2010

Development of a Spectral 2-D Fast Fourier Transform Analysis for Sand Ripple Morphology Interpretation

Christopher M. Englert
Colby College

DEVELOPMENT OF A SPECTRAL 2-D FAST FOURIER TRANSFORM ANALYSIS FOR SAND RIPPLE MORPHOLOGY INTERPRETATION

Christopher M. Englert '10

An Honors Thesis

Submitted to the Faculty of the Geology Department of
Colby College in Fulfillment of the Requirements for
Honors in Geology

Waterville, Maine

May 17, 2010

DEVELOPMENT OF A SPECTRAL 2-D FAST FOURIER TRANSFORM ANALYSIS FOR SAND RIPPLE MORPHOLOGY INTERPRETATION

Except where reference is made to the work of others, the work described in this thesis is my own or was done in collaboration with my advisory committee

_____

Christopher M. Englert '10

Certificate of Approval:

_____	_____
Dr. Robert A. Gastaldo, Chairman Whipple Coddington Professor Department of Geology, Colby College	Dr. Arthur C. Trembanis Assistant Professor Department of Geology, Univ. of Delaware
_____	_____
Dr. Bruce F. Rueger Visiting Assistant Professor Department of Geology, Colby College	Dr. Bruce Maxwell Associate Professor Computer Science, Colby College

ABSTRACT

A MATLAB-based 2-Dimensional Fast Fourier Transform spectral analysis is developed and tested for its applicability to bedform morphology and hydrodynamic data from a hydrodynamically complex field site. The aspects of the procedure tested are (1) the use of an 8 sub-region arrangement for analysis versus five other arrangements, (2) the procedure for averaging the wavelength-and-orientation outputs from each sub-region to produce a single representative value for each image, (3) methods for eliminating and/or filtering poor quality data, and (4) the procedural steps in the spectral analysis that can appropriately incorporate automation versus manual involvement and interpretation. The automated technique is limited in its ability to adapt to the dynamic data set, and recommendations towards different procedures are made for future studies intending to use similar spectral analysis tools. Most notably, the applicability of the spectral analysis is limited by the bedform morphology and hydrodynamic conditions, requiring preliminary study of data sets prior to analysis by a 2D FFT program.

ACKNOWLEDGEMENTS

I'd like to acknowledge the National Science Foundation for funding the Research Experience for Undergraduates at the University of Delaware that was the beginning for this research. I'd like to thank the advisors I had during this project, Dr. Robert Gastaldo for substantial comments and aid during the thesis process, and Dr. Art Trembanis for suggestions and comments on the development of the project; Committee members Dr. Bruce Maxwell and Dr. Bruce Reuger; my good friend and third advisor Adam Skarke, as well as the rest of Dr. Trembanis' lab; and finally a colleague at Colby College, Drew Vartanian, for help in aspects of my code and discussion on development. I'd also like to thank the Dean of Faculty and the Department of Geology at Colby College for awards to attend the Geological Society of America's national meeting in Portland, Oregon, and again Dr. Art Trembanis and the University of Delaware for funds to attend the Ocean Sciences meeting in Portland, Oregon.

TABLE OF CONTENTS

1. Introduction

Prediction of bedform morphology evolution over time has become increasingly important to all users of the coastal environment. Bedform morphology is the product of sediment transport, or the movement of sediment due to the influences of flow, with critical impacts to coastal and marine processes including the carrying of pesticides, radioactive materials, and nutrients; infilling and migration of shipping channels; and biogeochemical interactions in the continental shelf ecosystem (Davies and Thorne, 2008; Voulgaris and Morin, 2008, Brunsden, 2002). These impacts are wide-felt, as our world currently finds a growing human population living along the coasts, increasing economic sectors tied directly to coastal productivity (in fisheries, tourism, trade, etc.), and a mounting threat of sea-level rise due to climate change (Interagency Ocean Policy Task Force, 2009). An increased understanding of sediment transport in the coastal environment is crucial to predicting future impacts on coastal communities.

In efforts to characterize and eventually predict the impact of sediment transport on coastlines over time, researchers have investigated how sediment transport dictates the development of coastal bedform structures (e.g., Voulgaris and Morin, 2008; Traykovski, 2007). These structures range from the micro- to macroscopic scale and include sand ripples, spits, and barrier island complexes, respectively. The formation of the largest scale feature, barrier island complexes, for example, relies upon sediment transport for a consistent supply of sediment to counter-balance erosional forces. A reduction in sediment supply due to changes in sediment transport can result in devastating

consequences for these features, and subsequently coastline stability, as experienced recently by the United States' Gulf Coast during Hurricane Katrina (Day et al., 2007). The present study, however, is concerned with the morphology of smaller bedform structures, sand ripples. Bedform structures such as sand ripples reflect sediment transport processes and evolve over time to remain in equilibrium with the forcing hydrodynamic conditions. Therefore, the study of sediment transport in terms of its mechanisms (e.g., forcing hydrodynamics) may provide the capacity to predict bedform morphological evolution.

Progress in our ability to model and predict sediment transport in the coastal zone has been linked most closely to technological advances (Cacchione et al., 2006). Most notably, the application of bottom-mounted instrumented tripods developed in the 1960s provided investigators with the first *in situ* observations of active sediment transport in this dynamic environment. Until then, sediment transport and bedform structure regimes were thought to consist of only relict features little affected by modern processes. But, the tripod-based actualistic studies validated the observations made in laboratory-flume studies of active sediment transport (e.g., Shields, 1936). Development of new observational technologies throughout the 1980s and 1990s, such as acoustic Doppler current profilers, rotary fan beam sonars, and self-contained hard-drives and power sources, have advanced tripod-based studies to manage long-term *in situ* measurements of bedform morphology as well as the surrounding hydrodynamic conditions (Irish et al., 1998; Thorne and Hanes, 2001; Wynn et al., 2002; Lopes et al., 2004; Cacchione et al., 2006). The present study aims to use existing software technologies to automate the

processing of data from *in situ* studies, improving both the speed and repeatability of manual interpretations.

1.1 Sediment transport and bedform formation

Within the mounting work on sediment transport and bedform formation, sediment transport is subdivided into three types of transportation modes: dissolved load, suspended load, and bed load. Dissolved load is composed of ions in solution, dictated by the chemistry of the fluid environment and, therefore, is not involved in this study. Suspended and bed load, however, are the result of physical properties competing to both lift sediment particles from the bed surface (entrainment forces) and to bring them back to rest (settling forces), and are of interest to this study. To understand these forces mathematically, researchers have tested relationships of sediment transport to physical parameters of sediment (such as grain size and density) and hydrodynamic forces (such as fluid velocity) using lab-based flume studies (e.g., Davies and Thorne, 2008; Voulgaris and Morin, 2008; Catano-Lopera et al., 2009; Sekiguchi and Sunamura, 2004) and observation-based field studies (e.g., Douchette, 2002; Traykovski et al., 1999; Traykovski, 2007).

From these investigations, four important parameters in the mechanics of sediment transport have been identified. These include sediment size, sediment concentration, bottom stress, and bed roughness (Cacchione et al., 2006). Sediment size and concentration are measured using grab samples at the sediment-water interface, and these

parameters do not usually change significantly over short time scales. Therefore, these parameters are not addressed further in the present study. However, bottom stress and bed roughness are sampled using modern instrumented-tripod-based studies, and are related directly to the present study.

In order to lift a sand particle out of its resting place on the seafloor, inducing sediment transport, the bottom-stress forces exerted on the particle must be enough to overcome the forces holding the particle in place (Dean and Dalrymple, 2004). The forces acting on a sediment particle resting on the seafloor are a drag force (F_D) acting in the flow direction, a lift force (F_L) acting perpendicular to and away from the sand bed, and the weight of the particle (W_s) acting downwards. Figure 1 depicts these forces acting on a representative sand grain within a bed. These forces have been expressed in terms of the flow velocity *U* for an idealized spherical particle of diameter *d*:

$$F_D = \frac{1}{2} p C_D U^2 A_p \quad (1)$$

$$F_L = \frac{1}{2} p C_L U^2 A_p \quad (2)$$

$$W_s = (p_s - p) g V_p \quad (3)$$

where p_s and p are the densities of sand and water, and g is the acceleration due to gravity (m s^{-2}).

To induce movement of the particle, the lift force does not necessarily need to exceed the weight of the particle, because the particle can begin to roll before it is lifted into the

flow. The following formula describes the force needed to induce movement, termed the *critical* bed shear stress (τ_c):

$$\frac{\tau_c}{(p_s - p)gd} = f(\mathrm{Re}) \tag{4}$$

which is a function of the Reynolds number (Re), defined as:

$$\mathrm{Re} = \frac{u_* d}{v} \tag{5}$$

where v is the kinematic viscosity, d is the diameter of the particle, and the shear velocity u_* is defined as:

$$u_* = \left(\frac{\tau_b}{p}\right)^{1/2} \tag{6}$$

The Reynolds number is used to predict flow regimes, with low Reynolds numbers generally being associated with laminar flows (as well as high numbers associated with laminar-like sheet flow) and middle numbers associated with turbulent flows (Figure 2). Furthermore, the left-hand side of equation (4) is known as the Shields parameter. The Shields parameter deals with the conditions needed for the initiation of particle movement (Garcia, 2000). To determine the Shields parameter, moments of initial particle movement were determined empirically for a number of different sediments and various steady-flow strengths (e.g., Gilbert, 1914; Shields, 1936). A curve was mathematically fit to the collective results of these studies, and is known as the Shields curve (Figure 3). When the Shields parameter for a flow regime falls above the critical threshold of the Shields curve (τ_c), sediment will begin to move. For different strengths of the Shields parameter, it has also been determined that different bed forms may result (Dean and Dalrymple, 2004). For Re < 10, sand ripples form on the bed, whereas for a range of 10 < Re < 100, dunes form. When Re is large, the bed moves in sheetflow (when

sand moves as a flat blanket) with no resulting roughness. The presence of bed roughness, such as sand ripples, can act to reduce the Reynolds number required to initiate a turbulent flow, shifting otherwise laminar flow regimes into turbulent flows.

In the coastal environment, the sources of movement-initiating flow are tidally produced currents and waves. Tides are astronomically driven changes in relative sea-level caused by the gravitational pull of the moon and sun (Ritter et al., 2006). Tides can have a range (difference from high tide to low tide) of greater than 4 m (megatidal or macrotidal), to between 2-4 m (mesotidal), to < 2 m (microtidal) displacement. Tidal currents produced by the rise and fall of the tides act as steady directional flows (Jiménez et al., 1997). Depending on their velocity, the tidal flows create relatively high sediment-transport rates, especially in confined areas such as tidal inlets (e.g., Roelvink and Stive, 1990). Tidal current steady directional flow is described by the mathematical relationships discussed above.

Waves are, in general, the most energetic forcing agent acting in the coastal zone (Jiménez et al., 1997). Most waves that modify the coastal zone are produced in the open ocean when energy from strong winds is transferred to the water. This transfer of energy creates wind-generated surface waves, or gravity waves (Ritter et al., 2006). The three primary factors responsible for the formation of waves are: (i) how fast the wind is blowing (velocity); (ii) the length of time the wind blows (duration); and (iii) the distance over the water that the wind blows (fetch). The energy in the resulting gravity wave, E, in Joules m^{-2}, in the ocean is given by the following equation:

$$E = 1/8(\rho g H^2) \quad (7)$$

where p is the density of water, g is the acceleration due to gravity, and H is the wave height (m) (Soulsby, 1997).

Within a gravity wave, energy travels across and through the water in the direction of the propagation of the wave form due to restoring forces that cause an oscillatory or circular motion (unsteady flow) roughly sinusoidal in form (Davis and FitzGerald, 2004). Each individual wave can be described by its wavelength (λ), the horizontal distance between two like locations (i.e., crest to crest); the wave height (H), the vertical distance from the base of the trough and the crest; the steepness (H/λ); and the period (T), the time in seconds for a complete wavelength to pass (Figure 4). The wave form transfers wave energy through the water in the direction of propagation, while the water itself theoretically stays in the same location, moving only back and forth in oscillatory circles. However, in actuality, water in a wave form has net movement in the direction of wave propagation due to drag forces of the wind (Figure 4). The diameter of the oscillatory circle at the water's surface is equal to the wave height. This diameter decreases with depth until, at roughly ½ the wavelength of the surface wave, the orbital is too small to interact with the bottom (Dean and Dalrymple, 2004). The depth at which the orbital motion is just able to interact with the seafloor is referred to as wave base. When this interaction is strong enough, and the critical Shields threshold is surpassed, bed movement will be initiated. The critical Shields parameter is represented in different forms for oscillatory flows than for steady flows:

$$\frac{\tau_c}{(p_s - p)gd} = f1\left(\frac{(p_s - p)gd^3}{pv^2}\right) = f2\left(\frac{d}{4v}\sqrt{\frac{(p_s - p)}{p}gd}\right) = f3\left(\sqrt[3]{\frac{(p_s - p)g}{pv^2}}d\right) \quad (8)$$

These forms of the critical Shields relationship have an advantage over the equation for steady flow (Equation 4) because they do not explicitly involve the shear velocity u_*, which is a function of time in unsteady flows (Dean and Dalrymple, 2004).

Seasonality can be a factor in determining wave energy and its morphodynamic effect on the coast (Jiménez et al., 1997). For example, a period of low wave steepness will result in beach profile accretion and an advance in shoreline, while a period of high steepness results in shoreline retreat. Wave direction also factors into the morphodynamic effect of waves, as it controls the intensity of longshore transport, the lateral movement of sediment along a shoreline (Ritter et al., 2006). A final effect on wave energy in the coastal environment is due to the tides. Tide currents reposition the surf zone during different tidal stages (Jiménez et al., 1997). For example, the surf zone will migrate towards land during the flood tide and back to sea with the ebb tide. When the surf zone migrates landward, energy from waves that once broke further offshore is brought to more-nearshore areas.

To conclude the discussion on sediment transport, the entrainment forces discussed above must be balanced by addressing the countering settling forces. Fall velocity, *w*, is the maximum speed attained by a falling particle under the action of gravity, and is an important hydrodynamic characteristic of a sand particle when dealing with sediment

transport. The size of a suspended sand grain will depend on the fall velocity (Dean and Dalrymple, 2004). It will be less likely for a large sand grain, which falls rapidly, to be suspended as compared to a finer sand grain. The fall velocity can be calculated theoretically from a balance of the forces from the weight of the grain W, the buoyancy force provided by the water (F_B), and the fluid drag that the falling grain experiences (F_D), which when solved for fall velocity leads to:

$$w = \sqrt{\frac{4(p_s - p)gd}{3pC_D}} \tag{9}$$

where C_D is the drag coefficient of the falling grain, which is a function of the Reynolds number (Re). Through analytical studies, Stokes (1851) calculated that the drag coefficient for spheres where there is a very small Reynolds number was obtained through

$$C_D = 24/\mathrm{Re} \tag{10}$$

and substituting this into the fall velocity equation (9), provides what is known as Stokes law:

$$w = \frac{(p_s - p)gd^2}{18\mu} \tag{11}$$

The mathematical relationships of bottom stress and the subsequent forms of bed roughness due to sediment transport have been utilized to develop predictive models of bedform morphology (e.g., Grant and Madsen 1982; Nielsen 1981; Wiberg and Harris 1994). Recent work has integrated these models with field observations to relate hydrodynamic energy from both currents and waves to the evolution of coastal bedforms over time (e.g., Traykovski, 2007; Voulgaris and Morin, 2008; and references within

Davies and Thorne, 2008). However, comparison of the predicted bedform morphology from these models to those of observed field measurements has shown that the models are inaccurate predictors of bedform morphology within various field sites (e.g., Douchette, 2002; Traykovski, 2007; Li and O'Connor, 2007). Yet, with a growing database of observational data derived from field sites around the globe, researchers are able to continually modify their formulas and reform their models, increasing the predictive capabilities of these models.

A recent field-based observational study (Voulgaris and Morin, 2008) with the goal of testing a time-variable model (Traykoski, 2007) incorporated a MATLAB-based 2-Dimensional Fast Fourier Transform (2D FFT) spectral analysis for the automated extraction of sand-ripple morphology. The program determined sand-ripple wavelength (λ) and ripple-crest angles of orientation (φ) from seabed images. The objective of this honors project has been to develop a similar spectral analysis tool, and to test its limitations in extracting ripple wavelength and orientation values within a hydrodynamically complex field site at Hen and Chickens Shoal, Delaware.

1.2 2-Dimensional Fast Fourier Transform Spectral Analysis

Fourier theory is based on extracting component frequencies from sinusoidal signals. A basic illustration (Figure 5) depicts a clean (no noise) sinusoidal signal with one frequency analyzed with a Fourier transform. This is compared to the Fourier transform of a bimodal sinusoidal signal corrupted with random noise. The clean signal has a single peak in its Fourier transform, at the frequency of the signal. The second signal (Figure

5B) has two distinct peaks in its transform, with the random noise of the signal linked to multitudes of small magnitude peaks. The signals in Figure 5 are discrete time-based data and, therefore, a discrete Fourier transform (DFT) was used to transform the time-based data into frequency-based data.

In Fourier theory, however, *any* signal is the composition of the sum of a number of different sinusoidal signals (Lehar, 2010). In the present study, these signals are visual images of sonar returns from the seafloor. A visual image is a matrix of pixel values, and a 2-Dimensional Fast Fourier Transform (2D FFT) must be utilized. The 2D FFT computes the DFT for each column and then for each row of pixels and encodes all of the spatial frequencies present in the image (Matlab, 2008). Figure 6 provides examples of three different visual images analyzed with a 2D FFT. For images with a signal containing only a single spatial frequency (*f*), the plot has a single point along the axis, with the height of the peak corresponding to the amplitude (Figure 6). Due to computational methods, the Fourier plot also returns this single point with its mirror image, resulting in two peaks in the image. Lastly, there is a DC term, corresponding to zero frequency, which represents the average signal strength across the image. The peaks of high frequency signals (Figure 6A') are plotted further from the DC term than low frequency signals (Figure 6B'). The angle of orientation is also plotted by the 2D FFT (Figure 6C'). It has been demonstrated by Vougaris and Morin (2008) and through preliminary work to this study that bedform features, such as sand ripples, imaged using a rotary fan beam sonar can be interpreted by the 2D FFT as distinct sinusoidal signals.

2. Field Site and Collection of Data to be Analyzed Spectrally

Sonar-image and hydrodynamic data were collected during an *in situ* bedform study at Hen and Chickens Shoal (H&C), a bay-mouth spit associated shoal east of Cape Henlopen, Delaware, at the mouth of the Delaware Bay (Figure 7A, B). The purpose of the study was to (1) image the evolution of ripples over time while recording the forcing hydrodynamic conditions, and (2) to compare the observed ripple morphology to the predicted morphology of several models. An instrumented frame deployed at a depth of 7.4 m was fitted with an upward directed 1 megahertz (MHz) Nortek AWAC positioned 0.45 m above the bed and a 2.25 MHz Imagenex tilt head rotary imaging sonar 1 m above the bed (Figure 7C, D). Current direction and velocity were recorded every 10 minutes. Wave height, direction, and period were sampled for a 17-minute burst once every hour. The deployment began on August 22, 2006 and continued until October 16, 2006. These measurements record a generally dynamic and complex hydrodynamic environment with one major storm event caused by the remnants of Tropical Storm Ernesto (Figure 8). During non-storm background conditions, water depth remained relatively constant around 7 m. Wave height during the storm event was as high as 4 m. Hen and Chickens shoal experiences tidally-linked diurnal shifts in the dominant hydrodynamic force, from near bed currents during flood or ebb tidal stages to orbital wave energy during slack tides.

Acoustic rotary fan beam-sonar images were recorded with the Imagenex tilt head rotary imaging sonar every hour at 3600 s time intervals along 8-m radials to document the

morphological shifts of the bedform to prevailing wave-and-current activity. The Imagenex sonar is theoretically capable of imaging the seafloor at 3 cm per data pixel. The Hen and Chickens shoal rotary sonar data are characterized by useable high-intensity returns out to roughly 5 m from the instrumented frame, with low-intensity returns outside that range. The range of the usable high-intensity returns is the consequence of the tilt angle of the sonar head. The steep angle of the tilt head forced strong sonar returns adjacent to the instrument frame and low-intensity returns away from the frame. The useable data are shifted ~0.25 m from center towards the lower left region of the image from the beginning of the study until the storm event on Sept. 2, 2006. Then, the useable data shifted towards the upper right quadrant of the images immediately after the storm event. The power source for the sonar was depleted before the end of the study, and there are no usable images after September 6, 2006.

Preliminary assessment of the sonar images revealed that the tidally-linked diurnal shifts in dominant forcing hydrodynamics are linked to dynamic shifts in bedform morphology. When wave-and-current velocities alternately surpassed the Shields threshold for sediment transport, sediment transport was initiated and erased previous bedform structures. The bedform structures then shifted towards equilibrium with the new dominant hydrodynamic force. However, the bed did not reach equilibrium with the hydrodynamics due to the next repeated shift in dominant hydrodynamic forces (Skarke, pers. communic., 2009).

Time-lapse movies of the sonar images from background conditions and the storm event are included on a CD-ROM as Appendix I.

