Spectral Clustering for Identifying Coherent Eddy Structure in Turbulent Boundary Layers

Joanna H. Giforos*, Ellen K. Longmire†, Ivan Marusic†, Nick Hutchins† and Nikolaos P. Papanikolopoulos*
*Dept. of Computer Science & Engineering,
University of Minnesota, Minneapolis, MN 55455
Email: {jgiforos|npapas}@cs.umn.edu
†Dept. of Aerospace Engineering & Mechanics,
University of Minnesota, Minneapolis, MN 55455
Email: {ellen|marusic|hutchins}@cs.umn.edu

***Abstract*— To identify possible vortex packets in color images, we utilize a spectral clustering method. This pairwise segmentation algorithm allows for the use of spectral techniques on large data sets bypassing the normally prohibitive computational costs by employing a numerical solution to the eigenvalue problem known as the Nyström method. Furthermore, spectral clustering utilizes one of the most widely used central grouping clustering algorithms, k-means. Through experimental results, we show that this clustering algorithm does indeed capture the streaks in images. Furthermore, we show that with the proper selection of parameters, the output of the clustering algorithm is almost identical to the original, thereby encoding all color information.**

I. Introduction

In the field of fluid dynamics, the nature of the eddy structure in turbulent flows has been debated for many decades. The eddy structure is potentially important because of its role in, for example, generating surface drag at flow boundaries, mixing hot and cold fluid, transporting trace species or contaminants, and triggering or suppressing flow separation on airfoils. As evident from these examples, the structure of turbulence is important to a vast number of applications. An improved understanding of the eddy structure in wall-bounded flows will allow for the development of accurate numerical models applicable to atmospheric, transport, and vehicular flows which are too complex to simulate directly. In addition, if we are able to understand how eddies contribute to drag, for example, we can develop strategies to control eddy generation and organization in order to reduce drag, and/or promote combustion efficiency. This will have a major impact on many industries including aerospace, transportation, energy, and chemical processing.

Our fundamental understanding of these flows is limited, mainly because of our inability to understand the non-linear interactions that occur between the vortices (or "eddies") which make up the motion across a large range of length and time scales. Due to prohibitive limitations in computational speed and memory, the wide range of eddy scales present in most practical applications (which are associated with a *high* Reynolds number) cannot be simulated completely. Instead, we must rely on the results of numerical simulations at a relatively low Reynolds number as well as experimentation to further our understanding.

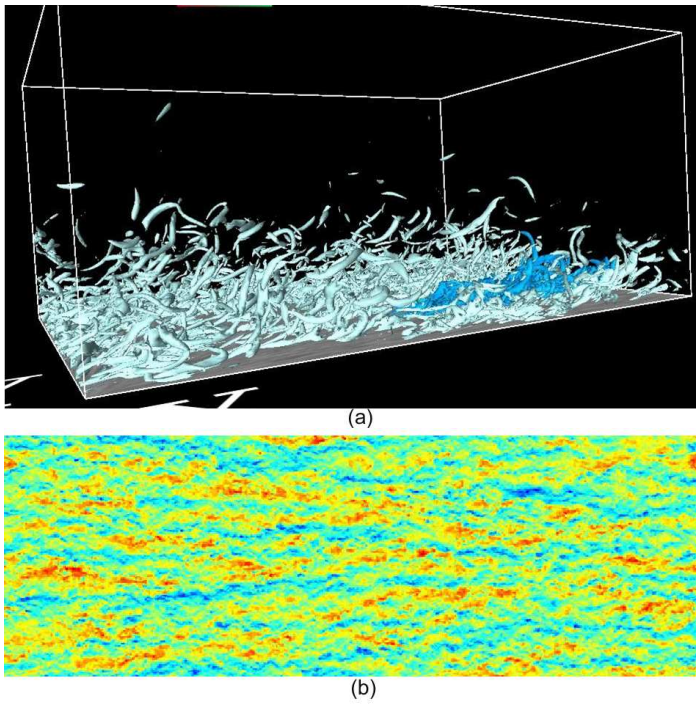

Fig. 1. (a) Direct numerical simulation of turbulent channel flow, individual vortices marked by swirl (b) 2D slice of a 3D image. Direction of flow is from left to right, blue regions indicating slow flow and red regions indicating fast flow.

