FP6-513663

FLUID
FLUid Image Description and analysis

Instrument: Strep

Priority: IST FET OPEN

Deliverable 2.4

Demonstrator on Multigrid Multiscale Motion Estimation

Patrick Heas, Etienne Memin, Nicolas Papadakis

Due date of deliverable: 31/11/06
Actual submission date: 15/01/07

Start date of project: 01/12/04 Duration: 3 years

Organisation name of lead contractor for this deliverable: INRIA

Revision [1]

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

This document aims at describing codes compiled under a Linux environment which are used for estimating fluid motion with the methods presented in [1][2]. The first part briefly describes the files (compiled codes and data examples) which are provided. The second part constitutes the reference manual which explains how the programs can be used.

1- List of files

```
<root>.

|-- Code
| |-- CodeDenseMotion
| | |-- DenseMotion_2DFluid_V1
| | |-- DenseMotion_2DFluid_V2
| | |-- DenseMotion_Meteo_V1
| | |-- DenseMotion_Meteo_V2
| | |-- param_2DFluid_V1
| | |-- param_2DFluid_V2
| | |-- param_Meteo_V1
| | ‘--param_Meteo_V2
| ‘-- CreateImages
|   |-- makeImages.sh
|   |-- RawIR2pgmImages
|   |-- Ir2transImages
|   |-- Pres2transImages
|   |-- inputRawIR
|   |-- inputFiles_IR
|   ‘-- inputFiles_P
|
‘-- DataExamples
  |-- Classification
  | |-- NAtl_CLA_scene_analysis.43
  | |-- NAtl_CLA_scene_analysis.44
  | |-- NAtl_CLA_scene_analysis.45
  | ‘-- raw2pgmClass.sh
  |-- CorrelationVectorField
  | |-- NAtl_cst_c10.8.43_44
  | |-- NAtl_cst_c10.8.44_45
  | |-- PIV00_01
  | ‘-- PIV01_02
  |-- PIV
  | |-- im00.pgm
  | |-- im01.pgm
  | ‘-- im02.pgm
  |-- IR
  | |-- NAtl_c10.8.43
  | |-- NAtl_c10.8.44
  | ‘-- NAtl_c10.8.45
  ‘-- Transmit
    |-- NAtl_CLA_cloud_top_press.43
    |-- NAtl_CLA_cloud_top_press.44
    ‘-- NAtl_CLA_cloud_top_press.45
```

2- User Manual

In this section we describe how the program developed can be used. Using multiscale approaches and a multigrid optimization technique, a user can estimate fluid flow motion through four different compiled codes dedicated either to two-dimensional experimental fluid flows or to atmospheric flows.

2-1 Motion estimators for two-dimensional fluid flows

Estimators `DenseMotion_2DFluid_V1` and `DenseMotion_2DFluid_V2` are dense estimators implementing the Optical Flow Constraint (OFC), which is a valid assumption in the case of two-dimensional flows.

2-1-1 Spatial regularization

Estimator `DenseMotion_2DFluid_V1` uses a div-curl spatial regularizer for closure within a multiresolution approach. For a detailed description please refer to [1] [4].

Image file (in *.pgm* format) paths and program parameters are grouped in a text file named `param_2DFluid_V1` which has necessary to be on the same directory. It comprises on each of its line a character string describing the input. Default inputs are saved in `param_2DFluid_V1` so that the program can be directly launched without fixing any options and using predefined images (provided by *Cemagref*). The inputs can be changed preserving the following file structure:

```
Number_of_images_in_the_sequence_(>=2): X
Image_t1_path(pgm_format): X
Image_t2_path(pgm_format): X
.
.
Regularisation_Coefficient: X
Coarsest_Resolution_for_multiresolution: X
Finest_Resolution_for_multiresolution: X
Coarsest_Grid_for_multigrid: X
Finest_Grid_for_multigrid: X
Robust_threshold_data_model: X
Regularization_coefficient_for_Div_and_Curl_smoothing: X
Robust_threshold_on_DivCurl: X
```

2-1-2 Spatio-temporal regularization with correlation-based constraints

Estimator `DenseMotion_2DFluid_V2` uses the same div-curl regularizer en sus of a curl temporal regularizer enforcing the time consistency of the flow vorticity according to Navier-Stokes equations. Moreover, in this last version, the user can choose to incorporate a sparse correlation-based vector field as an external constraint in a variational scheme which yields to a two-stage joint estimation based on correlation and optical flow. For a detailed description please refer to [2] [5].