3. Methods

During the summer of 2009, this author was a Research Experience for Undergraduates (REU) intern at the Coastal Sediments, Hydrodynamics, & Engineering Lab (CSHEL) of the Department of Geological Sciences at the College of Marine and Earth Studies, University of Delaware. The objective of the REU work was to develop an automated 2D FFT procedure in MATLAB for extracting a representative ripple wavelength (λ) and ripple crest angle of orientation (φ) for each sonar image from the Hen and Chickens shoal (H&C) data set. The project mimicked the design of Voulgaris and Morin (2008) but without access to their full code. By the end of the summer REU, a functioning program was produced, but due to time constraints, limited investigation could be given to the accuracy of the wavelength and orientation outputs. Therefore, the investigation in this honors thesis was designed to further develop and investigate the applicability of the automated 2D FFT spectral analysis tool for analyzing the hydrodynamically complex H&C data set. Based on preliminary tests from the summer project, investigation into four aspects of the spectral analysis procedure were incorporated in the current study: (1) the use of an 8 sub-region arrangement for analysis versus five other arrangements, (2) the procedure for averaging the wavelength-and-orientation outputs from each sub-region to produce a single representative value for each image, (3) methods for eliminating and/or filtering poor quality data, and (4) the procedural steps in the spectral analysis that can appropriately incorporate automation versus needing manual involvement and interpretation.

In order to test the suitability of the 8 sub-region approach of Voulgaris and Morin (2008) to the H&C data set, the 8 sub-region arrangement was tested versus five other arrangements: a single sub-region of 64 m^2 (Figure 9A), two arrangements of two equal sub-regions of 16 m^2 (Figure 9B), four equal sub-regions of 4 m^2 (Figure 9C), and an interactive manually selected sub-region system (Figure 9E). In the latter case, the location of each sub-region is not held constant for each image. Rather, a single sub-region of 4 m^2 is located by the user for each image. In the first five arrangements, however, the sub-regions do not change their location between images. The locations were assigned at the beginning of the study and held constant throughout. The location and size of the sub-regions in these arrangements were designed to provide the present study with two dimensions in which to compare the wavelength and orientation values of sub-region arrangements. These are (1) differences between larger scale sub-regions versus smaller scale sub-regions, and (2) differences between the locations of each sub-region within the sonar image.

Preliminary results from the REU highlighted that the procedure for averaging 8 sub-regions' Fourier transforms used by Voulgaris and Morin (2008) could negatively influence the final wavelength-and-orientation output. This occurred when sub-regions with inaccurate results were averaged with sub-regions with accurate results. Therefore, to quantify the effect of averaging the sub-regions in the present study, the wavelength-and-orientation outputs are compared pre-averaging to the outputs post-averaging. Two methods for the averaging procedure are compared: (1) average all the Fourier transform spectral plots from the sub-regions into a new averaged spectral plot first, and then output

a single wavelength-and-orientation value from that averaged spectral plot; and (2) output wavelength-and-orientation values from each sub-region's Fourier spectral plot to then average the wavelength-and-orientation values together producing a single averaged value.

Lastly, results of pilot studies conducted during the REU revealed that some images were either void of, or limited in, the clarity of bedform structures. Spectral analysis of these images produced inaccurate wavelength outputs such as infinity or five meters when manual interpretations determined that no bedform structures in the H&C data set scaled to these values. Furthermore, bedform structures with multiple orientations sometimes occur in a single image, and a single orientation output value would be inappropriate. Therefore, in the present study, investigation is given to the application of filters for clearing the program of entire sonar images, and/or individual sub-regions within the image, with poor clarity of bedform structures and/or with multiple bedform orientations. Automated procedures for this filtering were developed and tested.

3.1 Spectral analysis procedure

The spectral analysis procedure developed in this study imports and corrects the raw Imagenex rotary beam sonar images for slant range, orients them to geographic North, converts the data from polar to Cartesian coordinates (X-axis East, Y-axis North), and evaluates the data spectrally for ripple wavelength and orientation using the MATLAB 2-D Fast Fourier transform (fft2) routine (Figure 10). The MATLAB code for reading the

raw Imagenex data and rectifying them for their coordinate-system conversion were developed by Dr. George Voulgaris, and provided at the onset of this study. Conversely, the remainder of the spectral analysis procedure applied in the current study was developed herein, and is based upon the trials and iterations from a number of internal pilot studies into each aspect of the procedure. The code for the entire procedure is included as Appendix II.

The spectral analysis procedure is not performed on the entire image, but within square sub-regions of the sonar image. Each sub-region represents a select portion of the seafloor image, and the area within each sub-region is dependent upon the arrangement of the sub-regions. The sub-regions are all located within 4 m of the image center in order to incorporate only the high-intensity data discussed above. The low-intensity data in the outer regions are not deleted from the image, but trimmed from the analysis by positioning the sub-regions near the center.

3.1.1 Assigning the sub-region arrangement

The first two processing steps are to assign the sub-region arrangement (how many sub-regions will be used) and the locations of each sub-region within the Imagenex sonar image. When the MATLAB program is executed, the user is given a menu choice to select the sub-region arrangement:

```
subregion_arrangement=menu('Choose the subregion arrangement','1
subregion','2 subregions left and right', '2 subregions top and bottom', '4
subregions', '8 subregions', 'Manually selected sub-region');
```

The user's selection determines which arrangement will be used in the subsequent analysis. The program then uses this choice to run a case statement for assigning the location of each sub-region in the arrangement. If the user chooses option 1, case 1 will be run; if s/he chooses option 2, case 2 is run, etc. In each case, the sub-region boundaries are defined by an xminimum, xmaximum, yminimum, and ymaximum value, and the four values are placed into a matrix called "subregions_dimensions." Once the program has been initiated, these locations are held constant and the user does not need to redefine the boundaries for each image. In a sense, the process then becomes automated. However, it does not become adaptive.

(NOTE: Any code following a % is a comment, hence, it is not a formal aspect of the code, but is included for explanation.)

```
switch subregion_arrangement
case 1
        %Dimensions for 1 subregion
        xminimum=[-4],
        xmaximum=[4],
        yminimum=[-4],
        ymaximum=[4],
%subregion dimensions combined into a matrix
subregions_dimensions=[xminimum' xmaximum' yminimum' ymaximum'];

case 2
        %Dimensions for 2 subregions
                %left  right
        xminimum=[-4    0],
        xmaximum=[0     4],
        yminimum=[-2   -2],
        ymaximum=[2     2],
subregions_dimensions=[xminimum' xmaximum' yminimum'ymaximum'];

 case 3
        %Dimensions for 2 subregions
                %top  bottom
        xminimum=[-2   -2],
```

```
        xmaximum=[2     2],
        yminimum=[0    -4],
        ymaximum=[4     0],
subregions_dimensions=[xminimum' xmaximum' yminimum' ymaximum'];

case 4
        %Dimensions for 4 subregions
                %up      right        bot        left
        xminimum=[-1        1         -1         -3],
        xmaximum=[ 1        3          1         -1],
        yminimum=[ 1       -1         -3         -1],
        ymaximum=[ 3        1          1          1],
subregions_dimensions=[xminimum' xmaximum' yminimum' ymaximum'];

case 5
        %Dimensions for 8 subregions overlap
        %top, topright, right, bottomright, bottom, bottomleft, left, topleft
        xminimum=[-1   .5       1     .5   -1  -2.5  -3   -2.5],
        xmaximum=[ 1   2.5      3    2.5    1   -.5  -1    -.5],
        yminimum=[ 1   .5      -1  -2.5   -3  -2.5  -1     .5],
        ymaximum=[ 3   2.5      1    -.5    1   -.5   1    2.5],
subregions_dimensions=[xminimum' xmaximum' yminimum' ymaximum'];
```

The sixth option in the menu is the “manually selected sub-region,” and its case is run slightly different than the previous sub-region arrangements. Here, the sub-region locations are not held constant for each image, but interactively redefined by the user for every image. A sonar image is presented to the user on the screen, and by using a “ginput” function, a cursor is displayed on the image (Figure 9E). The cursor tracks with the mouse, and the center of a sub-region is assigned by a mouse click. The user can define multiple sub-regions with multiple clicks. To exit the selection mode, the user presses “enter” on the keyboard.

Code for the manual sub-region arrangment:

```
        otherwise
```

```
%this first section of code produces an image of the sonar from the raw
%data file.
for i=image_selection;
for ii=(iikey(i));
fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1;
sw=1;
[X,Y,Z]=sonarimage(Data,AA,z,sw,T);
datestr(T(1),30)
fnamef=datestr(T(1),30);

%This assigns the x and y coordinates using a ginput function
[xcoordinate,ycoordinate] = ginput;

close; %closes the figure window once you press enter
```

The "xcoordinate" and "ycoordinate" values from the ginput are the coordinates for the center of the sub-region. Therefore, to define the x,y minimum and x,y maximum, a value has to be added and subtracted to the center coordinate. In this study, this value is equal to 1, which is 1/2 the width of the 2 m x 2 m sub-region.

```
%The size of the sub-region is determined by adding and subtracting one half the length
of each side. Here, the squares will be 2 meters wide, so 1 is added.

xminimum=(xcoordinate)-1;
xmaximum=(xcoordinate)+1;
yminimum=(ycoordinate)-1;
ymaximum=(ycoordinate)+1;

subregions_dimensions=[xminimum xmaximum yminimum ymaximum];
```

3.1.2 Extracting data within sub-regions and regridding

Before running the 2D FFT spectral analysis, the x,y,z data values for each sub-region must be extracted from the original data matrices and regridded to a size that preserves

the original resolution of the seafloor in each sonar image. This is a two step processes: extraction and regridding. For extraction, a "find" function utilizes the x,yminimum and x,ymaximum for each sub-region to locate the associated data from the original data matrix. All values between the x-limits and the y-limits are identified, and the intersect function then is used to find the locations where they overlap.

```
%code to extract the appropriate data values.
B1x=find(X>(subregions_dimensions(r,1))& X<
(subregions_dimensions(r,2))); %Find indices for X values between xmin
and xmax
B1y=find(Y>(subregions_dimensions(r,3))&Y<
(subregions_dimensions(r,4)));%Find indicies for Y values between ymin
and ymax
B1i=intersect(B1x,B1y); %Finds where those indicies overlap
B1X=(X(B1i)); %Extracts the values corresponding to those indices from
all X data
B1Y=(Y(B1i)); %Extracts the values corresponding to those indices from
all Y data
B1Z=(Z(B1i)); %Extracts the values corresponding to those indices from
all Z data
```

The extracted data then must be regridded to a matrix while preserving the original data resolution. The Imagenex sonar is theoretically capable of imaging the seafloor at 3 cm per data pixel. Consequently, the length of seafloor represented by each pixel in the 2D FFT must not be less than 3 cm. Using a higher resolution during the regridding process introduces artifacts into the regridded matrix (Voulgaris and Morin, 2008).

In the current study, the length of seafloor analyzed by the 2D FFT is different for each different sub-region arrangement. The single sub-region arrangement represents 64 m^2 of seafloor; in both of the two sub-region arrangements each sub-region represents 16 m^2; and the four, eight, and manual arrangements' sub-regions each represent 4 m^2. To

preserve at least the 3 cm pixel widths, the sub-region arrangements are split into 256 x 256 pixels; 128 x 128 pixels; and 64 x 64 pixels, respectively. The logic is that the 8 m width of the single sub-region arrangement, for example, is divided by 256, resulting in 3.125 cm of seafloor represented per pixel. To deal with the slight differences in these calculations for each sub-region arrangement, a switch function utilizes the menu selection made at the onset of the program to run a case statement.

As an example of this process, the code for regridding the data into 64 x 64 pixels for the 4 m^2 sub-region in the four sub-region, eight sub-region, and manual arrangements is provided below. A meshgrid function is used to produce arrays XI and YI – the x,y coordinates for each pixel. The grid is actually first produced at 66 x 66, not 64 x 64, because the regridding process was found to produce two rows and columns of "not a number" (NaN) values. The NaN values are eventually removed to leave a 64 x 64 matrix. The first x,y values assigned by the meshgrid function to the XI and YI arrays are the x,y minima. The length of the sub-region (in this example, 2 m) is divided by 65 producing a constant (.0307), which is added to the previous XI YI values in a geometric series to give new XI and YI values:

```
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/65):(subregions_dimensions(r,2)),...
(subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/65):(subregions_dimensions(r,4)));
%Produces a 66x66 grid
```

A griddata function is utilized to fit a surface to the x,y,z image data extracted earlier, interpolated at the points in the XI YI arrays.

```
%Grid the indexed values from X,Y,Z with the meshgrid dimensions
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];

%Cut the grid down to 64 x 64
ZI_64=ZI(2:65,2:65);
ZI_64(isnan(ZI_64))=0; %Any remaining NaN values are zero padded
```

A detrend subroutine removes the mean value or linear trend for each column of the matrix, and then for each row.

```
ZF=ZI_64;
ZF=detrend(ZF'); %Removes long term trend from rows
ZF=detrend(ZF'); %Removes long term trend from columns
```

3.1.3 2-Dimensional Fast Fourier Transform

Once the extracted data are regridded, the routine calls upon the MATLAB routine (fft2) to perform the 2D FFT. The fft2 routine returns the 2-Dimensional Discrete Fourier Transform (DFT) of each sub-region matrix with a Fast Fourier transform (FFT) algorithm (Equation 12). The computation is performed as the one-dimension DFT of each column within the sub-region, then of each row of the result. The execution time for the fft2 function is fastest for powers of two, one reason why the size of each matrix in this study is 64 x 64; 128 x 128; or 256 x 256.

2D FFT code:

```
ZF=fft2(ZF); % Fourier Transform of matrix
ZF=abs(ZF); % Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(128*128); % Complex conjugate of matrix
ZF=fftshift(ZF); % Centers spectral peaks of matrix (outside corners are
flipped to center)
```

The function Y=fft(x) implements the transform for vectors of length N by:

$$X(k) = \sum_{j=1}^{N} x(j)\omega_N^{(j-1)(k-1)} \tag{12}$$

where,

$$\omega_N = e^{(-2\pi i)/N}$$

is an Nth root of unity.

The fft algorithm is based on a freeware C subroutine library for computing the discrete Fourier transform (DFT) called FFTW (Frigo and Johnson, 1998). To compute an N-point DFT when N is composite (when N=N1N2), the FFTW breaks down ("decomposes") the transform problem using the Cooley-Tukey algorithm (Cooley and Tukey, 1965). This algorithm first computes N1 transforms of size N2, and then computes N2 transforms of size N1 recursively until the problem is solved using one of several algorithms (Matlab, 2008).

3.1.4 Wavelength and Orientation Calculation

Following the procedure of Voulgaris and Morin (2008), the 2D FFT matrix must be converted from frequencies to wave numbers in order to scale the frequencies to actual distances along the seafloor. This conversion follows the instructions described in Krogstad (2004). Krogstad (2004) provides the following code for conversion from frequency to wave number:

```
kx1 = mod( 1/2 + (0:(M-1))/M , 1 ) - 1/2;
kx = kx1*(2*pi/ΔX);
ky1 = mod( 1/2 + (0:(N-1))/N , 1 ) - 1/2;
ky = ky1*(2*pi/ΔY);
[Xw,Yw] = meshgrid(kx,ky);
```

where, M=N=largest dimension of sub-region; and ΔX=ΔY=seafloor length per pixel

By assuming that kxm and kym are the east and north wavenumbers that correspond to the peak spectral energy, wavelength (λ) and orientation (φ) of the ripple crest are given by:

```
%MATLAB code to determine peak value
max(ZF_max);
peak_spectral_energy=max(ans)
```

The ripple wavelength is then determined by the following equation:

$$\lambda = \frac{2 \cdot \pi}{\sqrt{k_{xm}^{\ 2} + k_{ym}^{\ 2}}} \tag{13}$$

MATLAB code:

```
(wavenumber_index=find(ZF_max==peakmax);Ripple_Wavelength=(2*p
i)/(sqrt((Xw(wavenumber_index(1)))^2 + (Yw(wavenumber_index(1)))^2
))
```

And orientation of the ripple crest is determined by:

$$\varphi = \tan^{-1}\left(k_{ym} / k_{xm}\right) + \pi \tag{14}$$

MATLAB code:

```
Ripple_Orientation=atand((Xw(wavenumber_index(1)))/(Yw(wavenumbe
r_index(1)))) + pi
```

Running this analysis through the time-series of rotary images outputs a string of ripple wavelengths and orientations for the rotary sonar images. In Voulgaris and Morin (2008), the 2D FFT plots from all 8 sub-region are averaged together into one Fourier plot before the calculation of ripple wavelength and orientation. In the current study, this method is mimicked by creating a variable for each Fourier transform, "ZF_#", depending on the sub-region, then averaging these transforms together into a single plot and finding the wavelength and orientation within the averaged plot:

```
switch subregion_Arrangement
case 1
        %Average equals the single transformed sub-region.
        ZFA=((ZF_1));
case 2
        %Average the 2 transformed sub-regions together.
        ZFA=((ZF_1+ZF_2)/2);
case 3
        %Average the 2 transformed sub-regions together.
        ZFA=((ZF_1+ZF_2)/2);
case 4
        %Average the 4 transformed sub-regions together.
        ZFA=((ZF_1+ZF_2+ZF_3+ZF_4)/4);
case 5
        %Average all 8 transformed sub-regions together.
        ZFA=((ZF_1+ZF_2+ZF_3+ZF_4+ZF_5+ZF_6+ZF_7+ZF_8)/8);
```

In the present study, a second method for averaging the Fourier analysis was investigated, where the wavelength and orientation were calculated for each Fourier plot first and the average then taken of these values:

```
Wavelength_Average=mean(Subregion_Ripple_Wavelengths(i,:));

Orientation_Average=mean(Subregion_Ripple_Orientations(i,:));
```

3.1.5 Filters

From results obtained in the author's REU work and internal pilot investigations to the present study, two filters were deemed necessary to include in the present spectral analysis procedure. These filters are designed to eliminate unreliable wavelength outputs resulting from inherent computational difficulties with the H&C data set and current methodology. Both act as a response to symptoms of an unreliable wavelength output rather than a fix to the underlying source of the problem. Procedures and filters capable of solving the underlying problems require access to tools than were unavailable in the

present study, and these tools are mentioned and described in the discussion section below.

The first filter is used to eliminate a group of low frequency peaks in the Fourier plots that is an artifact of the sub-region regridding process and of low frequency noise from the low-quality outer portions of the tilt-head images. The regridding artifacts are present because the four outside edges of each sub-region appear to be like-parts on a sinusoidal signal, the Fourier analysis identifies the left-to-right / top-to-bottom edge pairs as like points on a single wavelength, and a very strong peak scaling to these dimensions is plotted in the Fourier plot. The central group of peaks in the Fourier plot must be removed, leaving behind only the peaks corresponding to the sand ripple wavelengths. To incorporate the filter, the data points in the Fourier plot corresponding to the central region are replaced with low values, thereby making the sand ripple wavelength peaks the most prominent peaks in the Fourier plot:

```
%Filter #1 for 256 x 256. Removes the central peak that is linked to sub-region dimensions
g=zeros(256,256); % creates a low frequency noise filter for matrix
g(118:138,118:138)=.5;
g(120:136,120:136)=.97;
g(122:134,122:134)=1;

ZF=ZF-(ZF.*g);%applies low frequency noise filter
```

The data points to be replaced must be determined for each sub-region dimension as the central peak scales to the sub-region's dimensions. The above example was developed to work for a 256 x 256 plot while below are the filters designed for a 128 x 128 and 64 x 64 plot, respectively.

```
%Filter #1 for 128 x 128. Removes the central peak that is linked to sub-region dimensions
g=zeros(128,128);% creates a low frequency noise filter for matrix
g(50:79,50:79)=.5;
g(57:72,57:72)=.97;
g(61:67,61:67)=.98;
g(62:66,62:66)=.99;
g(63:65,63:65)=.99;
```

```
%Filter #1 for 64 x 64. Removes the central peak that is linked to sub-region dimensions
g=zeros(64,64);% creates a low frequency noise filter for matrix
g(29:35,29:35)=.5;
g(30:34,30:34)=.98;
g(31:33,31:33)=.99;
```

The second filter in the spectral analysis process is used to eliminate any output values that do not make logical sense when compared to the actual scale of bedforms in the images. For example, a value of 7 m or infinity for ripple wavelengths is unreasonable for any image in the H&C data set. Inclusion of these outlier values into the averaging process will distort the representative wavelength-and-orientation values. To design this filter, a basic familiarization with the data set is necessary. Upon familiarization with the H&C data set, it was obvious that no images contained bedforms with wavelengths greater than 0.6 m, with a margin of error as well. Therefore, 0.6 m was designated as the cutoff for unreasonable wavelength calculations. For implementation, the filter is included after the wavelength-and-orientation calculations. The calculations for wavelength-and-orientation are performed and then an “if” statement is called upon to identify sub-regions with reasonable wavelength calculations. Those sub-regions with wavelength values less than 0.6 are saved into a matrix that can be used later for averaging.

```
%This is a 2nd filter that will get rid of any outputs that have the  being greater
than 0.6 meters because 0.6 meters is larger than any ripples imaged
if (Ripple_Wavelength < 0.6);
cutoffcount=1+cutoffcount;%makes a counter to place each value into its own
new column starting with column 1
Subregion_Ripple_Wavelengths_filtered(i,cutoffcount)=Ripple_Wavelength;
Subregion_Ripple_Orientations_filtered(i,cutoffcount)=Ripple_Orientation;
end
```

For the present study, the entire 2D FFT spectral analysis as described above in each of its sub-region arrangements was run to analyze the Hen and Chickens shoal data at least one time per sub-region arrangement. The data were then also analyzed with aspects of the code, such as the filters, removed, so that the contribution of each step could be quantified.

