subspaces from the root (the original image) to the leaves of the tree. Hence, the total number of subbands used is 21 ($1 + 4 + 4^2$). We used the simplest, Haar filters in the decomposition, where the lowpass is given by $g = \frac{1}{\sqrt{2}}(1,1)^{\mathrm{T}}$ whereas the highpass is $h = \frac{1}{\sqrt{2}}(1,-1)^{>}$. Given a 1D input sequence x, the MR transform we apply to each block of 4 elements (advancing each time by 4) is given by the matrix defined in (4). This is done first in the horizontal direction and then in the vertical one, producing 16 outputs (subbands). There are many other MRB blocks possible, the investigation of which is left for future work.

MR Frames

We now lift the restriction of no redundancy and allow m and n to be different ($m > n$). The resulting decompositions are called frames [23].

We use again the full MR tree with 2 levels, but remove downsamplers, as in the à trous algorithm [10]. Given a 1D input sequence x, the MR transform we apply to each block of 4 elements is identical to the one in (4) except that it is applied to every block of 4 elements (there is no downsampling). There are many other MRF blocks possible, the investigation of which is left for future work.

Reproducible research

All the material needed to reproduce results in this paper is available at the web site address [17] and provided in the Additional file 1 as well.

Authors' contributions

AC coordinated the work, helped conceive the approach and helped extensively with writing the manuscript. YB was in charge of unifying the code and designing and running the experiments involving multiresolution and frames. CJ designed new texture features and helped extensively with writing the manuscript. TM conceived the enhanced multiresolution approach. GS was involved in the original multiresolution approach to classification, the discussions throughout, and helped extensively with writing the manuscript. RFM helped throughout with discussions and editing the manuscript. JK conceived the multiresolution approach and was fully involved in every step of the work, from designing the experiments through interpretation as well as drafting the manuscript. All of the authors read and approved the final manuscript.

Additional material

Additional file 1

Compendium. 07_ChebiraBJMSMK_compendium.zip. This file is a compressed archive that contains the code that generated the results in this paper, the pseudo-code for the weighting algorithms, Table 1 with detailed results and index files of the web site containing all of this material [17].
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-210-S1.zip]

Acknowledgements

We gratefully acknowledge the comments we received from the anonymous reviewers which greatly improved the quality of the manuscript. We are also indebted to our colleagues in the Center for Bioimage Informatics at Carnegie Mellon University for their help and support. This work was supported in part by NSF grants CCF-0515152 and EF-0331657, as well as the PA State Tobacco Settlement, Kamlet-Smith Bioinformatics Grant.

References

1. Boland M, Markey M, Murphy R: **Classification of Protein Localization Patterns Obtained via Fluorescence Light Microscopy.** In *Proc IEEE Int Conf EMBS Society* Chicago, IL; 1997:594-597.
2. Boland M, Markey M, Murphy R: **Automated Recognition of Patterns Characteristic of Subcellular Structures in Fluorescence Microscopy Images.** *Cytometry* 1998, **33:**366-375.
3. Boland M, Murphy R: **A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells.** *Bioinformatics* 2001, **17(12):**1213-1223.
4. Perner P, Perner H, Muller B: **Mining Knowledge for Hep-2 Cell Image Classification.** *Journ Artificial Intelligence in Medicine* 2002, **26(1-2):**161-173.
5. Danckaert A, Gonzalez-Couto E, Bollondi L, Thompson N, Hayes B: **Automated Recognition of Intracellular Organelles in Confocal Microscope Images.** *Traffic* 2002, **3:**66-73.
6. Conrad C, Erfle H, Warnat P, Daigle N, Lorch T, Ellenberg J, Pepperkok R, Eils R: **Automatic identification of subcellular phenotypes on human cell arrays.** *Genome Research* 2004, **14:**1130-1136.
7. Chen X, Velliste M, Murphy R: **Automated Interpretation of Subcellular Patterns in Fluorescence Microscope Images for Location Proteomics.** *Cytometry* 2006, **69 (7):**631-640.
8. Glory E, Murphy R: **Automated Subcellular Location Determination and High Throughput Microscopy.** *Developmental Cell* 2007, **12:**7-16.
9. Mallat S: *A Wavelet Tour of Signal Processing* Academic Press; 1999.
10. Vetterli M, Kovačević J: *Wavelets and Subband Coding* Signal Processing, Englewood Cliffs, NJ: Prentice Hall; 1995.
11. Mallat S: **Wavelets for a vision.** *Proc IEEE* 1996, **33:**604-614.
12. Srinivasa G, Merryman T, Chebira A, Mintos A, Kovačević J: **Adaptive multiresolution techniques for subcellular protein location image classification.** In *Proc IEEE Int Conf Acoust., Speech and Signal Proc Volume V*. Toulouse, France; 2006:1177-1180.
13. Merryman T, Williams K, Srinivasa G, Chebira A, Kovačević J: **A multiresolution enhancement to generic classifiers of subcellular protein location images.** In *Proc. IEEE Int. Symp. Biomed. Imaging* Arlington VA; 2006:570-573.
14. Saito N, Coifman R: **Local discriminant bases and their applications.** *Math Imaging Vision* 1995, **5:**337-358.
15. Huang K, Murphy R: **Boosting accuracy of automated classification of fluorescence microscope images for location proteomics.** *BMC Bioinformatics* 2004, **18(5):**78.
16. Haralick R: **Statistical and structural approaches to texture.** *Proc IEEE* 1979, **67:**786-804.
17. Chebira , *et al.*: **A multiresolution approach to automated classification of protein subcellular location images.** [http://www.andrew.cmu.edu/user/jelenak/Repository/07_ChebiraBJMSMK/07_ChebiraBJMSMK.html].
18. **The Murphy Lab at Carnegie Mellon University** [http://murphylab.web.cmu.edu]
19. Coifman R, Meyer Y, Quake S, Wickerhauser M: **Signal Processing and Compression with Wavelet Packets.** In *Tech rep* Yale University; 1991.
20. Saito N, Coifman R: **Local discriminant bases.** *Proc SPIE Conf. Vis. Commun. and Image Proc* 1994:2-14.
21. Yeomans PH, Thornton J, Kovačević J, Kumar B: **Wavelet packet correlation methods in biometrics.** *Appl Opt, sp iss Biometric Recognition Systems* 2005, **44(5):**637-646.
22. Chen X, Velliste M, Weinstein S, Jarvik J, Murphy R: **Location proteomics – Building subcellular location trees from high resolution 3D fluorescence microscope images of randomly-tagged proteins.** In *Proc SPIE Volume 4962*. San Jose CA; 2003:298-306.
23. Kovačević J, Chebira A: **Life Beyond Bases: The Advent of Frames (Parts I and II).** *IEEE Signal Processing Magazine* 2007.