A key challenge lies in the analysis of the resulting data. With the advent of modern computing and experimental techniques, it has become possible to observe and visualize the eddy structure in relatively simple flows. Instantaneous fields reveal a structure that is highly complex: the quantities of interest are three dimensional, convoluted and of varying size and shape (see Fig. 1).

We seek to identify signatures of superstructures that dominate the energetics and transport within the field. For example, observations of planar fields of velocity vectors

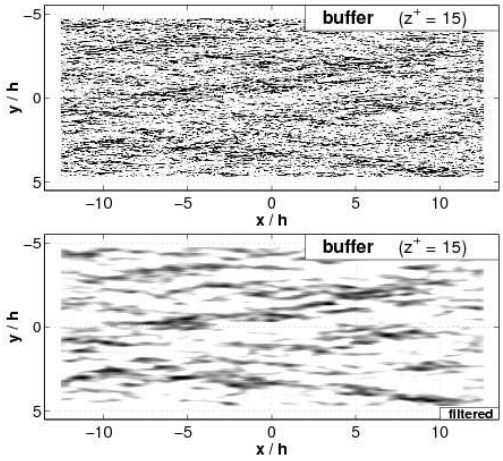

Fig. 2. Upper: Grayscale contours of negative streamwise velocity fluctuation for two streamwise-spanwise planes at $z^+ = 15$ (in near-wall viscous buffer region). Direct Numerical Simulation (DNS) channel data, $Re_\tau = 950$. Lower: Gaussian filtered plot highlight large scale structures ($> O(\delta)$). Note the "footprint" of large-scale structures in the $z^+ = 15$ plane.

and vorticity patterns have led researchers to believe that groups of eddies in turbulent boundary layers tend to travel together in the form of "packets." These packets occur relatively infrequently, however, they are responsible for a large fraction of momentum transport toward and away from the wall. These last conclusions result from an application of a relatively simple algorithm to identify sections of packets intersecting a measurement plane [1]. The algorithm appears reasonably successful, however, it still has significant shortcomings. First, we believe that it underestimates the streamwise extent of packets by failing to connect related substructures (groups of vortices). Second, it may underpredict the number or extent of packets by failing to recognize the full range of eddy scales within a given packet. Third, it does not make use of more comprehensive data (e.g., the full vorticity vector instead of one vorticity component) made available using a novel optical technique developed recently in our laboratory and available in simulated data.

The result in Fig. 2 is actually the main motivation behind this paper. For the turbulence researchers it was obvious that the superstructures could be identified by eye (albeit rather qualitatively and crudely), and it was also very clear that an objective algorithm-based methodology was needed to extract and quantify these important structures in a meaningful way, particularly when any significant analysis will require the "inspection" of thousands of realizations. Furthermore, Fig. 2 shows only one variable, while many other important variables need to also be considered for a complete physical understanding.

Our approach emphasizes the use of pattern recognition algorithms for the extraction of interesting underlying structures in turbulent flows. As shown in Fig. 1, the regions of concern are the dark blue and red colored streaks shown in the 2D image. Once these regions have been segmented we can then use region growing techniques to further segment the regions of interest and obtain a clearer understanding of their structure.

The organization of this paper is as follows. We begin by giving a brief overview of clustering methods in the literature and some previous attempts at identifying eddy structures. Next, we review spectral clustering using the Nyström extension in Section III. Experimental results and discussion are provided in Section IV and finally we conclude and provide future work direction in Section V.

II. Literature review

A great challenge with the analysis of planar velocity fields from Particle Image Velocimetry (PIV) or volumetric velocity and pressure fields from Direct Numerical Simulation (DNS) is how to identify and understand the dynamics captured within the very large and complex data sets. For example, one time step of data from a turbulent channel simulation requires 9 GB of memory [2]. A number of studies have used one of several schemes based on manipulation of the velocity gradient tensor (e.g., Jeong and Hussein [3], Zhou *et al.* [4]) to identify vortex cores within velocity fields. In these studies, vortex cores are marked typically by an isosurface, and individual velocity or vorticity fields are examined visually to assess the impact or effect of the cores on the surrounding flow dynamics. In addition, pattern recognition schemes [5] and wavelet analysis [6] have been employed to investigate the presence and probability distribution of vortex cores respectively in a more automated manner. Furthermore, Proper Orthogonal Decomposition (POD) has been used to characterize a dominant eddy structure in simulated turbulent channel flow [7]. By contrast, we are aware of very little work attempting to identify quantitatively the presence of larger superstructures that may contain multiple vortices or may result from the influence of multiple surrounding vortices.