4. Results

The results from the present study are presented in this section to both better describe the Hen and Chickens shoal data set and to quantify the ability of the developed 2D FFT spectral analysis to extract ripple wavelength-and-orientation values from images within the data set. The procedure for the 2D FFT spectral analysis was designed to output a single value for wavelength and a single value for orientation per each image. In the following results section, however, plots of the analytical outputs will only contain wavelength values. The trends represented by the wavelength plots are identical to the trends for orientation values, evident because the MATLAB code for calculating the wavelength and orientation values use the same spectral peak in the Fourier plot. Changes to the single spectral peak will affect the values equally. Also, the results below are

arranged in an order that reflects the procedural methodology of the spectral analysis. Specifically, the general characteristics of the bedforms in the data set are first, followed by output values, ending in the quantifications of the filters. This arrangement is intended to reflect the step-wise nature of the analysis program and to highlight the results obtained from each aspect of the code.

4.1 Hydrodynamic Conditions and Bedform Morphology During Summer 2006

The current-and-wave data from the acoustic Doppler current profiler record a complex hydrodynamic regime during the time of the H&C study that can be split into four stages (Figure 8D). Distinction is made between the four stages based on two factors, (1) the relative strengths (or dominance) of current and wave energy, and (2) whether wave and/or current energy surpasses the Shields critical threshold. The first hydrodynamic stage is where waves and currents alternate as the dominant hydrodynamic force, with current energy dominant over wave energy during flood and ebb stages, and wave energy dominant over current energy during slack tide. During flood and ebb stages, current energy surpasses the Shields critical threshold and during slack tide it does not. Whereas, wave energy does not surpass the Shields critical threshold at any time, even when it is dominant during slack tide. Within the time frame when sonar images were recorded (from August 21-September 6) stage one of the hydrodynamic regime occurs from August 21 - 26. The second hydrodynamic stage is when wave energy and current energy are again alternating as the dominant force, yet wave energy surpasses the Shields critical threshold. Again, current energy only surpasses the threshold during flood and ebb

stages. This stage occurs, for example, from August 29 - 30. The third wave-and-current regime stage is when wave energy is continuously above the Shields critical threshold and is always dominant over current energy, even during flood and ebb tides. This stage occurs during the build-up to the Tropical Storm Ernesto event from August 30 – September 1, as well as during other smaller storms such as from August 26 - 29. The fourth and final wave-and-current regime stage is during the large Ernesto storm event where wave energy greatly surpasses the Shields critical threshold current for a sustained time from September 1 – 2.

Three hundred and fifty four sonar images were successfully processed by the spectral analysis routine, while 21 images failed to be read into the program and/or converted to Cartesian Coordinates. All 21 of those images that failed to be read were recorded during the interval when the Imagenex was powering down (Table 1). The bedforms in the 354 sonar images processed by the program were assigned to one of six different morphological categories: (1) No recognizable bedforms (mostly due to distortion during the storm event), (2) large scale strong interference ripples in a diamond-shaped hummocky pattern, (3) small scale interference ripples with interspersed areas of a quasi-linear arrangement, (4) quasi-linear arrangement of interference patterns across the entire image, (5) complex overlapping ripples with poorly defined crests, and (6) complex ripples with well-defined crests (Figure 11). Of the 354 sonar images in the data set, 31 were category-one ripples, 50 were category-two, 99 were category-three, 80 were category-four, 60 were category-five, and 34 were category-six. The evolution of these bedforms over time demonstrates abrupt shifts in ripple morphology between consecutive images (Figure 12).

The shifts in ripple morphology throughout the study period can be matched to the forcing hydrodynamics (Figure 13). For example, during the second hydrodynamic regime stage, when current-and-wave energy both cross the Shields critical threshold, but neither is continuously dominant, bedform morphology shifts rapidly between poorly defined crested features (category 5) to strong interference ripples (category 2) depending on the dominant hydrodynamic force. Periods of more constant ripple categories occur when the dominant hydrodynamic force does not shift as abruptly. For example, the longest consecutive period of category 6 ripples occurred during stage 3 of the hydrodynamic regime at the time of the building storm event, when wave energy surpassed the energy of the current for an extended length of time.

4.2 Spectral Analysis Outputs

The spectral analysis routine was run in its entirety for each of the six sub-region arrangements. The approximate processing time for the spectral analyses were as follows: 15 seconds per image in the one sub-region arrangement; 13 seconds per image in the two sub-regions arrangements; 15 seconds for the four sub-region arrangement; 22 seconds per image for the eight sub-region arrangement; and 20 seconds per image with the manual arrangement method.

The Fourier transform plots produced by the spectral analysis in the present study can be separated to match the six morphological categories of the bedforms in the images (Figure 14). A common feature across all but the category 6 morphology plots is a group

of peaks in the center of the Fourier plot corresponding to low sinusoidal frequencies. The strength of a peak on these Fourier plots is represented by the intensity of its color, and the reds and yellows in these low frequency peaks signify their high strength above any other signal. When the wavelength-and-orientation values are extracted from these peaks their frequencies represent and scale to the size of the sub-regions in the analysis and to noise from the outer regions of the image, not to the bedform features. It is these peaks, therefore, that are identified and removed by the first filter in the procedure.

As the arrows in Figure 14C-F indicate, there are multiple other peaks in the plots corresponding to higher frequencies (shorter distances along the seafloor) scaling to ripple bedforms. Once the low frequency peaks are removed, these remaining peaks are the strongest remaining signals. The images with more well-defined linear ripples (categories 5 and 6) generally have two high frequency peaks remaining. One peak is dominant, with a lesser higher frequency peak at a slightly different orientation. The plots of morphology categories 3 and 4 (which had linear arrangements of hummocky features in parts of or the entire image between the categories, respectively) have multiple dominant peaks, at similar frequencies, but with orientations roughly orthogonal to each other. One peak in these bimodal plots corresponds to the linear arrangement of the features, while the orthogonal peak corresponds to the second orientation in the diamond-shaped interference ripple morphology. Lastly, the category 1 and 2 ripple morphologies do not have any well-defined high frequency peaks.

The Fourier transform plots described above for each morphology category were produced using the 8 sub-region arrangement of the spectral analysis, and showed variation between morphologies. Further variation in the Fourier plots is evident when the morphologies are analyzed by each of the other five sub-region arrangements. The distinctions are seen clearly in a "best-case scenario" analysis, which is defined here as analyzing a single image of category 6 ripples using each of the sub-region arrangements (Figure 15). The data in Figure 15 originate from August 30 at 2200 hours, and are image number 220. At the time the image was recorded, the wave energy was dominant over current energy and above the Shields critical threshold. In this scenario, the Fourier transform of the one sub-region arrangement plots only a low-frequency peak in the center of the plot, which would be removed by the filter (Figure 15A); both arrangements using two sub-regions plot two sets of peaks – one being the same low-frequency set in the center and the second an outer high frequency pair (Figure 15B, C); the four sub-region arrangement also outputs a Fourier plot with a central peak and two outer peaks, however, the central peak in these arrangements is not as strong a signal as in the plots from the arrangements above (Figure 15D); and six of the eight sub-regions in the 8 sub-region arrangement plot do not output a strong central peak, rather only having the high-frequency outer pair of peaks (Figure 15E).

The variation illustrated in Figure 15 occurs within a single image of the most well-defined ripples of the data set, in which everything was held constant but the sub-region arrangement. Therefore, the differences between plots increase when the five sub-region arrangements are compared to one another for the remaining 353 images with less well

defined ripple features. In Figure 16A, the wavelength calculations for each of the sub-region arrangements for every image are plotted together versus time (image number). Those values that plot as zero on the y-axis were output as infinity by the analysis and converted to zero for plotting (the orientation values for these infinity values was a NaN value). The single sub-region arrangement plotted the most zero-values across the data set with 294 out of 354. The two sub-region with a top/bottom arrangement plotted 24 zeros, while the remaining three approaches plotted two zero values. The two sub-region left and right, four sub-region, and 8 sub-region arrangements are the most similar in their remaining distribution. To further quantify the variation between the sub-region arrangements, the coefficient of variation (calculated as the standard deviation divided by the mean) was determined for each image (Figure 16B). Three periods of the study show the least amount of variation from the mean of roughly 10% or lower, with the remainder either around 50% or higher. The highest variation is over 200% from the mean. The images with the least variation between sub-region arrangements (highlighted in Figure 16B) correspond to category 5 and 6 ripple morphologies, while the highest variation in analysis between sub-region arrangements corresponded to category 1 and 2 ripple morphologies.

The variation in the Fourier plots and analysis outputs seen between morphology and sub-region arrangements is furthered by variation that occurs within each image. To illustrate the variation across a single image, Figure 17 plots the “perfect case scenario” from Figure 15E for the 8 sub-region arrangement. The 8 sub-regions are taken from across the image, and show differences in peak frequency and orientation, presence of a

low frequency central peak, and total number of peaks. The two individual sub-regions from the upper right and lower left regions have the most differences from the other six. The orientation of the ripples in these two regions is parallel with the path of the sonar beam from the rotary beam sonar. The individual sub-regions from the bottom and top left portions of the image have the best defined peaks in the plots, and the orientation of the ripples in their sub-regions are orthogonal to the path of the sonar.

4.3 Filters

The spectral analyses discussed above were run a second time with the two filters incorporated into the procedure. The first filter was designed to remove the low frequency peaks in the center of the plots associated with noise from the sub-region. The filter was applied to the Fourier plot of every sub-region before the wavelength-and-orientation values were determined. Figure 18 plots the effects of the filter on the two types of Fourier plots, unimodal and multimodal (which includes bimodal). In the unimodal plots (taken from sub-regions orthogonal to sonar beam in images with category 5 and 6 ripple morphologies), the filter properly eliminates the central low frequency peaks. However, for the bi to multimodal plots taken either from images of category 2, 3, 4 ripples, or from sub-regions of an image that have bedform orientation parallel to the sonar path, the plot remaining after the filter does not contain a single representative peak of the ripples' wavelength-and-orientation. Rather, there are multiple peaks with different wavelength frequencies and orientations across the entire plot (Figure 18B').

The second filter was designed to remove those unrealistic wavelength outputs before averaging the sub-regions into one representative value. For this data set, it was determined that an unrealistic wavelength value was greater than 0.6 meters, as no images had features scaled to this wavelength. Figure 19 plots the number of sub-regions per image post-filtration, illustrating the number of regions that had unreasonable output values. Eight images had all eight sub-regions output a reasonable value. These images all derive from the period leading up to the storm event when the hydrodynamic regime was in stage three. Two images did not produce any sub-regions with reasonable outputs. One of these images was from the storm event, and the other from stage 1 hydrodynamic conditions post-storm. The greatest number of images had six sub-region outputs remaining after this filter, with 108 images. Eighty-one images had 5 sub-regions, 77 had 7 sub-regions remaining, 44 had 4 remaining, 17 had 3 remaining, 10 had 2 remaining, and 7 had 1 remaining. The effect of removing these unreasonable outputs from the averaging procedure is illustrated in Figure 20. The plot in Figure 20A contains the wavelength values averaged from all eight sub-regions without any filtration. The values in this plot trend along a number of specific wavelength values. Fifteen of the averaged wavelength values are above the 0.6 m cutoff for reasonability. When the sub-regions are removed, the linear trends are no longer seen (Figure 20B).

5. Discussion

The dynamic and complex characteristics of the Hen and Chickens shoal data set provided the present study with a diverse series of hydrodynamic conditions and bedform

morphologies with which to develop and test an automated 2D FFT spectral analysis. A number of approaches to the procedure were performed to highlight difficulties encountered by the program within this data set. These difficulties will be discussed in this section to identify those areas of the analysis that need further development. The lessons learned and applied to address the complex H&C data set are applicable to, if not warranted for, 2D FFT analyses of other data sets. This section will discuss the results of this study in an order similar to the results section to follow the procedure for producing a reliable wavelength-and-orientation output. Attention is drawn to possible tools that may be incorporated into the spectral analysis to increase reliable automation in future studies.

First, comparison of the Fourier plots from multiple morphologies revealed that there are some morphology types in the H&C data set that are inappropriate to be analyzed for a single ripple wavelength and orientation. These plots include the bi- and multivariate plots in ripple morphology categories 3 and 4. The ability of the 2D FFT to plot multiple peaks at separate orientations and frequencies highlights the power of the analysis; however, evidence from the present study shows that it hinders the ability of the procedure to output a single representative ripple wavelength and orientation value per image. Besides morphological categories 1 and 2, which had no peaks in their Fourier plots, morphological categories 3 and 4 were the most variable in their wavelength-and-orientation outputs between sub-region arrangements. They were also the morphology categories with the greatest number of sub-regions removed from the averaging procedure due to unreasonable output values. Traykovski (2007) acknowledges problems with automated analysis of bivariate data. His solution was to utilize a manual instead of

automated analysis for all plots with bivariate distribution. The methodology in the present study did not call for the removal of the bivariate images from the data set to analyze manually. However, it is recognized that in future studies they should be removed, or utilize a new analysis procedure.

Conducting preliminary analyses on the forcing hydrodynamics in a data set may lead to the creation of a filter capable of removing these types of images from the data set before they are analyzed spectrally. In the present study, currents and waves influenced the bed in directions orthogonal to one another, thereby creating the category 2, 3, 4 ripples when the appropriate hydrodynamic conditions were present. These were stages 1 and 2 in the hydrodynamic conditions. All images from hydrodynamic stages 1 and 2, therefore, could be removed prior to analysis to rid the automated analysis of bi – multivariate plots. Similarly, the images from hydrodynamic stage 4 can also be removed, as the absence of bedforms in those images prevents them from being analyzed correctly. This leaves 94 out of 354 images to be assessed by the spectral analysis. At almost one third of the images in this data set, there is significant value in developing and using the spectral analysis. Similar hydrodynamic analyses can be conducted on data sets from other field sites to quantify the usefulness of this type of 2D FFT spectral analysis.

Results from the two filters developed for this study highlight a few further limitations of the automated 2D FFT analysis for the H&C data set, and also lead to suggestions for improvement. To start, the first of these filters was designed to remove the central peaks associated with the low-frequency noise from the outer portions of the sonar images and

with the sub-region regridding dimensions post-Fourier analysis. However, appropriate filters in future applications could remove the low frequency noise pre-analysis. This would solve the underying problem of noise and eliminate the need to create a post-Fourier analysis filter. This removal of the noise can be accomplished theoretically in three ways, none of which, however, were able to be developed or tested within this study. The first process was identified through a literature review and personal communication with committee advisor Dr. Bruce Maxwell. It deals with the windows used to perform the Fourier analysis. The sub-regions in this study were regridded in order to perform the 2D FFT in a square window. However, square windows have the effect of influencing the results of a Fourier transform due to edge effects. There are a number of window functions, such as Hamming and Gaussian windows, that can be performed prior to a Fourier transform in order to reduce these effects. These were not incorporated into the present study because the MATLAB version and license used to develop the program did not include the image processing toolbar containing the window routines.

The second procedure for removing the low-frequency noise is applicable to any data set that is characterized by regions of poor image quality, like the outer ring in the images from the H&C data set. A tool can be created to select and seemingly crop the high-quality portions of the image by replacing all data points in the low-quality areas with a single value, either zero or the average brightness of the image. Zero or the average brightness would be chosen because the 2D FFT routine analyzes images for differences

in brightness, and creating a constant color for these regions will remove their influence in the calculations.

Third, the low frequency noise can be lessened by enhancing the brightness of the ripple crests and darkening the troughs. A tool will either need to be created or adapted that is capable of identifying and tracking the crest of a ripple. Coding for such a tool was not available for, or developed within, the present study. However, fingerprint analysis technologies, currently being implemented and investigated by researchers at the University of New Hampshire and by Adam Skarke of the University of Delaware, are working towards these abilities. The tool would need to track the ripple crest data points and then increase their brightness value. It would also need to decrease the brightness of the trough data points. There would also be a second benefit from this type of tool. In the category 6 ripples, the Fourier plot did not contain one main spectral peak, but two. One spectral peak was associated with the ripple wavelength-and-orientation, while the second peak was at a slightly higher frequency and similar orientation. It is suspected that the frequency of this second peak scales to the distance from a single ripple crest to its own trough, rather than a full wavelength from one crest to another. Having better defined crests and troughs of the ripples may eliminate the second peak in the plots and create more accurate wavelength calculations by the 2D FFT.

A tool that could identify ripple crests would have a third use in a 2D FFT spectral analysis. The results of this study highlighted that regions within a single sonar image have different quality levels to their data. One such example was taken from the regions

of category 6 ripples that were parallel to the path of the sonar. In those regions, the output results were unreliable, whereas the results were more reliable in the regions where ripples are oriented orthogonal to the sonar. The second filter in this study was designed to eliminate these type of unreliable regions based on the reasonability of their wavelength outputs. However, the parameters for this filter were not extensive. The only quality was that the outputs not exceed 0.6 m, an unreasonable wavelength for this data set. Nonetheless, reasonable yet inaccurate results from these regions can pass through the filter. Therefore, these regions should be eliminated based on the quality of their original data, not the reasonability of their outputs. Similar to the strategy discussed above for removing low frequency noise in the outer regions of the image, the regions of the image that are identified to be void of ripple features can by replaced by a single value in order to remove them from being analyzed by the 2D FFT. The manual assignment of sub-region location incorporated into this study is based on a similar premise. The user avoids those regions of low quality when selecting a location for the sub-region, thereby "removing" those regions from the analysis.

If a tool for identifying and tracking ripple features were not made available, it is also possible that the hydrodynamic data could be used to develop a filter for removing these low quality regions. The low quality regions occur where ripples are parallel with the path of the sonar. Because ripple orientation is a product of the forcing hydrodynamics, the direction of flow from the hydrodynamics could be utilized to identify the regions of the image that would have features parallel to the sonar direction.

The present study's results also draw attention to the appropriateness of using an 8 sub-region arrangement versus the other four arrangements. The single sub-region and both two sub-region arrangements produced more unreasonable wavelength results than the 4 and 8 sub-region approaches. Only in images with category 6 ripples were the results of these approaches similar between the sub-region arrangements. Therefore, the applicability of each sub-region arrangement may be more a factor of the morphology of the images. In this study, the dynamic shifting between morphologies required a sub-region arrangement that could adapt to the images. The 8 sub-region arrangement covers more areas of the image and is able to delete individual sub-regions that are unreliable. Therefore, for this type of data set with multiple regions of images needing to be removed, the 8 sub-region approach provides the most initial sub-regions and is, therefore, more appropriate. Further comparison of the sub-regions is inconclusive and would need further investigation in other data sets.

Lastly, the most accurate procedure for averaging the reliable sub-region outputs into a single representative wavelength-and-orientation value is dependent on the filters and processing that is used prior in the procedure. In the present study and with the H&C data set, unreliable regions of images are identified and removed post-Fourier transform based on the reasonability criteria. Following the procedure of Voulgaris and Morin (2008), averaging all eight of the sub-regions would be inappropriate. Perhaps the spectral peaks from the unreliable regions are more dominant in the averaged plot than the peaks from the reliable regions. Then, the averaged output would be representative of the unreliable data and not the reliable regions. If, however, filters discussed above are able to remove

the unreliable regions prior to a Fourier-analysis, each and every sub-region would be reliable. Therefore, the averaging procedure of Voulgaris and Morin (2008) would be appropriate.

6. Conclusions

The MATLAB-based 2-Dimentional Fast Fourier Transform spectral analysis developed in this study is a powerful tool, but is found to have limitations in its applicability to a hydrodynamically complex field site. These limitations correspond to bi – multivariate Fourier plots stemming from certain ripple morphologies and hydrodynamic conditions. The analysis is most reliable in images with well-defined linear ripple features. Future studies that incorporate an automated spectral analysis may benefit from conducting a preliminary analysis of the hydrodynamic conditions and ripple morphologies. When hydrodynamic conditions form ripples with a single orientation and wavelength, the automated analysis is appropriate. Further work is recommended to develop tools which will be able to identify images and sub-regions to images that contain ripple features and to pre-process the data to either delete regions without ripple features and/or to better highlight the crests and troughs of the ripple features. Without these tools for making the spectral analysis more reliable, studies will need to perform similar quality assessments similar to the present study.

References

Brunsden, D., 2002. Geomorphological roulette for engineers and planners: some insights into an old game. Quarterly Journal of Engineering Geology and Hydrogeology 35, 101-142.

Cacchione, D.A., Sternberg, R.W., Ogston, A.S., 2006. Bottom Instrumented tripods: History, applications, and impacts. Continental Shelf Research 26, 2319-2334.

Catano-Lopera, Y.A., Abad, J.D., Garcia, M.H., 2009. Characterization of bedform morphology generated under combined flows and currents using wavelet analysis. Ocean Engineering 36, 617-632.

Cooley, J.W., Tukey, J.W., 1965. An Algorithm for the Machine Computation of the Complex Fourier Series. Mathematics of Computation 19, 297-301.

Davis Jr, R.A., FitzGerald, D.M., 2004. Beaches and Coasts. Blackwell Science Ltd, Malden, MA., 404p.

Davies, A.G., Thorne, P.D., 2008. Advances in the study of moving sediments and evolving seabeds. Surveys in Geophysics 29, 1-36.

Day, J.W., Boesch, D.F., Clairain, E.J., Kemp, G.P., Laska, S.B., Mitsch, W.J., Orth, K., Mashriqui, H., Reed, D.J., Shabman, L., Simestad, C.A., Streever, B.J., Twilley, R.R., Watson, C.C., Wells, J.T., Whigham, D.F., 2007. Restoration of the Mississippi Delta: Lessons from Hurricanes Katrina and Rita. Science 315, 1679-1684.

Dean, R.G., Dalrymple, R.A., 2004. Coastal processes with engineering applications. Cambridge University Press, New York, NY. 475p.