Image file (in *.pgm* format) paths and program parameters are grouped in a text file named `param_2DFluid_V2` which has necessary to be on the same directory. It comprises on each of its line a character string describing the input. Default inputs are saved in `param_2DFluid_V2` so that the program can be directly launched without fixing any options and using predefined images (provided by *Cemagref*) and correlation vector fields (provided by *LaVision*). The inputs can be changed preserving the following file structure:

```
Number_of_images_in_the_sequence_(>=2): X
Image_t1_path(pgm_format): X
Image_t2_path(pgm_format): X
.
.
Spatial_Regularisation_Coefficient: X
Coarsest_Resolution_for_multiresolution: X
Finest_Resolution_for_multiresolution: X
Coarsest_Grid_for_multigrid: X
Finest_Grid_for_multigrid: X
Robust_threshold_data_model: X
Regularization_coefficient_for_Div_and_Curl_smoothing: X
Robust_threshold_on_DivCurl: X
Constrain_velocity_estimates_with_correlation_yes(1)_no(0): X
Weighting_factor_for_correlation_constraints: X
Temporal_regularization_Coefficient: X
Standard_deviation_of_gaussian_extending_the_correlation_influence: X
Correlation_vector_field_t1_t2_path: X
.
.
```

The correlation vector fields are text files where each vector is saved on a line with the following structure (format produced by *LaVision* software) :

```
x     y     u     v
```

where `x` and `y` denotes the pixel coordinates, `u` and `v` the horizontal and vertical displacements.

2-2 Motion estimators for atmospheric flows

Estimators `DenseMotion_Meteo_V1` and `DenseMotion_Meteo_V2` are dense estimators implementing the Integrated Continuity Equation (ICE) model. They are presented just after describing the code to derive infrared and transmittance images in *pgm* format from EUMETSAT binary files.

2-2-1 Creation of infrared and layer-related transmittance images in *pgm* format

An example of infrared and layer-related transmittance images computation can easily be created by simply launching the script `makeImages.sh`.

For other examples, this script can be modified by fixing the options of the sub-routine `RawIR2pgmImages` for creating infrared images in *pgm* format and of the sub-routine `Pres2transImages` for creating layer-related transmittance images in *pgm*

format from top of clouds images and cloud classification maps. In the case top of cloud pressure images are not available, an alternative is to perform a coarse approximation of transmittance images using the sub-routine `Ir2transImages` which is based on infrared images and classification maps.
The option list of each of the following sub-routines can be displayed when they are launched with the option `-h`

Infrared images: `RawIR2pgmImages`
The infrared image files in 2 bytes *pgm* format can be derived from MSG 2 bytes binary images by the routine named `RawIR2pgmImages` which takes as inputs a text file `inputRawIR` composed on each line of a path related to an input binary image (option `–path`), the image dimensions (options `–dy` and `-dx`), the number of images in the sequences (option `-n`) and inputs for an optional cropping procedure (options `–ofx` , `–ofy` and `-Dx`). For example, one can launch the routine using the following command:

```
./RawIR2pgmImages -h -path inputRawIR -dy 512 -dx 512 -n 3 -ofy 40 -
ofx 80 –Dx 1024
```