Clustering techniques can be classified into one of the two following two categories: pairwise and central groupings. In central grouping methods, all data points are compared to k central clusters. In the well known k-means algorithm [16] for example, the goal is to segment the data so that the within-cluster squared Euclidean distance is minimized. Alternatively, pairwise clustering techniques measure similarities or dissimilarities between pairs of data points. In general, pairwise techniques cluster based on the eigenvectors of a matrix derived from the data. For the purpose of image segmentation, the pairwise techniques have shown much promise.

Recently, there has been a surge in finding alternative clustering techniques due to their wide use within the pattern recognition and data mining domains. Many researchers have altered the original k-means algorithm so as to address some of its inherent drawbacks. In [8], Banerjee et al. provide a more general k-means type algorithm which uses any Bregman divergence as the distortion measure.

Charalampidis [9] recently suggested a modified k-means algorithm for clustering vectors containing directional information. In [10], Su and Chou use the idea of point symmetry to motivate a new non-metric distance measure.

One of the main drawbacks of central grouping techniques such as k-means and *EM* clustering with Gaussian mixture models is that they require model selection. In high dimensional space the use of the squared Euclidean distance as a loss function is not meaningful. Yet, without knowing the underlying distribution of the data, it may be difficult to find either kernel k-means techniques or those which utilize different loss function such as in [8]. Pairwise clustering algorithms do not suffer from such setbacks.

K-means type methods assume that the data are linearly separable. For most applications involving high dimensional data, this requirement is either not known or infeasible. This restriction has required researchers to construct new methods for the separation of non-linearly separable data. One such method is kernel k-means, where the data are projected into higher dimensional space and linear separators are used in this new space. The second method is spectral clustering where data is projected along the principle component axes of a matrix derived from the data and linearly separated in that space. These spectral methods have proven promising for use in image segmentation.

III. Spectral Clustering using Nystrom Approximation to NCut

Spectral clustering has its origin in spectral graph partitioning. Most algorithms use the eigenvectors of the Laplacian of the graph adjacency (pairwise similarity) matrix to find the optimal graph partitioning. To identify possible vortex packets, we utilized the spectral clustering algorithm presented in [11]. This pairwise segmentation algorithm allows for the use of spectral techniques on large data sets bypassing the normally prohibitive computational costs by employing a numerical solution to the eigenvalue problem known as the Nyström method. Furthermore, spectral clustering utilizes one of the most widely used central grouping clustering algorithms, k-means. We will begin our discussion with a brief outline of the Normalized Cut (NCut) [12] followed by the Nyström extension application to NCut [11]. We then outline the spectral clustering algorithm within this context and give a brief discussion of kernel selection. We conclude this section with a synopsis of the k-means algorithm.

A. Normalized Cuts

We begin with a set of points $X = \{x_1, x_2, ..., x_n\}$ in $\mathcal{R}^l$ that we want to cluster into k subsets and a weighted adjacency matrix $W \in \mathcal{R}^{N \times N}$ for a graph $G = (V, E)$ where V are the nodes and E are the edges. The function used to calculate W is known as the *kernel*; we will discuss the selection of an appropriate kernel in Section III-D. If the matrices A and B represent the bipartition of V, i.e., $A \cup B = V$ and $A \cap B = \emptyset$, then the degree of dissimilarity between these two matrices can be computed as the total weight of the edges that have been removed, i.e., cut(A,B)= $\sum_{i \in A, j \in B} W_{ij}$. The degree of the *i*th node is defined as the sum over the column of the weighted adjacency matrix, i.e., $d_i = \sum_j W_{ij}$ and the volume of a set is the sum of the degrees within that set, $vol(A) = \sum_{i \in A} d_i$. We define the *Normalized Cut (Ncut)* as the fraction of the total edge connections to all the nodes in the graph:

$$\begin{aligned} Ncut(A,B) &= cut(A,B)\left(\frac{1}{vol(A)} + \frac{1}{vol(B)}\right) \\ &= \frac{2cut(A,B)}{vol(A)||vol(B)} \end{aligned}$$

where $||$ denotes the harmonic mean.