Doucette, J.S., 2002. Geometry and grain-size sorting of ripples on low-energy sandy beaches: field observations and model predictions. Sedimentology 49, 483-503.

Frigo, M., Johnson, S.G., 1998. FFTW: An Adaptive Software Architecture for the FFT. Proceedings of the International Confrence on Acoustics, Speech, and Signal Processing 3, 1381-1384.

Garcia, M.H., 2000. The legend of A.F. Shields. Journal of Hydraulic Engineering-ASCE 126, 718-720.

Gilbert, G.K., 1914. The transportation of débris by running water. Profl. Paper No. 86, U.S. Geological Survey, Washington, D.C.

Grant, W.D., Madsen, O.S., 1982. Movable bed roughness in unsteady oscillatory flow. Journal of Geophysical Research 87, 469-481.

Interagency Ocean Policy Task Force, 2009. Interim Report of the Interagency Ocean Policy Task Force. White House Council on Environmental Quality. Available: http://www.whitehouse.gov/assets/documents/09_17_09_Interim_Report_of_Task_Force_FINAL2.pdf

Irish, J.D., Lynch, J.F., Traykovski, P.A., Newhall, A.E., Prada, K., Hay, A.E., 1998. A Self-Contained Sector-Scanning Sonar for Bottom Roughness Observations as Part of Sediment Transport Studies. Journal of Atmospheric and Oceanic Technology 16, 1830-1841.

Jiménez, J.A., Sánchez-Arcilla, A., Valemoro, H.I., Gracia, V., Nieto, F., 1997. Processes reshaping the Ebro Delta. Marine Geology 144, 59-79.

Krogstad, H.E., 2004. How to use the MATLAB FFT2-routines. Unpublished Report, http://www.mathworks.com/matlabcentral/files/11639

Lehar, S., 2010. An Intuitive Explanation of Fourier Theory. Online resource. Available: http://cns-alumni.bu.edu/~slehar/fourier/fourier.html

Li, M., and O'Connor, B.A., 2007. Numerical study of sediment transport above rippled beds under the action of combined waves and currents. Sedimentology 54, 1345-1363.

Lopes, J.L., Lim, R., Commander, K.W., 2004. Mid Frequency Sonar Backscatter Measurements from a Rippled Bottom. High Frequency Ocean Acoustics Conference. AIP Conference Proceedings 728, 141-148.

Matlab, 2008. fft2: 2-D discrete Fourier transform, Algorithm. Help menu. Matlab version 7.7.0 (R2008b).

Nielsen, P., 1981. Dynamics and geometry of wave-generated ripples. Journal of Geophysical Research 86, 6467-6472.

Pitt, R., 2004. Sediment transport in storm drainage systems. Online resource. Available: http://rpitt.eng.ua.edu/Class/StormWaterManagement/M3a%20Characteristics%20and%20Sources%20Internet%20material/M3a3%20sediment%20movement%20in%20sewers.htm

Ritter, D.F., Kochel, R.C., Miller, J.R., 2006. Process Geomorphology 4th Edition. Waveland Press, Inc., Long Grove, IL. 546 p.

Roelvink, J.A., Stive, M.J.F., 1990. Sand transport on the shoreface of the Holland Coast. Proc. 22nd ICCEDelft, ASCE, New York 1990, 5.1-5.13.

Sekiguchi, T., Sunamura, T., 2004. A laboratory study of formative conditions for

characteristic ripple patterns associated with a change in wave conditions. Earth Surface Processes and Landforms 29, 1431-1435.

Shields, A., 1936. Application of similarity principles and turbulence research to bed-load movement. Hydrodynamics Laboratory Publ. No. 167, Ott, W. P., van Uchelen, J. C. trans., U.S. Dept. of Agr., Soil Conservation Service Cooperative Laboratory, California Institute of Technology, Pasadena, Calif.

Soulsby, R., 1997. Dynamics of marine sands. London, Thomas Telford. 249 p.

Stokes, G.G. 1851. On the effect of the internal friction on the motion of pendulums. Transactions of the Cambridge Philosophical Society 9, 8–106.

Thorne, P.D., Hanes, D.M., 2001. A review of acoustic measurement of small-scale sediment processes. Continental Shelf Research 22, 603-632.

Traykovski, P., 2007. Observations of wave orbital scale ripples and a nonequilibrium time-dependant model. Journal of Geophysical Research 112: C06026.

Traykovski, P., Hay, A.E., Irish, J.D., Lynch, J.F., 1999. Geometry, migration, and evolution of wave orbital ripples at LEO-15. Journal of Geophysical Research 104, 1505-1524.

Voulgaris, G., Morin, J.P., 2008. A long-term real time sea bed morphology evolution system in the South Atlantic Bight. Proceeding of the IEEE/OES/CMTC Ninth Working Conference on Current Measurement Technology.

Wiberg, P.L., Harris, C.K., 1994. Ripple geometry in wave-dominated environments. Journal of Geophysical Research 99, 775-789.

Wynn, R.B., Masson, D.G., Bett, B.J., 2002. Hydrodynamic significance of variable ripple morphology across deep-water barchan dunes in the Faroe-Shetland Channel. Marine Geology 192, 309-319.

Table 1. Directory for images being processed by the MATLAB program. # is the sample number of the image; ii # is the index of the image in the data set; date and time are the recorded data and time for the image. Rows separated by a red line indicate non-consecutive images.

#	ii #	Date and time	#	ii #	Date and time	#	ii #	Date and time	#	ii #	Date and time
1	22	8/21 1900	**46**	202	8/23 1600	**91**	382	8/25 1300	**136**	562	8/27 1000
2	26	8/21 2000	**47**	206	8/23 1700	**92**	386	8/25 1400	**137**	566	8/27 1100
3	30	8/21 2100	**48**	210	8/23 1800	**93**	390	8/25 1500	**138**	570	8/27 1200
4	34	8/21 2200	**49**	214	8/23 1900	**94**	394	8/25 1600	**139**	574	8/27 1300
5	38	8/21 2300	**50**	218	8/23 2000	**95**	398	8/25 1700	**140**	578	8/27 1400
6	42	8/22 0000	**51**	222	8/23 2100	**96**	402	8/25 1800	**141**	582	8/27 1500
7	46	8/22 0100	**52**	226	8/23 2200	**97**	406	8/25 1900	**142**	586	8/27 1600
8	50	8/22 0200	**53**	230	8/23 2300	**98**	410	8/25 2000	**143**	590	8/27 1700
9	54	8/22 0300	**54**	234	8/24 0000	**99**	414	8/25 2100	**144**	594	8/27 1800
10	58	8/22 0400	**55**	238	8/24 0100	**100**	418	8/25 2200	**145**	598	8/27 1900
11	62	8/22 0500	**56**	242	8/24 0200	**101**	422	8/25 2300	**146**	602	8/27 2000
12	66	8/22 0600	**57**	246	8/24 0300	**102**	426	8/26 0000	**147**	606	8/27 2100
13	70	8/22 0700	**58**	250	8/24 0400	**103**	430	8/26 0100	**148**	610	8/27 2200
14	74	8/22 0800	**59**	254	8/24 0500	**104**	434	8/26 0200	**149**	614	8/27 2300
15	78	8/22 0900	**60**	258	8/24 0600	**105**	438	8/26 0300	**150**	618	8/28 0000
16	82	8/22 1000	**61**	262	8/24 0700	**106**	442	8/26 0400	**151**	622	8/28 0100
17	86	8/22 1100	**62**	266	8/24 0800	**107**	446	8/26 0500	**152**	626	8/28 0200
18	90	8/22 1200	**63**	270	8/24 0900	**108**	450	8/26 0600	**153**	630	8/28 0300
19	94	8/22 1300	**64**	274	8/24 1000	**109**	454	8/26 0700	**154**	634	8/28 0400
20	98	8/22 1400	**65**	278	8/24 1100	**110**	458	8/26 0800	**155**	638	8/28 0500
21	102	8/22 1500	**66**	282	8/24 1200	**111**	462	8/26 0900	**156**	642	8/28 0600
22	106	8/22 1600	**67**	286	8/24 1300	**112**	466	8/26 1000	**157**	646	8/28 0700
23	110	8/22 1700	**68**	290	8/24 1400	**113**	470	8/26 1100	**158**	650	8/28 0800
24	114	8/22 1800	**69**	294	8/24 1500	**114**	474	8/26 1200	**159**	654	8/28 0900
25	118	8/22 1900	**70**	298	8/24 1600	**115**	478	8/26 1300	**160**	658	8/28 1000
26	122	8/22 2000	**71**	302	8/24 1700	**116**	482	8/26 1400	**161**	662	8/28 1100
27	126	8/22 2100	**72**	306	8/24 1800	**117**	486	8/26 1500	**162**	666	8/28 1200
28	130	8/22 2200	**73**	310	8/24 1900	**118**	490	8/26 1600	**163**	670	8/28 1300
29	134	8/22 2300	**74**	314	8/24 2000	**119**	494	8/26 1700	**164**	674	8/28 1400
30	138	8/23 0000	**75**	318	8/24 2100	**120**	498	8/26 1800	**165**	678	8/28 1500
31	142	8/23 0100	**76**	322	8/24 2200	**121**	502	8/26 1900	**166**	682	8/28 1600
32	146	8/23 0200	**77**	326	8/24 2300	**122**	506	8/26 2000	**167**	686	8/28 1700
33	150	8/23 0300	**78**	330	8/25 0000	**123**	510	8/26 2100	**168**	690	8/28 1800
34	154	8/23 0400	**79**	334	8/25 0100	**124**	514	8/26 2200	**169**	694	8/28 1900
35	158	8/23 0500	**80**	338	8/25 0200	**125**	518	8/26 2300	**170**	698	8/28 2000
36	162	8/23 0600	**81**	342	8/25 0300	**126**	522	8/27 0000	**171**	702	8/28 2100
37	166	8/23 0700	**82**	346	8/25 0400	**127**	526	8/27 0100	**172**	706	8/28 2200
38	170	8/23 0800	**83**	350	8/25 0500	**128**	530	8/27 0200	**173**	710	8/28 2300
39	174	8/23 0900	**84**	354	8/25 0600	**129**	534	8/27 0300	**174**	714	8/29 0000
40	178	8/23 1000	**85**	358	8/25 0700	**130**	538	8/27 0400	**175**	718	8/29 0100
41	182	8/23 1100	**86**	362	8/25 0800	**131**	542	8/27 0500	**176**	722	8/29 0200
42	186	8/23 1200	**87**	366	8/25 0900	**132**	546	8/27 0600	**177**	726	8/29 0300
43	190	8/23 1300	**88**	370	8/25 1000	**133**	550	8/27 0700	**178**	730	8/29 0400
44	194	8/23 1400	**89**	374	8/25 1100	**134**	554	8/27 0800	**179**	734	8/29 0500
45	198	8/23 1500	**90**	378	8/25 1200	**135**	558	8/27 0900	**180**	738	8/29 0600

181	742	8/29 0700	**225**	918	8/31 0300	**269**	1094	9/1 2300	**313**	1270	9/3 1900
182	746	8/29 0800	**226**	922	8/31 0400	**270**	1098	9/2 0000	**314**	1274	9/3 2000
183	750	8/29 0900	**227**	926	8/31 0500	**271**	1102	9/2 0100	**315**	1278	9/3 2100
184	754	8/29 1000	**228**	930	8/31 0600	**272**	1106	9/2 0200	**316**	1282	9/3 2200
185	758	8/29 1100	**229**	934	8/31 0700	**273**	1110	9/2 0300	**317**	1286	9/3 2300
186	762	8/29 1200	**230**	938	8/31 0800	**274**	1114	9/2 0400	**318**	1290	9/4 0000
187	766	8/29 1300	**231**	942	8/31 0900	**275**	1118	9/2 0500	**319**	1294	9/4 0100
188	770	8/29 1400	**232**	946	8/31 1000	**276**	1122	9/2 0600	**320**	1298	9/4 0200
189	774	8/29 1500	**233**	950	8/31 1100	**277**	1126	9/2 0700	**321**	1302	9/4 0300
190	778	8/29 1600	**234**	954	8/31 1200	**278**	1130	9/2 0800	**322**	1306	9/4 0400
191	782	8/29 1700	**235**	958	8/31 1300	**279**	1134	9/2 0900	**323**	1310	9/4 0500
192	786	8/29 1800	**236**	962	8/31 1400	**280**	1138	9/2 1000	**324**	1314	9/4 0600
193	790	8/29 1900	**237**	966	8/31 1500	**281**	1142	9/2 1100	**325**	1318	9/4 0700
194	794	8/29 2000	**238**	970	8/31 1600	**282**	1146	9/2 1200	**326**	1326	9/4 0900
195	798	8/29 2100	**239**	974	8/31 1700	**283**	1150	9/2 1300	**327**	1330	9/4 1000
196	802	8/29 2200	**240**	978	8/31 1800	**284**	1154	9/2 1400	**328**	1334	9/4 1100
197	806	8/29 2300	**241**	982	8/31 1900	**285**	1158	9/2 1500	**329**	1338	9/4 1200
198	810	8/30 0000	**242**	986	8/31 2000	**286**	1162	9/2 1600	**330**	1342	9/4 1300
199	814	8/30 0100	**243**	990	8/31 2100	**287**	1166	9/2 1700	**331**	1346	9/4 1400
200	818	8/30 0200	**244**	994	8/31 2200	**288**	1170	9/2 1800	**332**	1350	9/4 1500
201	822	8/30 0300	**245**	998	8/31 2300	**289**	1174	9/2 1900	**333**	1362	9/4 1800
202	826	8/30 0400	**246**	1002	9/1 0000	**290**	1178	9/2 2000	**334**	1370	9/4 2000
203	830	8/30 0500	**247**	1006	9/1 0100	**291**	1182	9/2 2100	**335**	1374	9/4 2100
204	834	8/30 0600	**248**	1010	9/1 0200	**292**	1186	9/2 2200	**336**	1378	9/4 2200
205	838	8/30 0700	**249**	1014	9/1 0300	**293**	1190	9/2 2300	**337**	1386	9/5 0000
206	842	8/30 0800	**250**	1018	9/1 0400	**294**	1194	9/3 0000	**338**	1390	9/5 0100
207	846	8/30 0900	**251**	1022	9/1 0500	**295**	1198	9/3 0100	**339**	1406	9/5 0500
208	850	8/30 1000	**252**	1026	9/1 0600	**296**	1202	9/3 0200	**340**	1410	9/5 0600
209	854	8/30 1100	**253**	1030	9/1 0700	**297**	1206	9/3 0300	**341**	1422	9/5 0900
210	858	8/30 1200	**254**	1034	9/1 0800	**298**	1210	9/3 0400	**342**	1426	9/5 1000
211	862	8/30 1300	**255**	1038	9/1 0900	**299**	1214	9/3 0500	**343**	1430	9/5 1100
212	866	8/30 1400	**256**	1042	9/1 1000	**300**	1218	9/3 0600	**344**	1434	9/5 1200
213	870	8/30 1500	**257**	1046	9/1 1100	**301**	1222	9/3 0700	**345**	1442	9/5 1400
214	874	8/30 1600	**258**	1050	9/1 1200	**302**	1226	9/3 0800	**346**	1446	9/5 1600
215	878	8/30 1700	**259**	1054	9/1 1300	**303**	1230	9/3 0900	**347**	1450	9/5 1700
216	882	8/30 1800	**260**	1058	9/1 1400	**304**	1234	9/3 1000	**348**	1454	9/5 1800
217	886	8/30 1900	**261**	1062	9/1 1500	**305**	1238	9/3 1100	**349**	1458	9/5 1900
218	890	8/30 2000	**262**	1066	9/1 1600	**306**	1242	9/3 1200	**350**	1462	9/5 2000
219	894	8/30 2100	**263**	1070	9/1 1700	**307**	1246	9/3 1300	**351**	1486	9/6 0100
220	898	8/30 2200	**264**	1074	9/1 1800	**308**	1250	9/3 1400	**352**	1490	9/6 0200
221	902	8/30 2300	**265**	1078	9/1 1900	**309**	1254	9/3 1500	**353**	1502	9/6 0500
222	906	8/31 0000	**266**	1082	9/1 2000	**310**	1258	9/3 1600	**354**	1518	9/6 0900
223	910	8/31 0100	**267**	1086	9/1 2100	**311**	1262	9/3 1700			
224	914	8/31 0200	**268**	1090	9/1 2200	**312**	1266	9/3 1800			

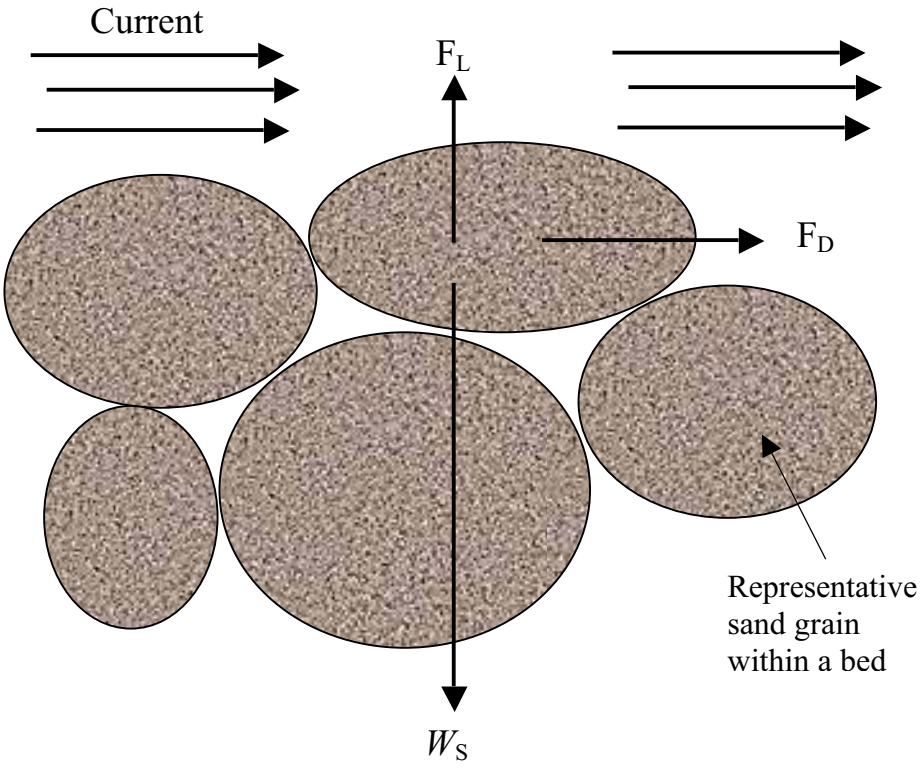

Figure 1. Forces on a sand particle in a bed. F_D is the drag force, F_L is the lift force, and W_S is the weight of the particle acting downwards (after Dean and Dalrymple, 2004).

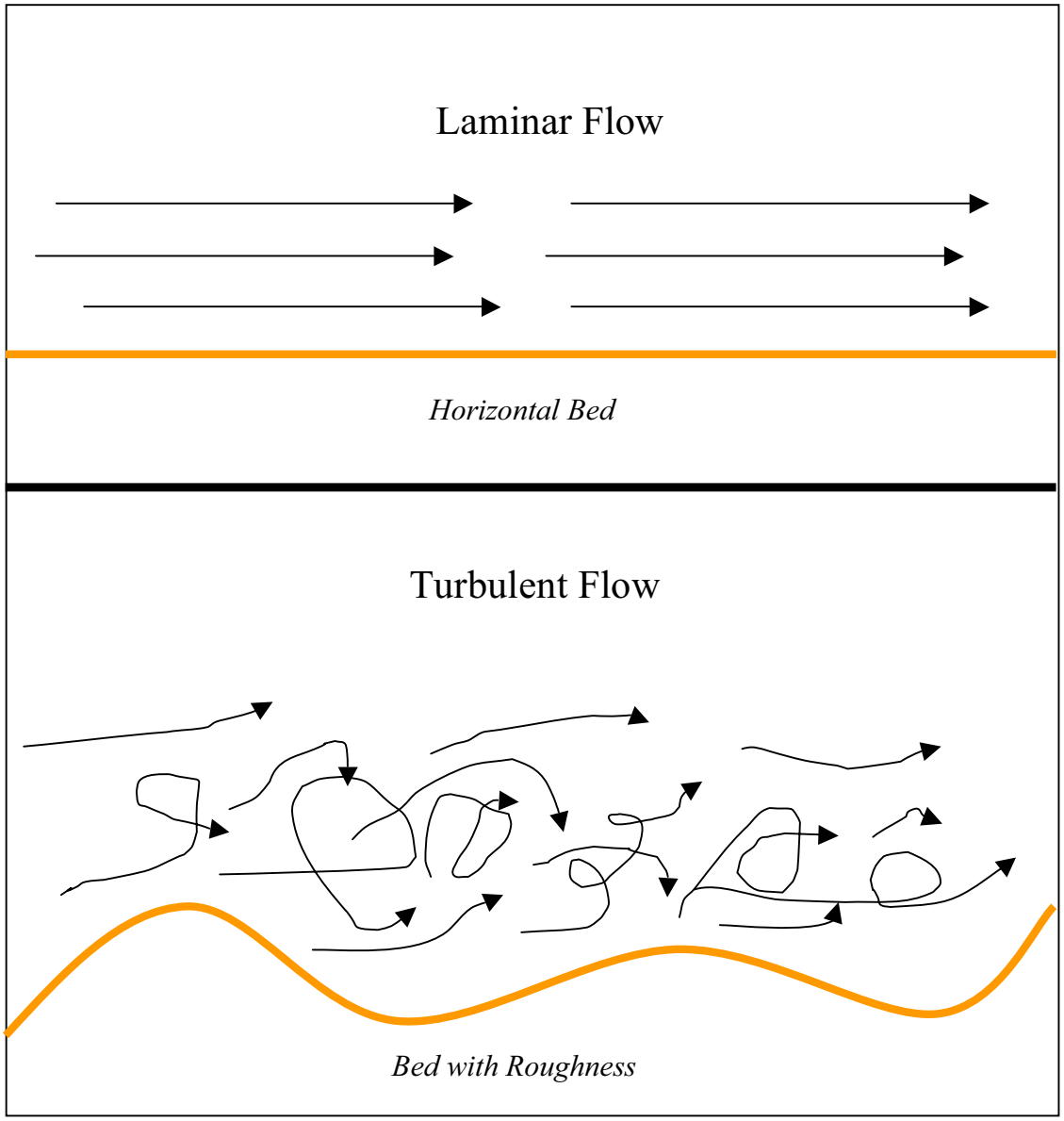

Figure 2. Unidirectional flow: laminar versus turbulent. The roughness of the bed acts to disturb the flow and induce turbulent flow.

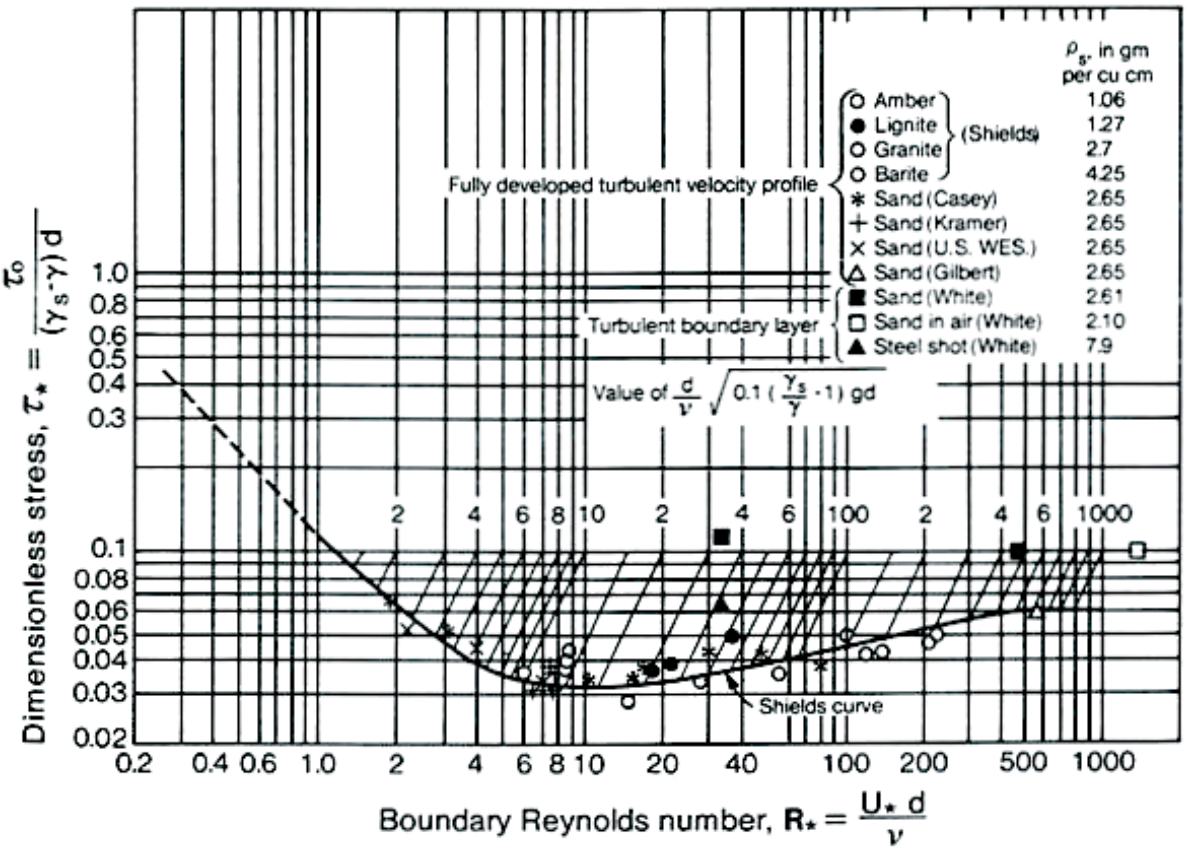

Figure 3. Parameters for the initiation of particle motion as a function of dimensionless critical shear stress and boundary Reynolds number. Actualistic studies tested for particle motion using different grains and flow regimes. When the results from these studies are combined, a curve can be fit to the data, known as the Shields curve. Movement of grains will occur for conditions above the Shields curve (from Pitt, 2004).

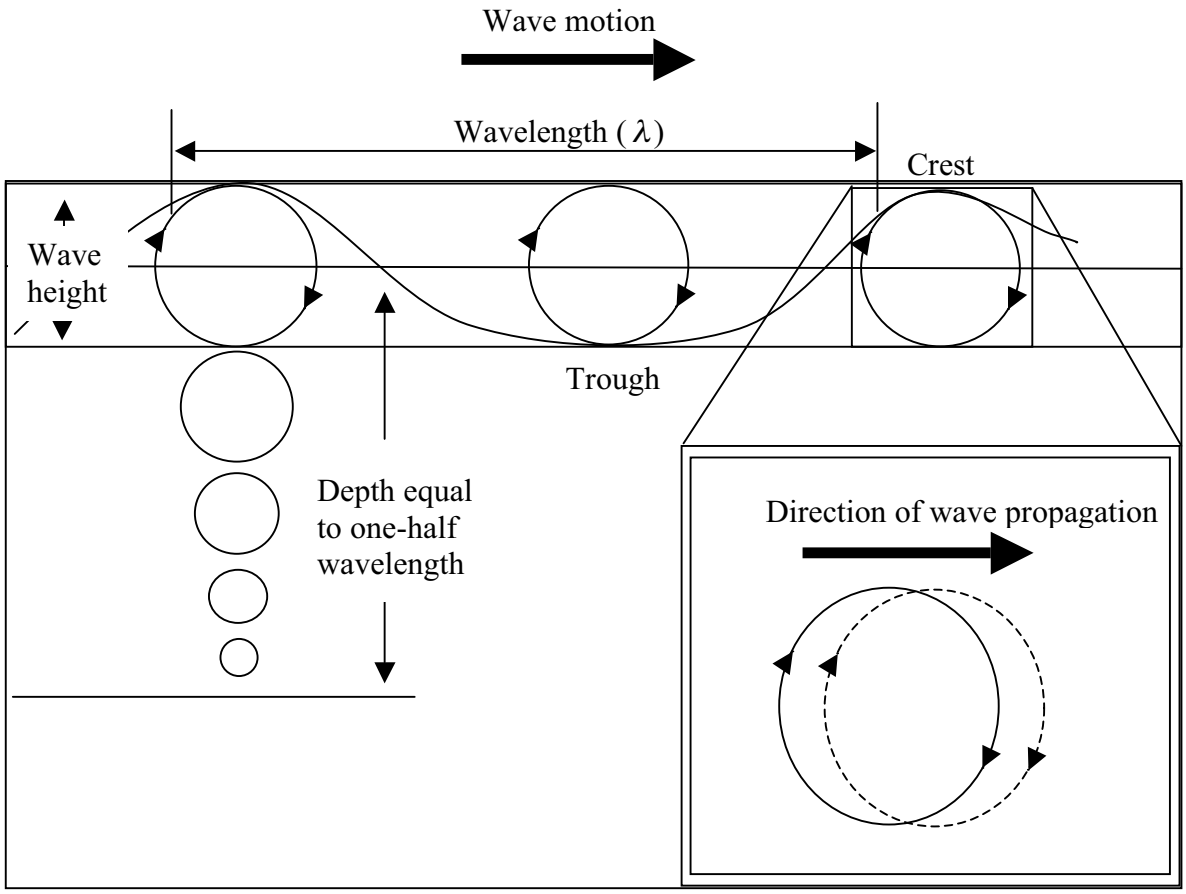

Figure 4. Water motion and wave-energy propagation in an orbital wave. Diagram shows the decrease in the diameter of the orbital paths with depth and the slight landward movement of water due to wind friction (after Davis and FitzGerald, 2004).

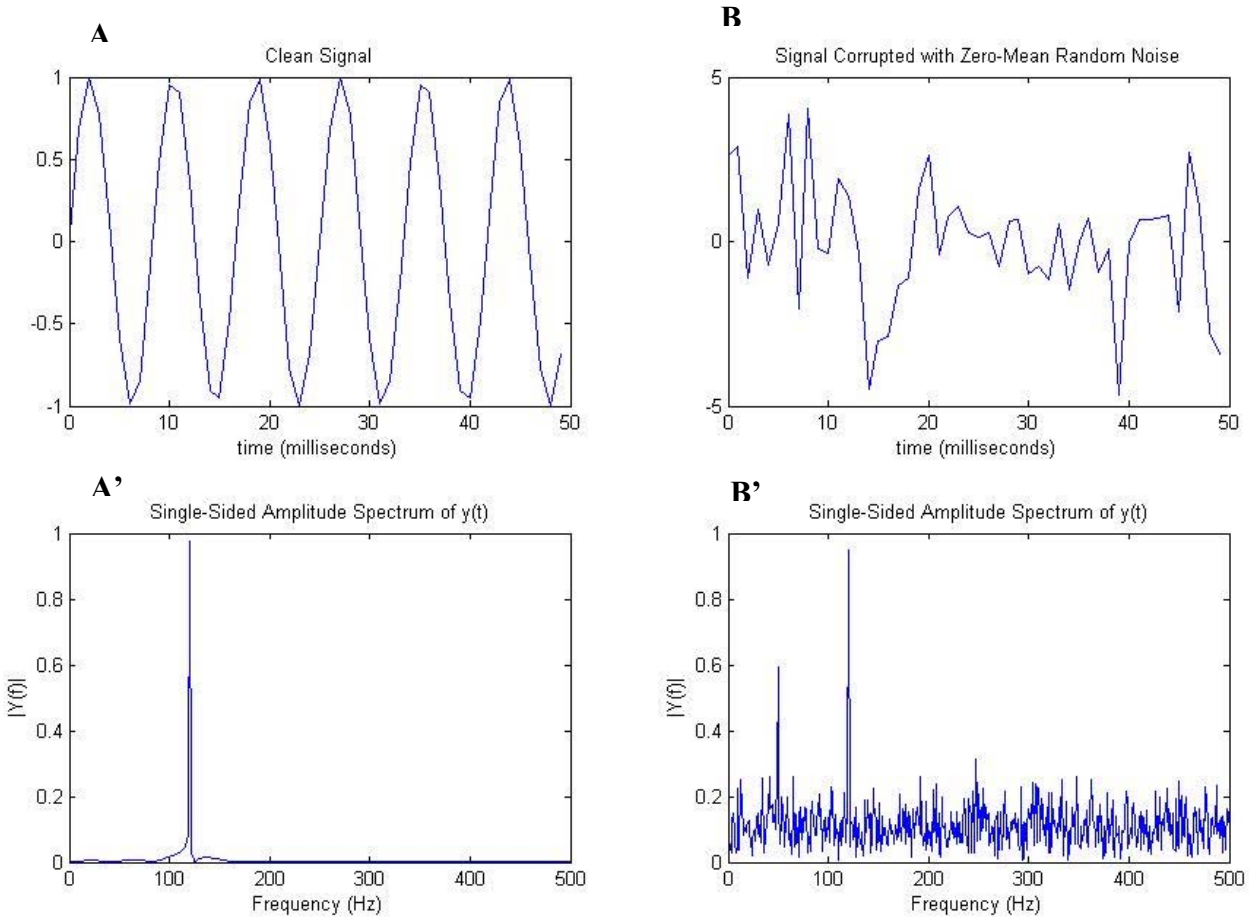

Figure 5. Discrete Fourier Transform (DFT) calculated for representative sinusoidal signals. A: Clean signal with one frequency. A': DFT has one distinct peak at frequency 120. B: Two sinusoidal signals corrupted with a random noise. B': The DFT identifies the frequencies of the two signals with distinct peaks, while the random noise has multitudes of smaller peaks.

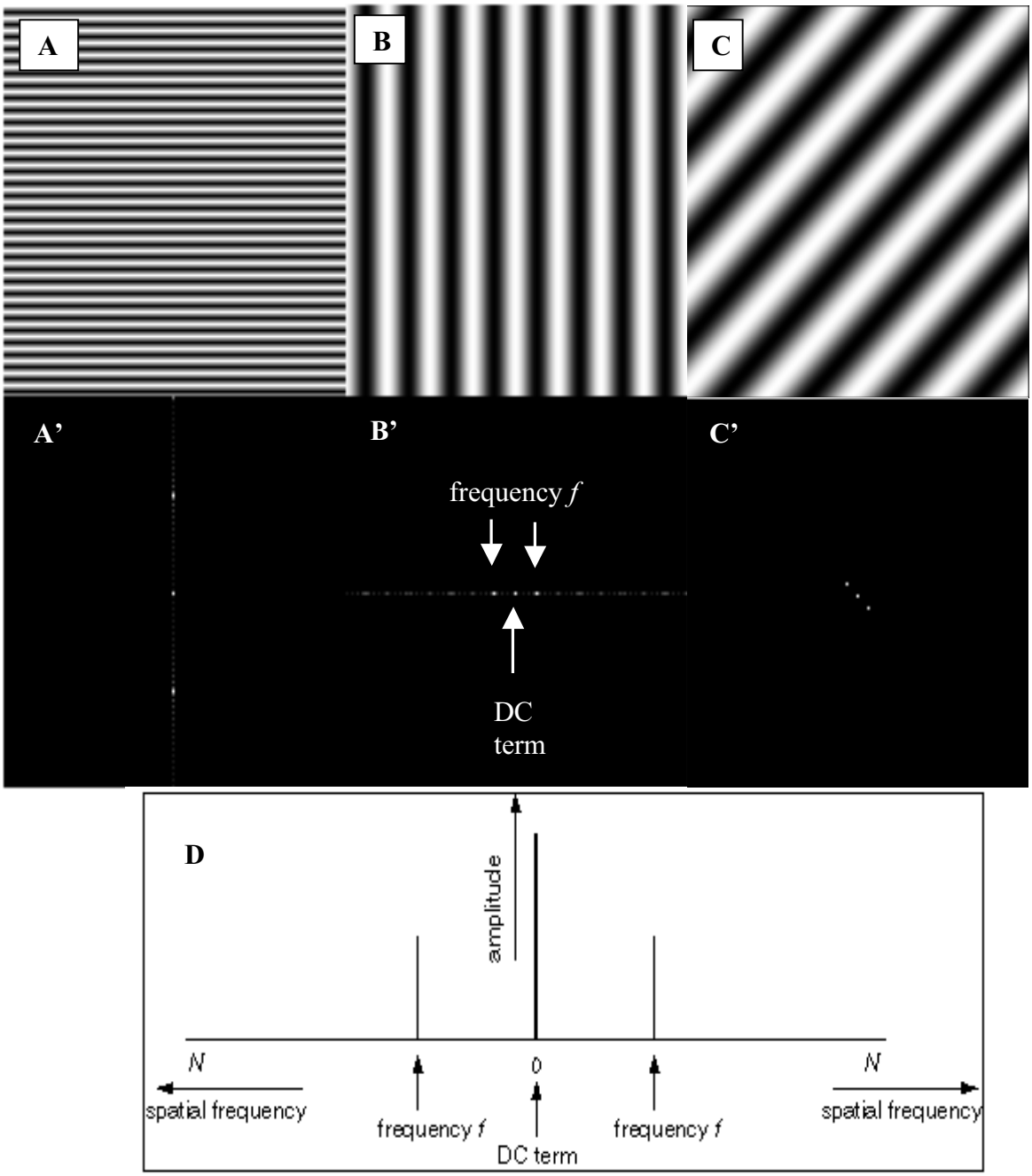

Figure 6. 2-Dimensional Fast Fourier Transform calculated for three representative images. A,B,C: Image with one frequency component. A',B',C': Fourier transform plots of each image. The frequency identified by the Fourier transform is displayed along with a DC term at the origin. D: Profile view of what is being displayed from above in A',B',C' (after Lehar, 2010).

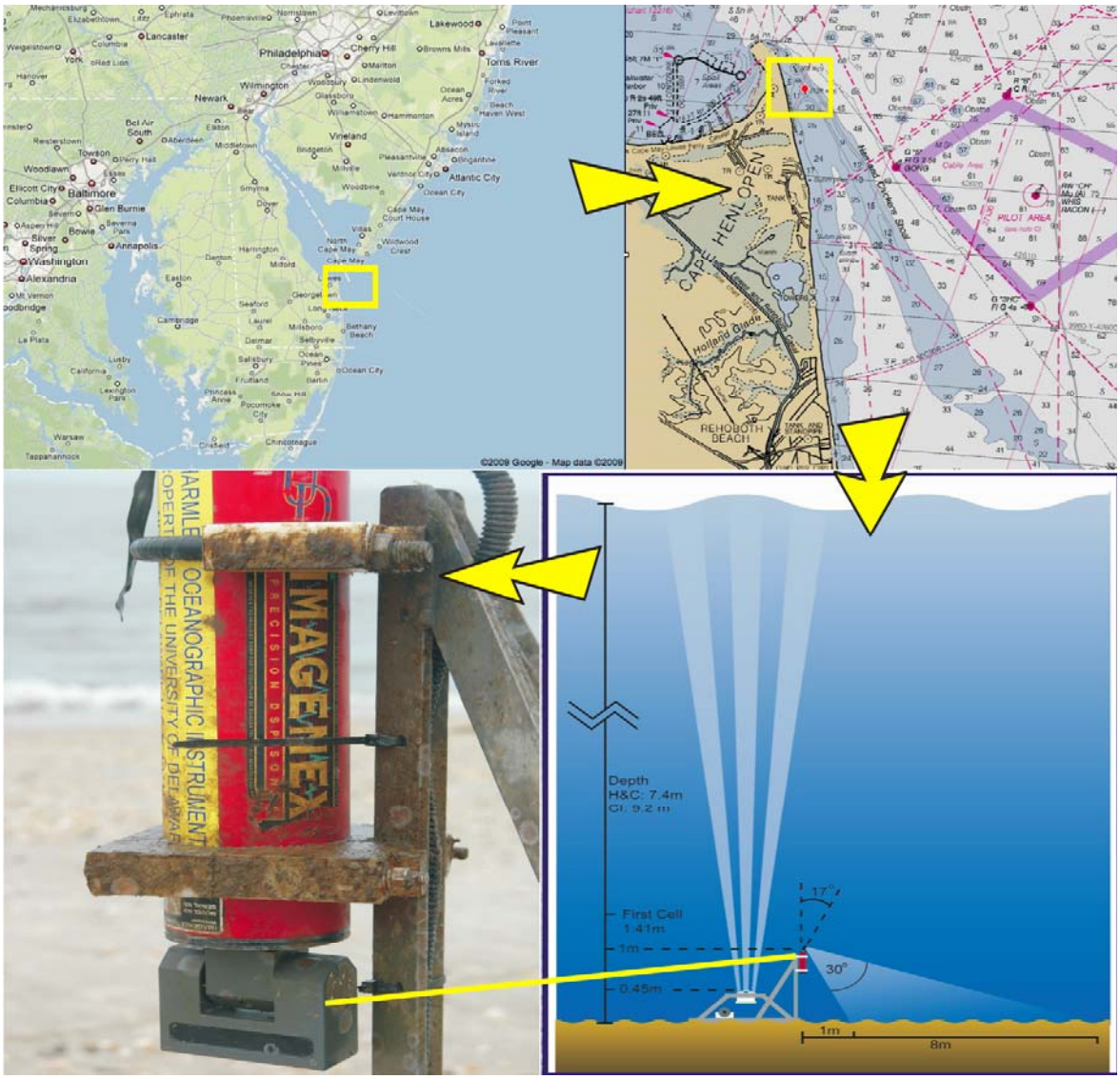

Figure 7. Map of the field site at Hen and Chickens Shoal and schematic of the instrumented frame used during the *in situ* bedform morphology and hydrodynamic study.