The optimal bipartitioning of a graph is the one that minimizes this cut value. In [12], Shi and Malik show that this minimization can be formulated as a generalized eigenvalue problem. An approximate solution is obtained by thresholding the eigenvector corresponding to the second smallest eigenvalue of the normalized Laplacian:

$$\begin{aligned} \mathcal{L} &= D^{-1/2}(D - W)D^{-1/2} \\ &= I - D^{-1/2}WD^{-1/2} \end{aligned} \quad (1)$$

where D is the diagonal matrix $D_{ii} = d_i$ and I is the identity matrix. It can be shown that the matrix $\mathcal{L}$ is positive semi-definite for all matrices W and its eigenvalues lie in the interval $[0, 2]$, therefore the eigenvalues of $D^{-1/2}WD^{-1/2}$ are confined to $[-1, 1]$.

B. Nyström Approximation to Ncut

For color images, finding the eigenvectors of the normalized Laplacian matrix $\mathcal{L}$ (Eq. (1)) is computationally expensive. One approach to reduce the complexity of this problem is to threshold the number of connections per pixel, thereby restricting the number of pairs considered. This allows for the use of efficient sparse representations, yet may result in oversegmentation of homogeneous regions since it discourages the use of long-range connections. Fowlkes *et al.* [11] present an approximation technique as an alternative method based on random sampling, which is the method we present here.

Initially, m samples are initially chosen at random from the set of N pixels. If we reorder the pixels so that the m samples come first and the remaining $n = N - m$ come next, we can partition the weighted adjacency matrix W as:

$$W = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \quad (2)$$

where $A \in \mathcal{R}^{m \times m}$, $B \in \mathcal{R}^{m \times n}$, $C \in \mathcal{R}^{n \times n}$, and $N = m + n$, with $m \ll n$. Here, A represents the subblock of weights among the random samples, B contains the weights from the random samples to the rest of the pixels and C contains the weights between the remaining pixels. Since $m \ll n$, C is a very large matrix. To overcome this problem, the Nyström approximation implicitly approximates C using

TABLE I

SPECTRAL CLUSTERING ALGORITHM USING NYSTRÖM APPROXIMATION TO NCUT

given $X = x_1, \ldots, x_n$
begin initialize *niter*, k, σ, *nsamp* (number of sample points),
do
select *nsamp* points at random
compute affinities of sampled and remaining points
A =exp$(-\chi^2/2\sigma^2)$ (affinity between sampled points)
B =exp$(-\chi^2/2\sigma^2)$ (affinity between remaining points)
compute connection weight $\hat{\mathbf{d}}$ from Eq. (5)
normalize A and B (Eqs. (6) and (7))
find eigenvectors (V) using Nyström approximation (Eq. (4))
choose k eigenvectors corresponding to the k largest eigenvalues
normalize eigenvectors so that $V^T V = I$ (matrix $\hat{V}$)
cluster $\hat{V}$ using k-means algorithm (see Tbl. II)
point x_i is assigned to cluster j if and only if
row i of $\hat{V}$ is assigned to cluster j
end

$C = B^T A^{-1} B$. The quality of the approximation of the full weight matrix

$$\hat{W} = \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix} \tag{3}$$

can be quantified as the norm of the Schur complement $||C - B^T A^{-1} B||$. The size of this norm is governed by the extent to which C is spanned by the rows of B giving an approximation to the entire weight matrix based on a subset of rows/columns.