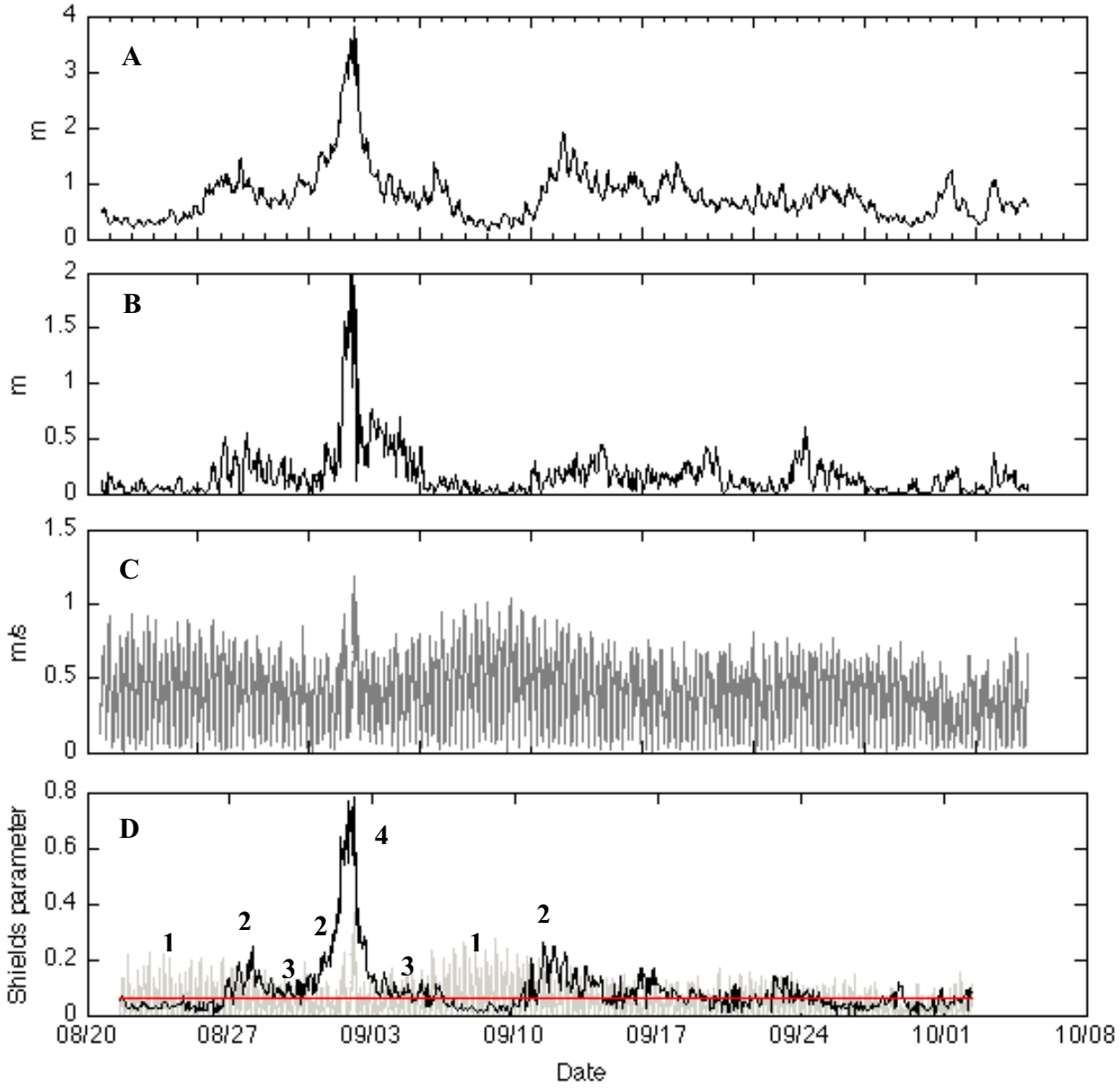

Figure 8. Wave and current hydrodynamic conditions plotted throughout the study period. A: Wave height B: Wave orbital diameter C: Near-bed mean current velocity D: Wave (black line) and current (gray line) Shields parameter. The red line represents the critical Shields parameter. The hydrodynamic conditions can be split into three stages. 1: background with wave and current competing to be dominant, 2: wave energy dominant over with current energy 3: wave and current energy competing to be dominant during flood and ebb stages 4: storm event where wave energy greatly surpasses the current energy.

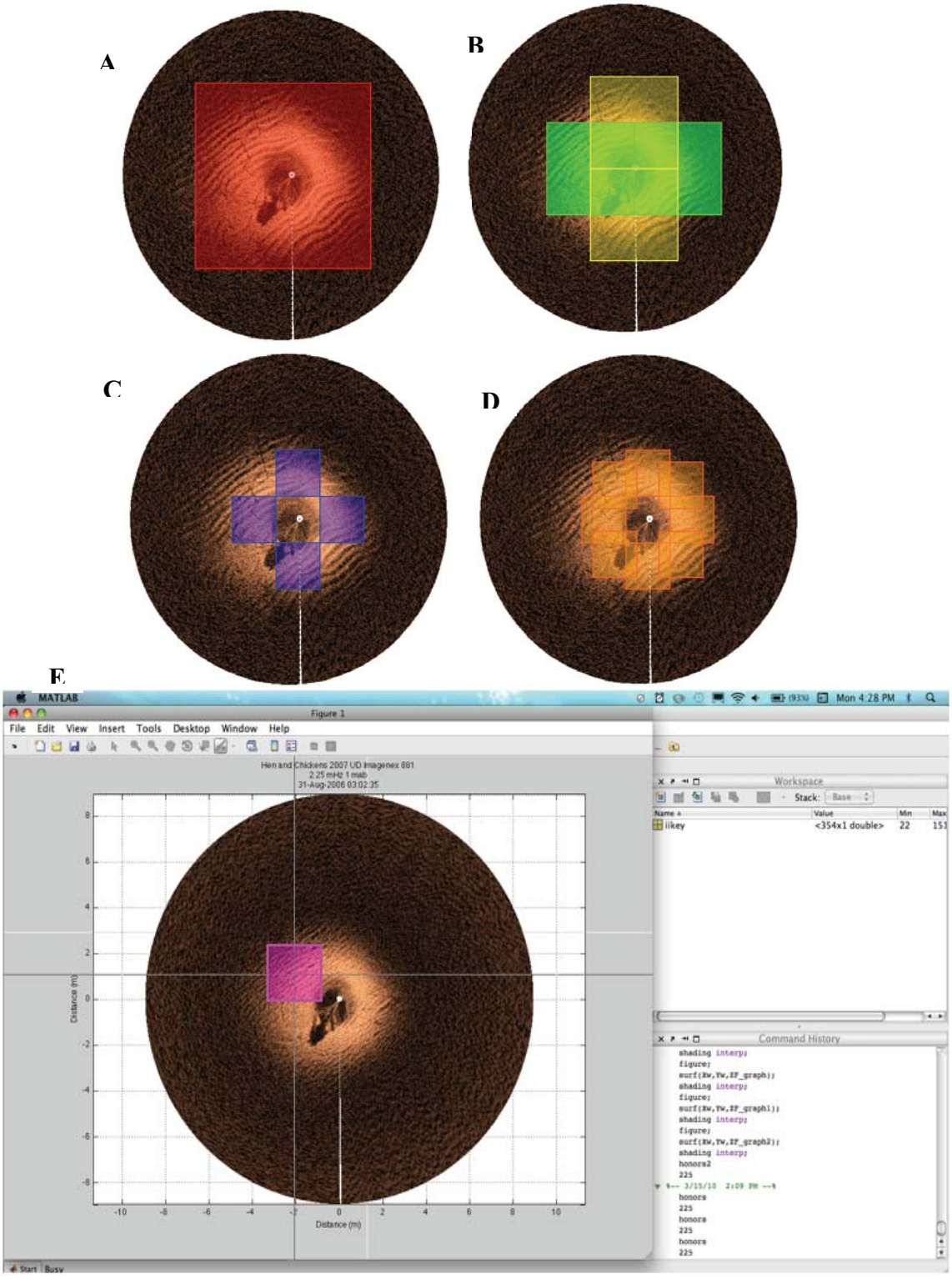

Figure 9. Sub-region size and location for the six arrangements developed within the present study. A.1 sub-region. B. 2 sub-regions. C. 4 sub-regions. D. 8 sub-regions. E. Manual assigning of sub-region.

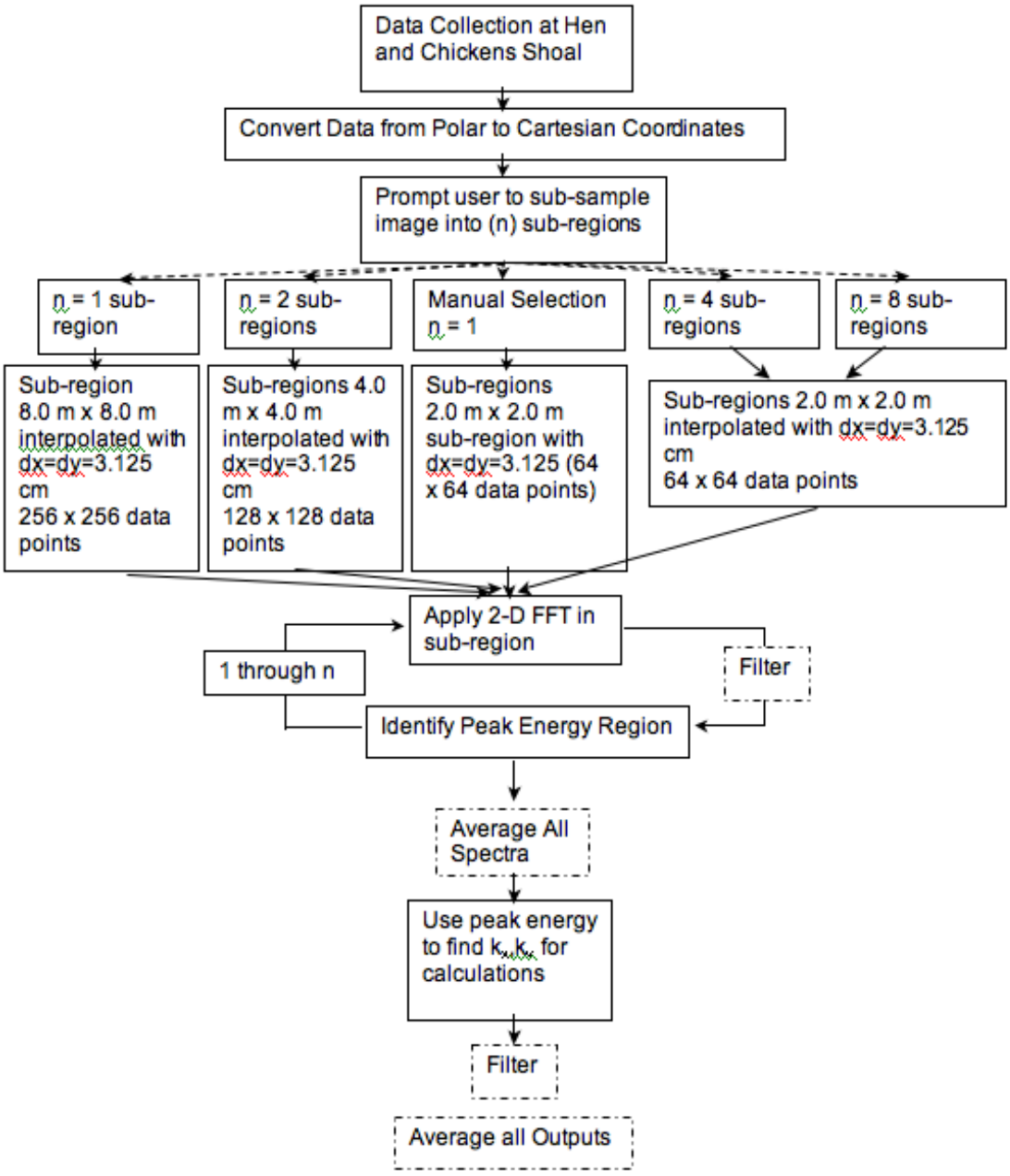

Figure 10. Flow chart of the spectral analysis procedure. Boxes that are dotted lines are steps which can be altered, and would need further investigation to solidify.

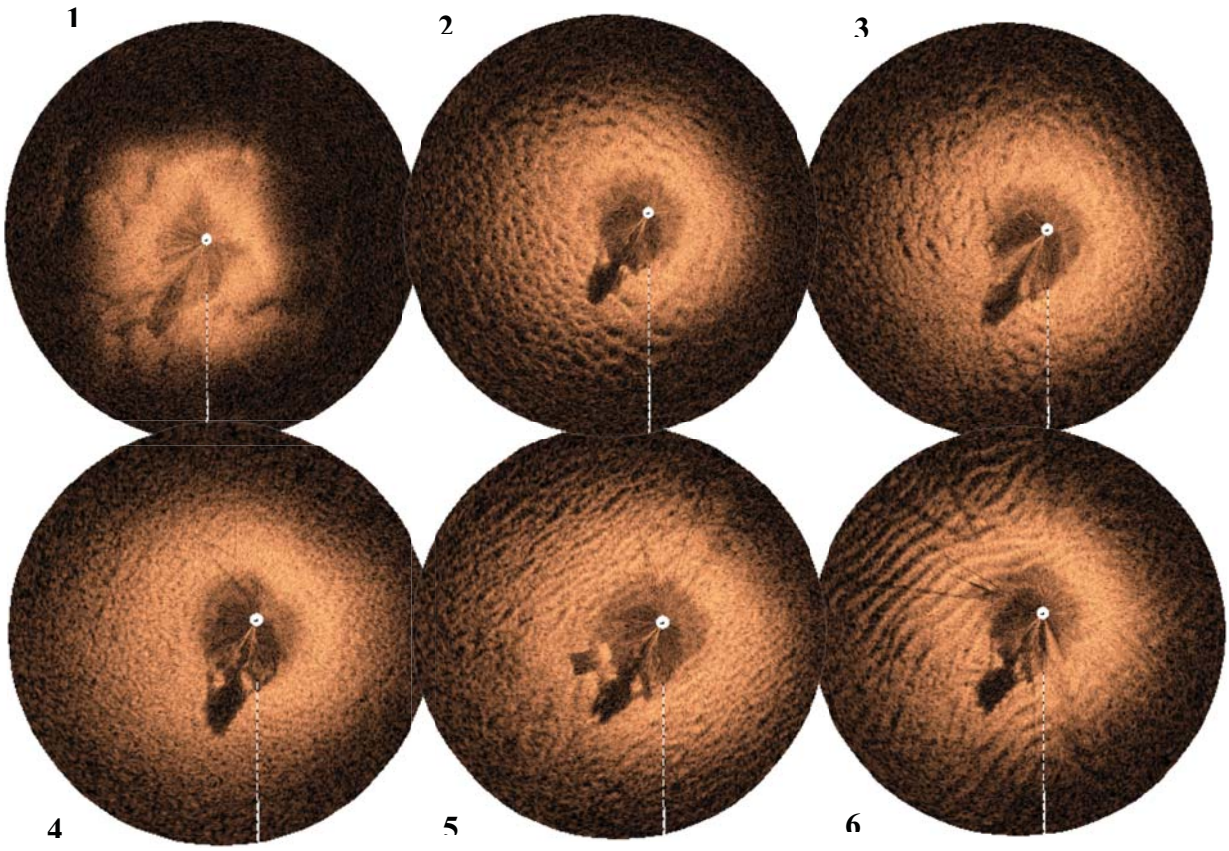

Figure 11. Representative images of the six bedform morphology categories analyzed during the study. (1) No recognizable bedforms (mostly due to distortion during the storm event) (2) large scale strong interference ripples in a diamond-shaped hummocky pattern, (3) small scale interference ripples with interspersed areas of a quasi-linear arrangement, (4) quasi-linear arrangement of interference patterns, (5) complex overlapping ripples, (6) complex ripples with well-defined crests.

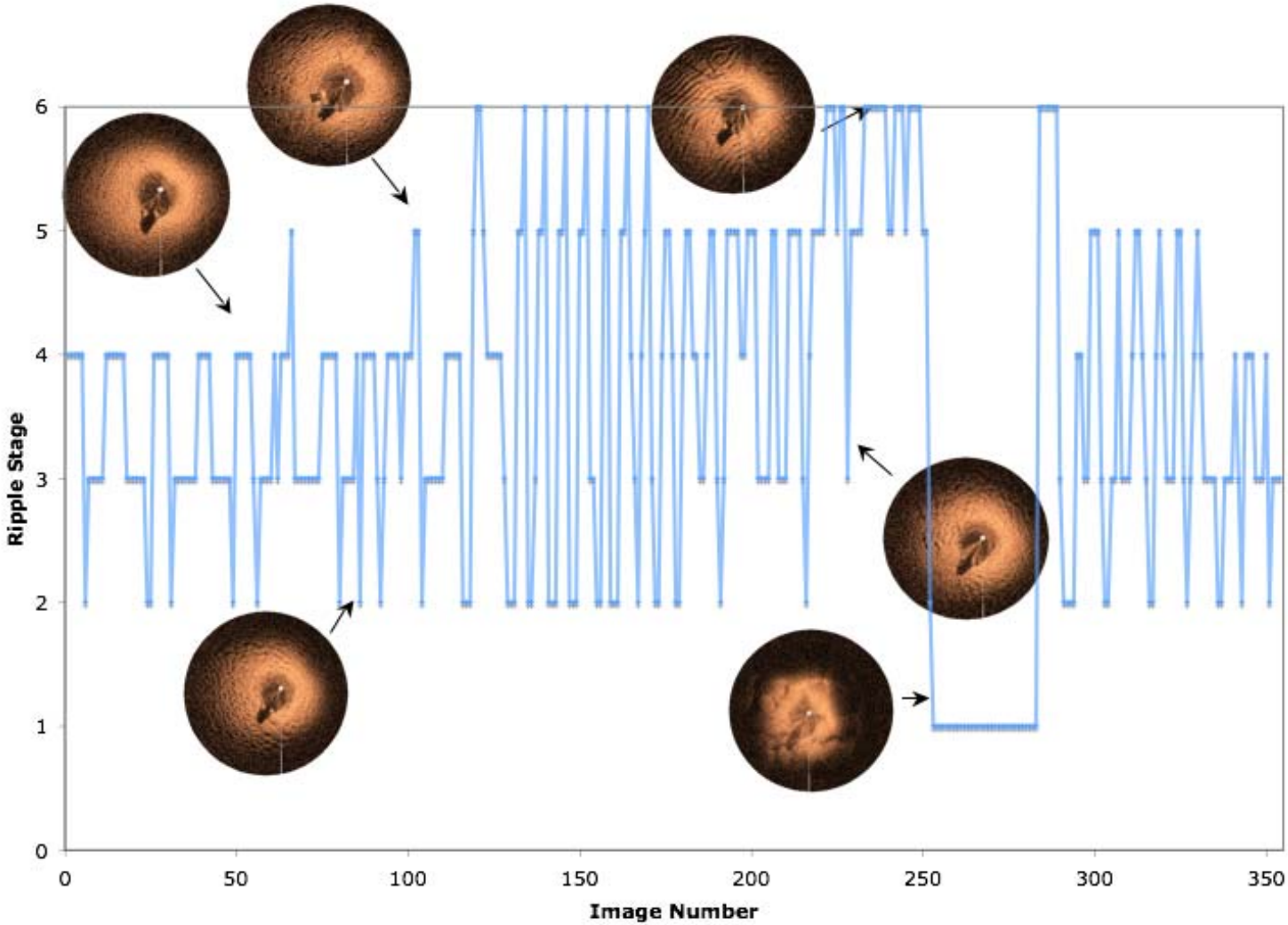

Figure 12. Evolution of bed morphology categories over the duration of the study.