$\hat{W}$ can be diagonalized in the following manner. If A is positive definite (the selection of kernel can guarantee the positive definiteness of A, see Section III-D) and $A^{1/2}$ is the square root matrix of A, we define a new matrix $S = A + A^{-1/2} B B^T A^{-1/2}$ and diagonalize S as $S = U \Lambda U^T$. It can be shown [11] that $\hat{W}$ is diagonalized by V and Λ, i.e. $\hat{W} = V \Lambda V^T$ where

$$V = \begin{bmatrix} A \\ B^T \end{bmatrix} A^{-1/2} U \Lambda^{-1/2} \tag{4}$$

and V is orthonormal i.e., $V^T V = I$. If there is redundancy in the random samples, pseudoinverses may be used in place of inverses.

We can now apply this approximation to Ncut by first computing the row sums of $\hat{W}$. Instead of explicitly evaluating the last block in Eq. (3) this can be done using,

$$\begin{aligned} \hat{\mathbf{d}} = \hat{W}\mathbf{1} &= \begin{bmatrix} A\mathbf{1}_m + B\mathbf{1}_n \\ B^T\mathbf{1}_m + B^T A^{-1} B \mathbf{1}_n \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{a}_r + \mathbf{b}_r \\ \mathbf{b}_c + B^T A^{-1} \mathbf{b}_r \end{bmatrix} \end{aligned} \tag{5}$$

where $\mathbf{b}_c \in \mathcal{R}^n$ denotes the column sum of B and $\mathbf{a}_r, \mathbf{b}_r \in \mathcal{R}^m$ denote the row sums of A and B, respectively and $\mathbf{1}$ denotes a column vector of ones (the subscript indicates the dimension of this vector).

Given $\hat{\mathbf{d}}$, we can now evaluate the blocks of $\hat{D}^{-1/2} \hat{W} \hat{D}^{-1/2}$, twhich are necessary for approximating the leading eigenvectors using

$$A_{ij} \leftarrow \frac{A_{ij}}{\sqrt{\hat{\mathbf{d}}_i \hat{\mathbf{d}}_j}}, \qquad i, j = 1, \ldots, m \tag{6}$$

and

$$B_{ij} \leftarrow \frac{B_{ij}}{\sqrt{\hat{\mathbf{d}}_i \hat{\mathbf{d}}_{j+m}}}, \quad i = 1, \ldots, m \ , \ j = 1, \ldots, n \tag{7}$$

These can then be applied to Eq. (4) for computing the approximate eigenvalues of the matrix $I - \hat{D}^{-1/2} \hat{W} \hat{D}^{-1/2}$.

TABLE II

K-MEANS CLUSTERING ALGORITHM

begin initialize n, k, $\mu_1, \mu_2, \ldots, \mu_k$
do classify n samples according to nearest μ_j
recompute μ_j
until no change in μ_j
return $\mu_1, \mu_2, \ldots, \mu_k$
end

C. Spectral Clustering Algorithm

Once we have an approximate solution to the eigenvector problem, we can proceed with the spectral clustering algorithm as presented in [13]. Specifically, the k eigenvectors (matrix V in Eq. (4)) associated with the k largest eigenvalues are normalized to form the new matrix $\hat{V}$. Treating each row of $\hat{V}$ as a point in $\mathcal{R}^k$, we cluster them into k clusters via the k-means algorithm (see Section III-E). For the case where $k = 2$, k-means is not necessary as we can simply threshold the eigenvectors and obtain a two-way partition. Finally, the original point x_i is assigned to cluster j if and only if row i of the matrix $\hat{V}$ was assigned to cluster j. Table I summarizes the algorithm.

D. Kernel Selection

In Section III-B we mentioned that the matrix A must be positive definite in order to calculate the approximation. The positive definiteness of the matrix A depends on the selection of kernel. Choosing the appropriate kernel for a given application is an important and well-studied problem. Within the context of color image segmentation, it has been shown that using a kernel which utilizes histogram information is a robust measure for texture and color separation [14]. One such kernel is the Red, Green and Blue (RGB) histogram comparison using the χ^2 test.