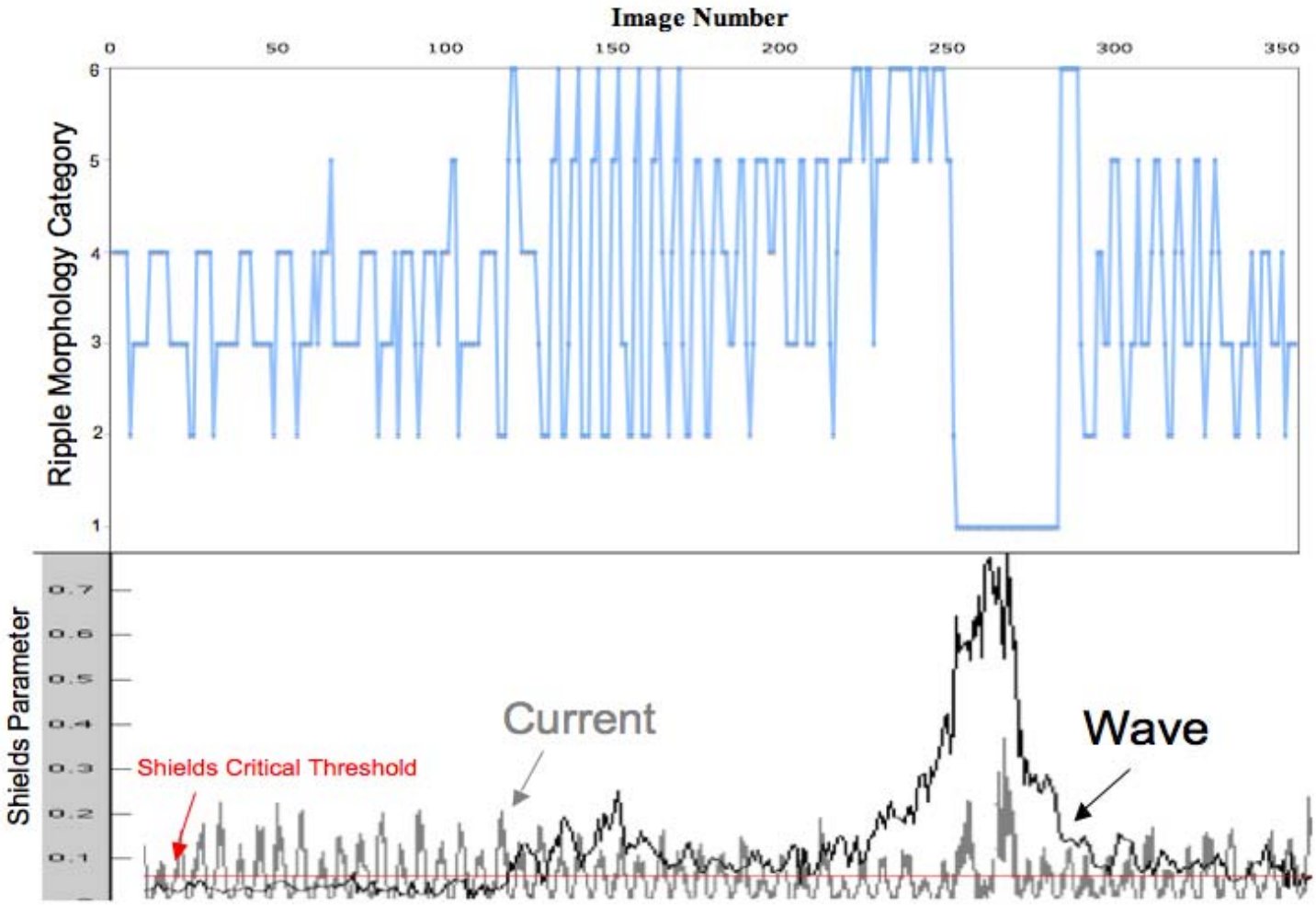

Figure 13. Evolution of bedform morphology categories over time as compared to hydrodynamic conditions. The morphological changes can be linked to the hydrodynamics. For example, while the storm event was building (image numbers 240-250), wave energy overtook current energy as the dominant hydrodynamic force and formed the most linear ripples (stage 6)

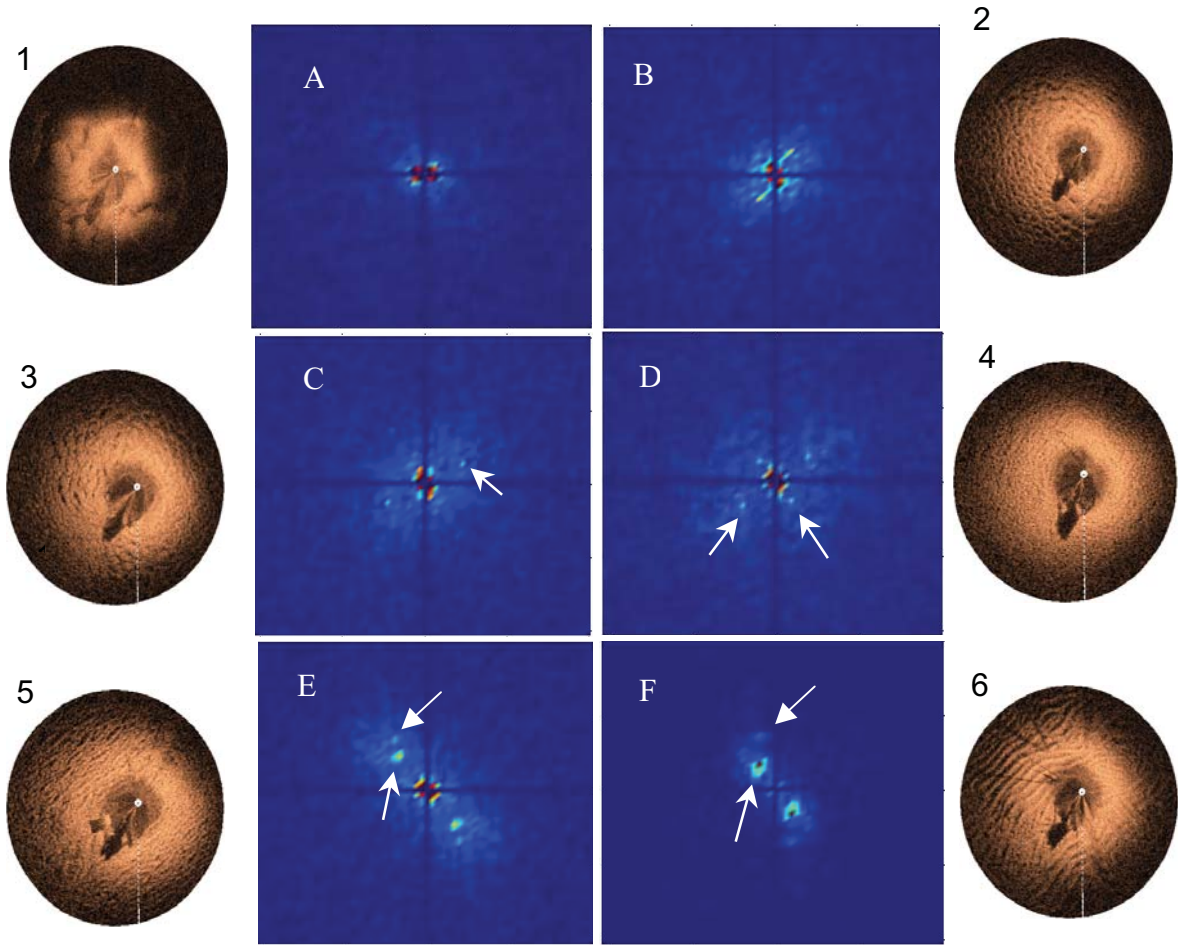

Figure 14. Representative plot of the Fourier analysis for each of the six bedform morphology categories. All beds exhibit a high peak in the center (red in A-E), which is the frequency from end to end of the square sub-region. Unique characteristics of A: morphology 1, no peaks identified by transform. B: morphology 2, peak defined, but with a large amount of noise. C: two peaks, one for the hummocky features, and one for the building linear ripple features. D: stronger two peaks resulting from hummocks and ripples. E: Peak identified for the complex ripples, but noise as the crests are not strong. F: the mostly linear complex ripples with well-defined crests have a strong peak with little noise.

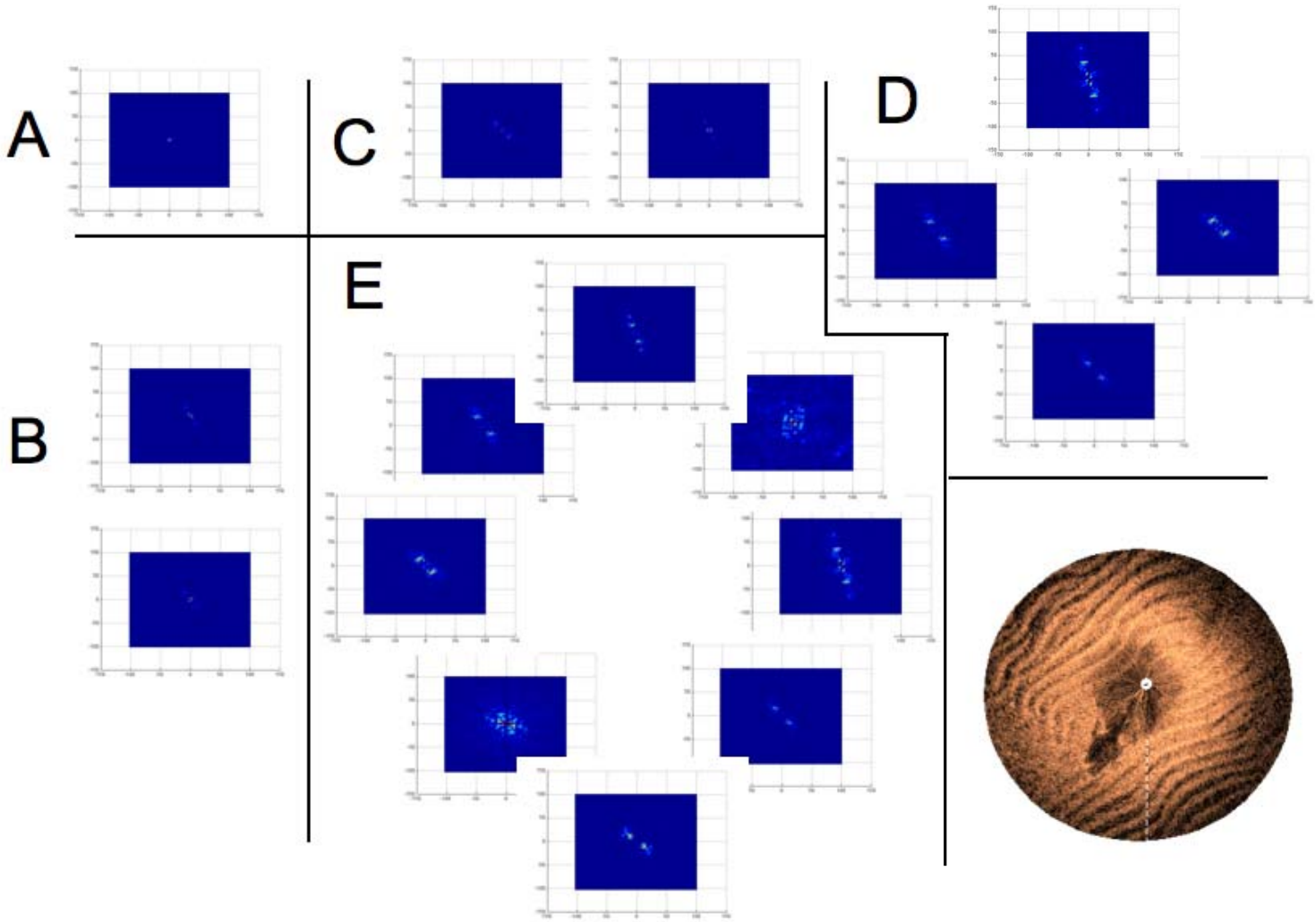

Figure 15. Differences in Fourier plots from each sub-region arrangement as calculated for the same image. A. Single sub-region. B. Two sub-regions, top and bottom. C. Two sub-regions, left and right. D. Four sub-regions. E. Eight sub regions.

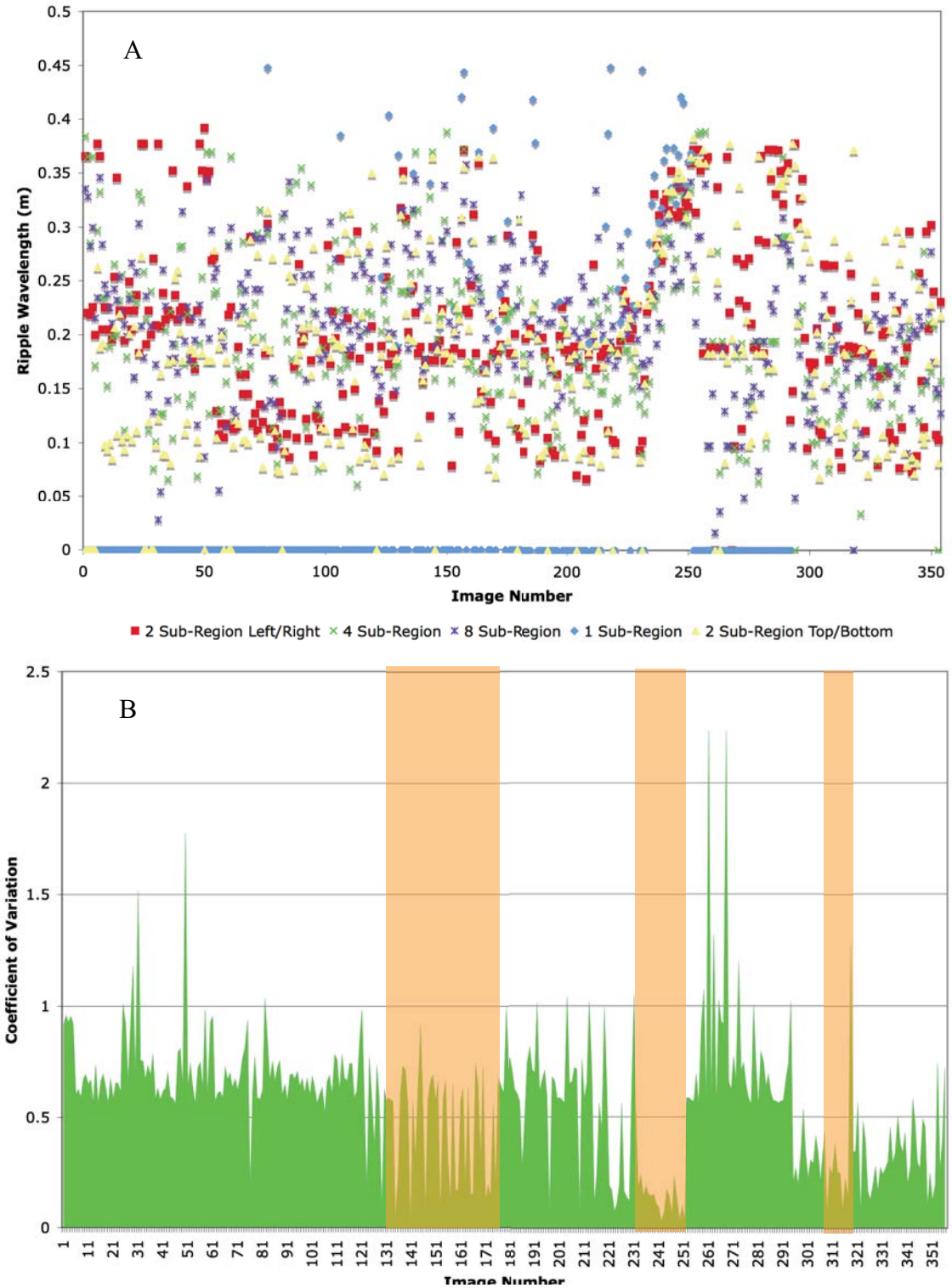

Figure 16. Variability of ripple wavelength outputs for the five different sub-region arrangements. A. Sub-region arrangements with wavelength outputs for each image. B. Coefficient of variation for the sub-region arrangements. Highlighted areas have the least variation and correspond to morphology categories 5 and 6.

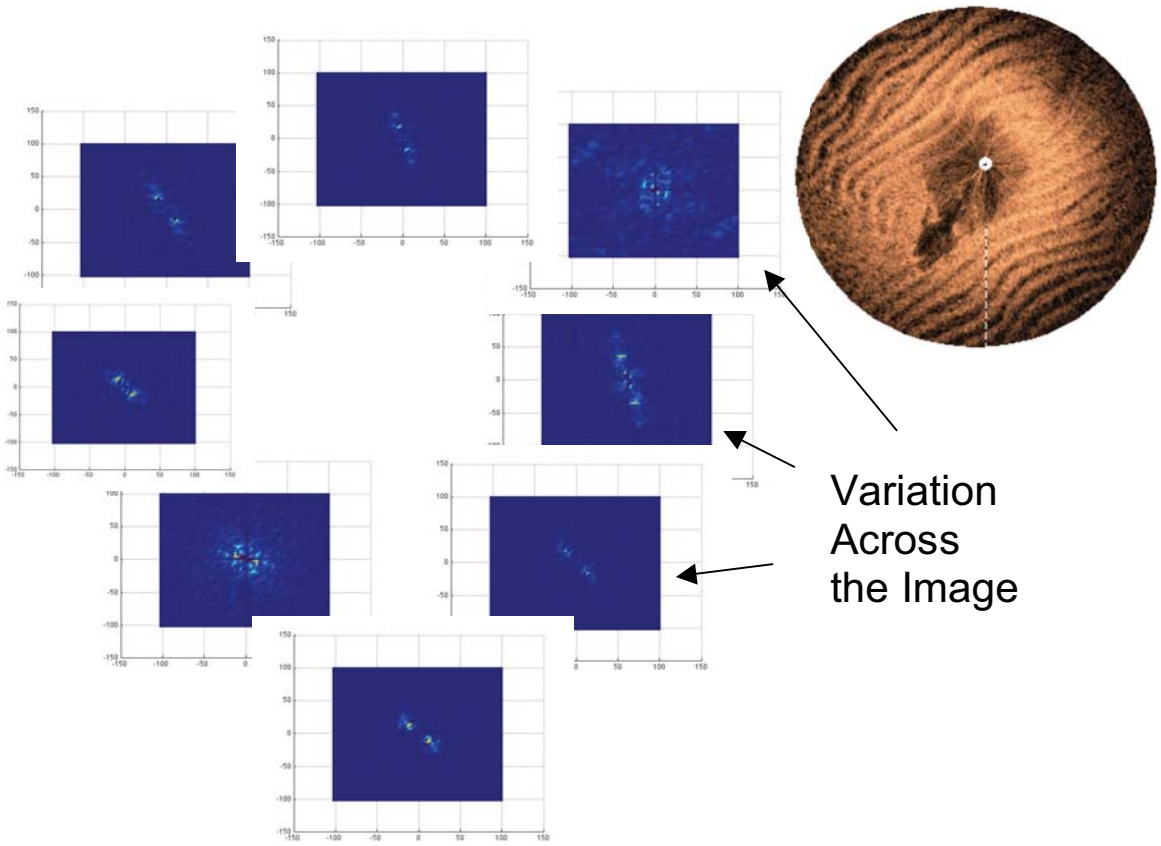

Figure 17. Variation in Fourier analysis between sub-regions of a single image. Areas of the image where ripple orientation is parallel with the direction of sonar are more variable than areas where the features are oriented orthogonal to the sonar.

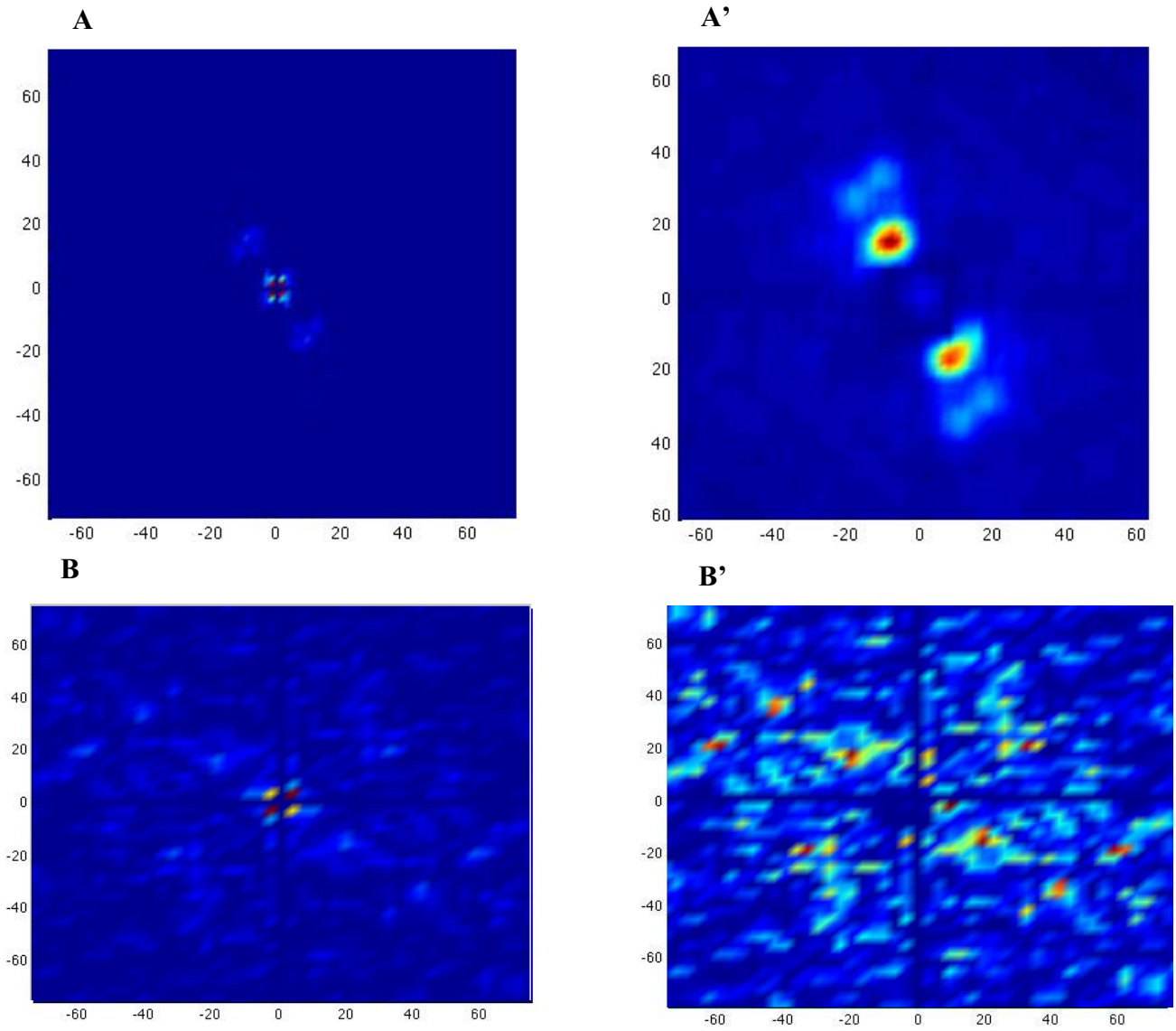

Figure 18. First filter as run on two different bedform morphologies. A' is able to leave the appropriate peaks behind that correspond to a single wavelength-and-orientation value. B' removes the central peaks but further processing is needed to identify the appropriate peak that may relate to actual ripple wavelengths and orientations.

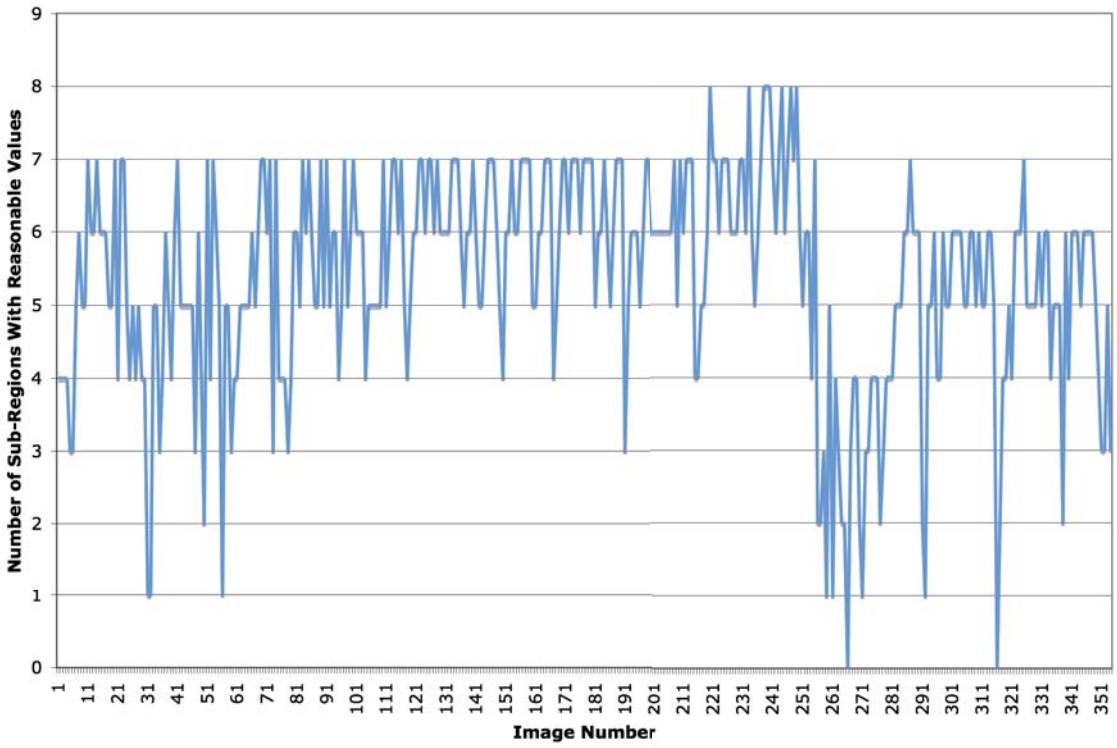

Figure 19. Number of sub-regions remaining per image after incorporating filter to remove regions with unreasonable ripple wavelength outputs. The majority of images have at least one sub-region removed.

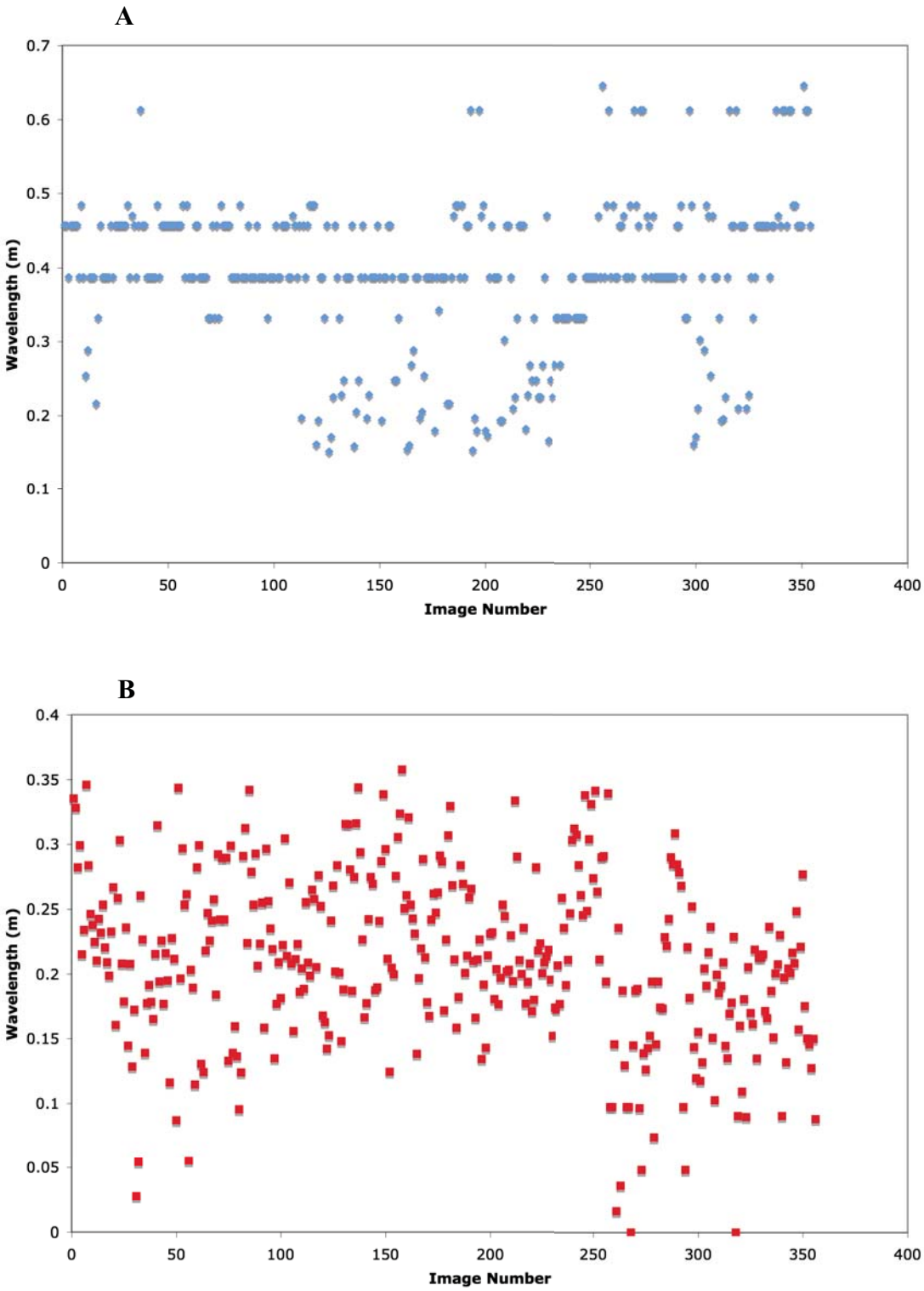

Figure 20. Difference in ripple wavelength outputs when averaged with unreliable values versus when they are removed. A: The unreliable low-frequency noise is not removed before making the wavelength-and-orientation calculations. B: The unreliable regions are removed and the linear features seen in the top plot are removed.

Appendix I

Refer to CD-ROM for time-lapse videos of ripple morphology evolution from two hydrodynamic conditions. File “Hen and Chickens Background Bimodal” contains images taken from hydrodynamic stages 1 and 2; file “Hen and Chickens Storm” contains images taken from a major storm event (hydrodynamic stage 4). Each successive image in the video is taken one hour after the previous image. The green line is a vector representing direction and strength of current energy. The blue vector represents strength of wave energy and direction of wave propagation.

Appendix II

```
%mfile to load, plot, and save Fanbeam data
dirname='/Users/Chris/Desktop/HONORS/Data To Work With/Matlab
Data/HONORS/';
dd=dir(dirname);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


image_selection= input('Enter the image number to process. e.g. 20 or
20:25');

subregion_Arrangement=menu('Choose the subregion arrangement','1
subregion','2 subregions left and right', '2 subregions top and
bottom', '4 subregions', '8 subregions', 'Manually selected sub-
region');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=image_selection%1:354;
    for ii=(iikey(i));%length(iikey)))%images;
    %464:4:length(dd)-33; %ACT


switch subregion_Arrangement
    case 1
 fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1;
[X,Y,Z]=rectify_plot_image_81ACT(Data,AA,z,sw,T);
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);
    case 2
         fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1;
[X,Y,Z]=rectify_plot_image_81ACT(Data,AA,z,sw,T);
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);
    case 3
         fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1;
[X,Y,Z]=rectify_plot_image_81ACT(Data,AA,z,sw,T);
```

```
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);
    case 4
         fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1;
[X,Y,Z]=rectify_plot_image_81ACT(Data,AA,z,sw,T);
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);
    case 5
         fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1;
[X,Y,Z]=rectify_plot_image_81ACT(Data,AA,z,sw,T);
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);

    otherwise

        fname=[dirname dd(ii).name];
[T,AA,Data]=read_image_81b(fname);
A=AA;
z=1; %height above bed in m
sw=1; %set to 1 to plot
[X,Y,Z]=sonarimage(Data,AA,z,sw,T);
% use a datetimestamp for the file name
datestr(T(1),30)
fnamef=datestr(T(1),30);
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Enter in the desired axis limits for the subregion arrangements

switch subregion_Arrangement
    case 1
        %Dimensions for 1 subregion
        xminimum=[-4],
        xmaximum=[4],
        yminimum=[-4],
        ymaximum=[4],
        %subregion dimensions combined into a matrix
        %xmin=1st column, xmaximum=2nd column, ymin=3rd column,
ymax=4th column
        subregions_dimensions=[xminimum' xmaximum' yminimum'
ymaximum'];


    case 2
```

```
        %Dimensions for 2 subregions
                %left  right
        xminimum=[-4    0],
        xmaximum=[0     4],
        yminimum=[-2   -2],
        ymaximum=[2     2],
        %subregion dimensions combined into a matrix
        %xmin=1st column, xmaximum=2nd column, ymin=3rd column,
ymax=4th column
        subregions_dimensions=[xminimum' xmaximum' yminimum'
ymaximum'];

    case 3
        %Dimensions for 2 subregions
                %top  bottom
        xminimum=[-2   -2],
        xmaximum=[2     2],
        yminimum=[0    -4],
        ymaximum=[4     0],
        %subregion dimensions combined into a matrix
        %xmin=1st column, xmaximum=2nd column, ymin=3rd column,
ymax=4th column
        subregions_dimensions=[xminimum' xmaximum' yminimum'
ymaximum'];

    case 4

        %Dimensions for 4 subregions
                %up       right          bot          left
        xminimum=[-1          1           -1          -3],
        xmaximum=[ 1          3            1          -1],
        yminimum=[ 1         -1           -3          -1],
        ymaximum=[ 3          1            1           1],
        %subregion dimensions combined into a matrix
        %xmin=1st column, xmaximum=2nd column, ymin=3rd column,
ymax=4th column
        subregions_dimensions=[xminimum' xmaximum' yminimum'
ymaximum'];

    case 5
        %Dimensions for 8 subregions overlap
            %top subregion top-right   right   bottom-right  bottom
bottom-left      left   top-left
        xminimum=[-1        .5           1       .5             -1
-2.5            -3     -2.5],
        xmaximum=[ 1       2.5           3       2.5             1
-.5            -1      -.5],
        yminimum=[ 1        .5          -1      -2.5            -3
-2.5            -1       .5],
        ymaximum=[ 3       2.5           1       -.5             1
-.5              1      2.5],
        %subregion dimensions combined into a matrix
        %xmin=1st column, xmaximum=2nd column, ymin=3rd column,
ymax=4th column
        subregions_dimensions=[xminimum' xmaximum' yminimum'
ymaximum'];
    otherwise
```

```
       %use ginput
       [xcoordinate,ycoordinate] = ginput;

       close;

       xminimum=(xcoordinate)-1;
       xmaximum=(xcoordinate)+1;
       yminimum=(ycoordinate)-1;
       ymaximum=(ycoordinate)+1;

       subregions_dimensions=[xminimum xmaximum yminimum ymaximum];

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


subregioncount=1;
cutoffcount=0;



%Run the sub-sampling as many times as there are numbers of subregions
for r=1:size(xminimum,2)

B1x=find(X>(subregions_dimensions(r,1))& X<
(subregions_dimensions(r,2)));%Find indices for X values between xmin
and xmax
B1y=find(Y>(subregions_dimensions(r,3))&Y<
(subregions_dimensions(r,4)));%Find indicies for Y values between ymin
and ymax
B1i=intersect(B1x,B1y);%Finds where those indicies overlap
B1X=(X(B1i));%Extracts the values corresponding to those indices from
all X data
B1Y=(Y(B1i));%Extracts the values corresponding to those indices from
all Y data
B1Z=(Z(B1i));%Extracts the values corresponding to those indices from
all Z data


%2D FFT Analysis for sub-region arrangement 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

switch subregion_Arrangement
    case 1
        %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/257):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/257):(subregions_dimensions(r,4)));
```

```
%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));



%Get rid of any NaN values by zero-padding
ZI_256=ZI(2:257,2:257);
ZI_256(isnan(ZI_256))=0;


ZF=ZI_256;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns



ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(256*256);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)


ZF_prefilter=ZF;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filter #1 for 256 x 256. Removes the central peak that is linked to
sub-region dimensions
g=zeros(256,256);
g(118:138,118:138)=.5;
g(120:136,120:136)=.97;
g(122:134,122:134)=1;

ZF=ZF-(ZF.*g);%applies low frequency noise filter

h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data

%2D FFT Analysis for sub-region arrangement 2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 2
        %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/129):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
```

```
(subregions_dimensions(r,4)))/129):(subregions_dimensions(r,4)));


%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));



%Cut the grid down to 128 x 128 like Vougaris
ZI_128=ZI(2:129,2:129);
ZI_128(isnan(ZI_128))=0;



ZF=ZI_128;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns



ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(128*128);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)



ZF_prefilter=ZF;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Filter #1 for 128 x 128. Removes the central peak that is linked to
sub-region dimensions
g=zeros(128,128);% creates a low frequency noise filter for matrix
g(50:79,50:79)=.5;
g(57:72,57:72)=.97;
g(61:67,61:67)=.98;
g(62:66,62:66)=.99;
g(63:65,63:65)=.99;

ZF=ZF-(ZF.*g);%applies low frequency noise filter

h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%2D FFT Analysis for sub-region arrangement 3
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    case 3
        %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/129):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/129):(subregions_dimensions(r,4)));


%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));



%Cut the grid down to 128 x 128 like Vougaris
ZI_128=ZI(2:129,2:129);
ZI_128(isnan(ZI_128))=0;



ZF=ZI_128;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns


ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(128*128);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)



ZF_prefilter=ZF;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filter #1 for 128 x 128. Removes the central peak that is linked to
sub-region dimensions
g=zeros(128,128);% creates a low frequency noise filter for matrix
g(50:79,50:79)=.5;
g(57:72,57:72)=.97;
g(61:67,61:67)=.98;
g(62:66,62:66)=.99;
g(63:65,63:65)=.99;

ZF=ZF-(ZF.*g);%applies low frequency noise filter
```

```
h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data



%2D FFT Analysis for sub-region arrangement 4
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    case 4
        %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/65):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/65):(subregions_dimensions(r,4)));


%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));



%Cut the grid down to 64 x 64
ZI_64=ZI(2:65,2:65);
ZI_64(isnan(ZI_64))=0;



ZF=ZI_64;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns



ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(64*64);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)


ZF_prefilter=ZF;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filter #1 for 64 x 64. Removes the central peak that is linked to sub-
region dimensions
g=zeros(64,64);% creates a low frequency noise filter for matrix
g(29:35,29:35)=.5;
```

```
g(30:34,30:34)=.98;
g(31:33,31:33)=.99;

ZF=ZF-(ZF.*g);%applies low frequency noise filter

h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data


%2D FFT Analysis for sub-region arrangement 5
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    case 5
        %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/65):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/65):(subregions_dimensions(r,4)));


%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));


%Cut the grid down to 64 x 64
ZI_64=ZI(2:65,2:65);
ZI_64(isnan(ZI_64))=0;


ZF=ZI_64;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns

ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(64*64);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)

ZF_prefilter=ZF;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filter #1 for 64 x 64. Removes the central peak that is linked to sub-
region dimensions
g=zeros(64,64);% creates a low frequency noise filter for matrix
g(29:35,29:35)=.5;
g(30:34,30:34)=.98;
```

```
g(31:33,31:33)=.99;

ZF=ZF-(ZF.*g);%applies low frequency noise filter


h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data


%2D FFT Analysis for manual sub-region arrangement
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    otherwise
                %Meshgrid to the size of the subregions
[XI,YI] =
meshgrid((subregions_dimensions(r,1)):(abs((subregions_dimensions(r,1))
-(subregions_dimensions(r,2)))/65):(subregions_dimensions(r,2)),...
    (subregions_dimensions(r,3)):(abs((subregions_dimensions(r,3))-
(subregions_dimensions(r,4)))/65):(subregions_dimensions(r,4)));


%Grid the indexed values from X,Y,Z with the meshgrid dimensions above
ZI=[griddata(B1X,B1Y,B1Z,XI,YI)];


%Give a variable name to each of the subregions
(subregion_1,subregion_2,subregion_3...etc)
eval(sprintf('subregion_%d = [ZI];',r));
%Saves the values from each subregion
eval(sprintf('BZI_%d = [ZI];',r));
eval(sprintf('BXI_%d = [XI];',r));
eval(sprintf('BYI_%d = [YI];',r));


%Cut the grid down to 128 x 128 like Vougaris
ZI_64=ZI(2:65,2:65);
ZI_64(isnan(ZI_64))=0;


ZF=ZI_64;
ZF=detrend(ZF');%Removes long term trend from rows
ZF=detrend(ZF');%Removes long term trend from columns


ZF=fft2(ZF);% Fourier Transform of matrix
ZF=abs(ZF);% Absolute value of matrix (makes all values real)
ZF=ZF .* conj(ZF)/(64*64);% Complex conjugate of matrix
ZF=fftshift(ZF);% Centers spectral peaks of matrix (outside corners are
flipped to center)

ZF_prefilter=ZF;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Filter #1 for 64 x 64. Removes the central peak that is linked to sub-
region dimensions
g=zeros(64,64);% creates a low frequency noise filter for matrix
g(29:35,29:35)=.5;
```

```
g(30:34,30:34)=.98;
g(31:33,31:33)=.99;

ZF=ZF-(ZF.*g);%applies low frequency noise filter

h=ones(5,5)/5;
ZF=filter2(h,ZF);%smooths data

end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%make the variables ZF_1 through ZF_8 in order to average them below
eval(sprintf('ZF_%d = [ZF];',r));%ZF{r}


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This subregion of code determines wave number domain as explained by
Krogstad (2004).

switch subregion_Arrangement
    case 1
        kx1 = mod( 1/2 + (0:(256-1))/256 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.03100775);
        ky1 = mod( 1/2 + (0:(256-1))/256 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.03100775);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);

    case 2
        kx1 = mod( 1/2 + (0:(128-1))/128 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.0307692);
        ky1 = mod( 1/2 + (0:(128-1))/128 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.0307692);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);

    case 3
        kx1 = mod( 1/2 + (0:(128-1))/128 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.0307692);
        ky1 = mod( 1/2 + (0:(128-1))/128 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.0307692);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);
```

```
    case 4
        kx1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.03030303);
        ky1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.03030303);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);


    case 5
        kx1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.03030303);
        ky1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.03030303);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);


    otherwise
        kx1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        kx = kx1*(2*pi/0.03030303);
        ky1 = mod( 1/2 + (0:(64-1))/64 , 1 ) - 1/2;
        ky = ky1*(2*pi/0.03030303);
        [Xw,Yw] = meshgrid(kx,ky);
        Xw=fftshift(Xw);
        Yw=fftshift(Yw);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%determine the wavelength and orientation of each sub-region

max(ZF);
peakmax=max(ans);

wavenumber_index=find(ZF==peakmax);
Ripple_Wavelength=(2*pi)/(sqrt((Xw(wavenumber_index(1)))^2 +
(Yw(wavenumber_index(1)))^2))
Subregion_Ripple_Wavelengths_All(i,subregioncount)=Ripple_Wavelength;

Ripple_Orientation=
atand((Xw(wavenumber_index(1)))/(Yw(wavenumber_index(1)))) + pi
Subregion_Ripple_Orientations_All(i,subregioncount)=Ripple_Orientation;


%This is a 2nd filter that will get rid of any outputs that have the
wavelength being greater than .6 meters because .6 meters is larger
than any ripples imaged
if (Ripple_Wavelength < 0.6);
    cutoffcount=1+cutoffcount;%makes a counter to place each value into
its own new column starting with column 1

Subregion_Ripple_Wavelengths_filtered(i,cutoffcount)=Ripple_Wavelength;
```

```
Subregion_Ripple_Orientations_filtered(i,cutoffcount)=Ripple_Orientatio
n;
end

subregioncount=1+subregioncount;


end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Averages the wavelength and orientation outputs from each sub-region

if (cutoffcount > 0)

Wavelength_Average=mean(Subregion_Ripple_Wavelengths_filtered(i,:))
Wavelength_Averages(i,1)=Wavelength_Average;
Orientation_Average=mean(Subregion_Ripple_Orientations_filtered(i,:))
Orientation_Averages(i,1)=Orientation_Average;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Averages the Fourier transform of each sub-region together


switch subregion_Arrangement
    case 1
        ZFA=((ZF_1));
    case 2
        ZFA=((ZF_1+ZF_2)/2);
    case 3
        ZFA=((ZF_1+ZF_2)/2);
    case 4
        ZFA=((ZF_1+ZF_2+ZF_3+ZF_4)/4);
    case 5
       %Average all 8 transformed subregions together.
        ZFA=((ZF_1+ZF_2+ZF_3+ZF_4+ZF_5+ZF_6+ZF_7+ZF_8)/8);
    otherwise
        ZFA=((ZF_1));

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculates the wavelength and orientation of the averaged Fourier
transform
```

```
max(ZFA);
peakmax=max(ans)

wavenumber_index=find(ZFA==peakmax);
Ripple_Wavelength_fourier_averaged=(2*pi)/(sqrt((Xw(wavenumber_index(1)
))^2 + (Yw(wavenumber_index(1)))^2))

Ripple_Orientation_fourier_averaged=
atand((Xw(wavenumber_index(1)))/(Yw(wavenumber_index(1)))) + pi
%Ripple_Orientations(i,1)=Ripple_Orientation;

Averaged_FFT_Spectra(i,1)= Ripple_Wavelength_fourier_averaged;
Averaged_FFT_Spectra(i,2)= Ripple_Orientation_fourier_averaged;
Averaged_FFT_Spectra(i,3)= peakmax;


    end
end
```