Given two normalized RGB histograms $h_i(k)$ and $h_j(k)$, we can define:

$$\chi^2_{ij} = \frac{1}{2} \sum_{k=1}^{K} \frac{(h_i(k) - h_j(k))^2}{h_i(k) + h_j(k)} \ .$$

Any term in the sum for which $h_i(k) = 0$ and $h_j(k) = 0$ is set to zero. Applying the χ^2 test, we can define the affinity between a pair of histograms as W_{ij} =exp$^{(-\chi^2/2\sigma^2)}$. This kernel is positive definite, as shown in [15], making the Nyström approximation as outlined in Eq. (4) feasible.

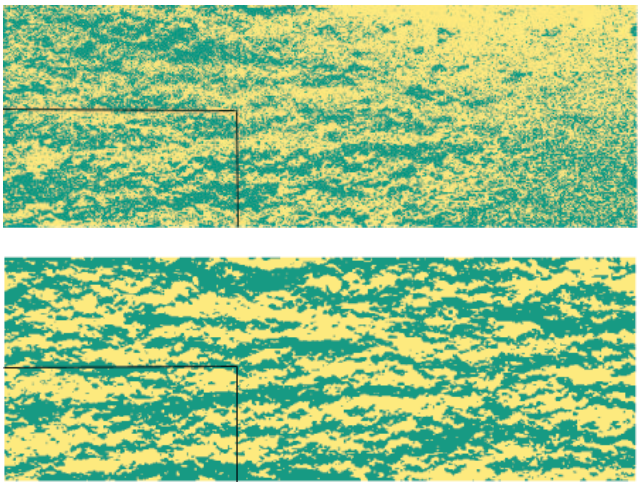

Fig. 3. Results of the SC algorithm on the original image. Top image corresponds to the case where k=2 and σ=500, and the bottom image k=2 and σ=100.

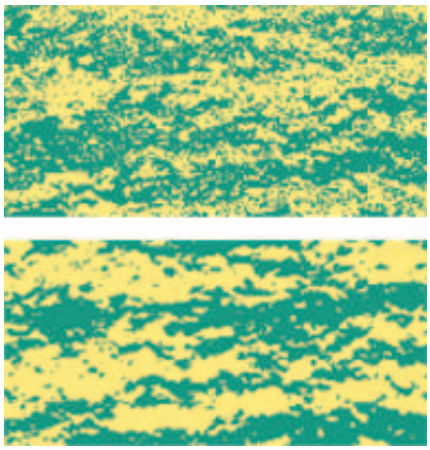

Fig. 4. (Sections of the images in Fig. 3) Results of the spectral clustering algorithm on the original image. Top image corresponds to the case where k=2 and σ=500, and the bottom image k=2 and σ=100.

E. K-Means

One of the most widely used clustering algorithms within the pattern recognition community is k-means. The goal of k-means is to segment a given image into k clusters so that the within-cluster sum of squares is minimized. Specifically, n data points are clustered into k subsets, k_j, containing n_j data points so as to minimize $\sum_{j=1}^{k} \sum_{n\in\ k_j} \mid x_n - \mu_j \mid^2$ where x_n is the nth data point and μ_j is the mean of cluster j.

The algorithm begins by initializing the k-means. If no information about the data is known *a priori*, we choose a random initialization. The algorithm then proceeds to iteratively minimize the aforementioned summation through the following two procedures: (*i*) *Assignment step:* Assign pixels to one of the k clusters. In the case of images, we assign an object to the cluster whose pixel value is the closest and (*ii*) *Re-estimation step:* Calculate the new group means based on the assignments. The process terminates when no movement of an object to another group will reduce the within-group sum of squares. Table II adapted from [16] summarizes the algorithm.

IV. Experimental Results

The images in Fig. 1 are attained from Direct Numerical Simulation (DNS) of experimental dual plane Particle Image Velocimetry (PIV) data. The PIV dataset was obtained from experiments conducted in a suction-type boundary layer wind tunnel by [17]. A three-camera polarization based dual plane PIV system was used. Here, two independent PIV systems captured images of olive oil droplets of size $\sim 1\mu m$. In the first of the two systems, a stereoscopic setup was used, which provided all three velocity components within a plane. In the second system, a conventional PIV system was utilized. In this setup, in-plane velocity components were measured in a neighboring plane located 21 wall units above in the wall normal direction. To isolate one camera system from another, the polarization property of laser light sheets was used to capture simultaneous measurements see [17] and [18]. Furthermore, this study is performed in the logarithmic region of a turbulent boundary layer located 110 wall units from the wall.

The DNS dataset we use in this paper is a fully developed channel flow performed by del Alamo [2]. While this simulation is not an exact replication of the PIV dataset, under certain assumptions, it has been shown to be a good approximation. The simulation of interest in the present study has $Re_\tau = 934$, which is referred to as L950. The computational domain in this simulation was $8\pi h$ units in the streamwise direction and $3\pi h$ units in the spanwise direction, where h is the channel half width. The sizes of the domain were set to account for all the energy containing structures in the flow, especially features with dimensions of the order of h. The spacing between the vectors was 11.46×5.73 wall units in the streamwise and spanwise directions respectively after de-aliasing.

Fig. 1b shows the image we applied the Spectral Clustering (SC) algorithm to. The original image is 213×604. Our goal is to have all gradations of blue in one cluster and all gradations of red in another. Fig. 3 shows the output of the SC algorithm on the original 2-dimensional image shown in Fig. 1b. Here the green and yellow regions correspond to the blue and red regions, respectively. In the top image, the number of clusters, k, is 2 and σ=500. In this case, it is clear that σ is too large. σ is a measure of when two pixels are considered similar. As σ grows so do the dissimilarities between pixel values therefore accentuating the differences. The result is that for large values of σ, we have many small "islands" of isolated pixels. Upon reducing σ, we have more smooth results, (see bottom image where σ=100) which is desirable for this application. This becomes more evident in Fig. 4, where have have zoomed a small region of the clustered output. Clearly, it will be easier to utilize region growing methods to detect the length of the streams in the

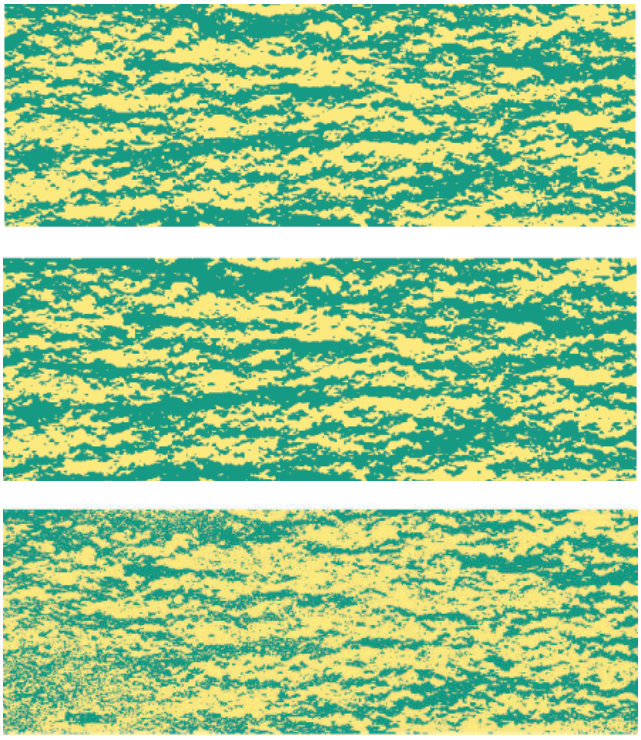

Fig. 5. Effects of the number of samples used in the eigenvalue approximation when k=2 and σ=100. Top image corresponds to the case where the number of samples is 100, in the middle image the number of samples is 115, and in the bottom image the number of samples is 75.

image with the lower σ value.

Fig. 5 shows the effects of choosing a different number of samples for which we apply the Nyström approximation. In [11], the authors suggest that 100 randomly chosen samples are enough to capture salient groups in natural images. In the present study, the image is not natural but synthetic. As shown in Fig. 5, although we have achieved slightly better results for the case where the number of samples is 115, in practice and for the purpose of our application, this may not be as significant when one considers the additional computation time (66.250 cpu seconds for 100 samples compared to 130.188s for 115 samples). As expected, when the number of samples is reduced to 75, there is a significant decrease in performance.

V. Conclusions and Future Work

In this paper, we have applied spectral clustering using the Nyström method, as presented in [11], to the problem of identifying coherent eddy structure in turbulent boundary layers. Using a 2-dimensional image from the DNS dataset, we have shown that with the proper selection of σ, this algorithm captures most of the underlying structure. Furthermore, we have studied the effects of selecting a smaller and larger number of samples and have concluded that 100 works well for this application in terms of both computation time and accuracy.

We intend to use this clustering to aid in the development of region growing methods to extract "regions of interest" autonomously. This can be done through an interactive tool, where a user will initialize the process by selecting a small region of the clustered image to be expanded.

Acknowledgements

The authors would like to thank Prof. Robert Moser for providing the DNS data for the channel flow. Support from the University of Minnesota, Digital Technology Center is gratefully acknowledged.

References

[1] E.K. Longmire, B. Ganapathisubramani, I. Marusic, T. Urness, and V. Interrante, "Effective Visualization of Stereo Particle Image Velocimetry Vector Fields of a Turbulent Boundary Layer," *Journal of Turbulence*, vol. 4, no. 24, 2003.

[2] J.C. Del Alamo, J. Jimenez, P. Zandonade, and R.D. Moser, "Scaling of the Energy Spectra of Turbulent Channels," *Journal of Fluid Mechanics*, vol. 500, pp. 135–144, 2004.

[3] J. Jeong and F. Hussain, "On the Identification of a Vortex," *Journal of Fluid Mechanics*, vol. 285, pp. 69, 1995.

[4] J. Zhou, R. J. Adrian, S. Balachandar, and T. M. Kendall, "Mechanisms for Generating Coherent Packets of Hairpin Vortices in Channel Flow," *Journal of Fluid Mechanics*, , no. 387, pp. 353–396, 1999.

[5] F. Scarano, C. Benocci, and M. L. Riethmuller, "Pattern Recognition Analysis of the Turbulent Flow Past a Backward Facing Step," *Physics of Fluids*, vol. 11, no. 12, pp. 3808–3818, 1999.

[6] A. Siegel and J. B. Weiss, "A Wavelet-Packet Census Algorithm for Calculating Vortex Statistics," *Physics of Fluids*, vol. 9, no. 7, pp. 1988–1999, 1997.

[7] P. Moin and R.D. Moser, "Characteristic Eddy Decomposition of Turbulence in a Channel," *Journal of Fluid Mechanics*, vol. 200, pp. 471–509, 1989.

[8] A. Banerjee, S. Merugu, I.S. Dhillon, and J. Ghosh, "Clustering with Bregman Divergences," *Journal of Machine Learning Research*, vol. 6, pp. 1705–1749, 2005.

[9] D. Charalampidis, "A Modified K-Means Algorithm for Circular Invariant Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1856–1865, December 2005.

[10] M.C. Su and C.-H. Chou, "A Modified Version of the K-Means Algorithm with a Distance Based on Cluster Symmetry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 674–680, June 2001.

[11] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral Grouping Using the Nyström Method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 214–225, February 2004.

[12] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, August 2000.

[13] A. Ng, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," in *Proceedings of Neural Information Processing Systems (NIPS 2002)*, 2002, pp. 849–856.

[14] J. Puzicha, T. Hoffman, and J. Buhmann, "Non-Parametric Similarity Measures for Unsupervised Texture Segmentationand Image Retrieval," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 1997, pp. 267–272.

[15] S. Belongie, C. Fowlkes, F. Chung, and J. Malik, "Spectral Partitioning with Indefinite Kernels Using the Nyström Extension," in *European Conference on Computer Vision*, Copenhagen, Denmark, 27 May - 2 June 2002, pp. 531–542.

[16] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, John Wiley and Sons, 2001.

[17] B. Ganapathisubramani, *Investigation of Turbulent Boundary Layer Structure using Stereoscopic Particle Image Velocimetry*, Ph.D. thesis, University of Minnesota, USA, 2004.

[18] C.J. Käler, "Investigation of the Spatiotemporal Flow Structure in the Buffer Region of a Turbulent Boundary Layer by means of Multiple Plane Stereo PIV," *Exp. Fluids*, vol. 36, pp. 114–130, 2004.