Layer transmittance images from top of cloud pressure: `Pres2transImages`
Layer related transmittance image files in 2 bytes *pgm* format can be derived from 2 bytes binary images of top of cloud pressure by the routine named `Pres2transImages` which takes as inputs a text file `inputFiles_P` composed on each line of a path related to an input binary image and to its corresponding binary cloud classification map (option `–path`), the image dimensions (options `–dy` and `-dx`), the number of images in the sequences (option `-n`), the number of layer which should be equal to 3 or less (options `–nbLayer`), the maximum and minimum class index defining each layer (options `–cMinL`, `-cMaxL`, `–cMinM`, `-cMaxM`, `–cMinH`, `-cMaxH`), and inputs for an optional cropping procedure (options `–ofx` , `–ofy` and `-Dx`). For example, one can launch the routine using the following command :

```
./Pres2transImages -h -path inputFiles_P -dy 512 -dx 512 -n 3 –cMinL
101 -cMaxL 103 –cMinM 104 -cMaxM 106 –cMinH 107 -cMaxH 109 -ofy 40 -
ofx 80 –Dx 1024
```

Layer transmittance images from infrared images: `Ir2transImages`
The infrared image files in 2 bytes *pgm* format can also be used with cloud classification maps in *pgm* format for the coarse approximation of layer–related transmittance images. The routine used for computation is named `Ir2transImages`. It is based on the coarse retrieval of top of cloud pressure according to the procedure described in [3] with no cloud transparency corrections. The sub-routine takes as inputs a text file `inputFiles_IR` composed on each line of a path related to an input *pgm* image and to its corresponding *pgm* cloud classification map (option `–path`), the image dimensions (options `–dy` and `-dx`), the number of images in the sequences (option `-n`), the number of layer which should be equal to 3 or less (options `–nbLayer`), the maximum and minimum class index defining each layer (options `–cMinL`, `-cMaxL`, `–cMinM`, `-cMaxM`, `–cMinH`, `-cMaxH`), the black body calibration coefficient (options `–alpha`), the space count (options `–Ro`) and inputs for an optional cropping procedure (options `–ofx` , `–ofy`). Black body calibration coefficient and space count are values routinely provided by the EUMETSAT consortium. Binary

Classification can be converted to *pgm* format launching (after modification for other examples) the script `raw2pgmClass.sh`. For example, one can launch the routine using the following commands :

```
cd ../../DataExamples/Classification
raw2pgmClass.sh
cd ../../Code/CreateImages/
./ Ir2transImages -h -path inputFiles_IR -dy 512 -dx 512 -n 3 –cMinL
101 -cMaxL 103 –cMinM 104 -cMaxM 106 –cMinH 107 -cMaxH 109 -ofy 40 -
ofx 80–alpha 0.205034 –Ro -10.4568
```

2-2-2 Infra-red data model with a spatial regularizer

Estimator `DenseMotion_Meteo_V1` uses MSG infra-red images for the calculation of a global wind field related to all observable clouds. Motion is estimated for div-curl spatial regularizer for closure within a multiresolution approach. For a detailed description please refer to [1] [4].

Image file (in *.pgm* format) paths and program parameters are grouped in a text file named `param_Meteo_V1` which has necessary to be on the same directory. It comprises on each of its line a character string describing the input. Default inputs are saved in `param_Meteo_V1` so that the program can be directly launched without fixing any options and using predefined images (provided by *LMD*). The inputs can be changed preserving the following file structure:

```
Number_of_images_in_the_sequence_(>=2): X
Image_t1_path(pgm_format): X
Image_t2_path(pgm_format): X
.
.
Coarsest_Resolution_for_multiresolution: X
Finest_Resolution_for_multiresolution: X
Coarsest_Grid_for_multigrid: X
Finest_Grid_for_multigrid: X
Robust_threshold_data_model: X
Robust_threshold_data_model: X.
Regularization_coefficient_for_Div_and_Curl_smoothing: X
```

2-2-3 Transmittance data model, shallow water spatio-temporal regularizer and correlation-based constraints

Estimator `DenseMotion_Meteo_V2` uses a set of transmittance images related to a stacks of layers. Contrarily to previously, the ICE model is in this case physically sound for such images. The transmittance images are computed from cloud classification maps and top of clouds pressure maps or, if the latter are unavailable, the transmittance images may be derived from cloud classification maps and infra-red images. The same div-curl regularizer is used en sus of a shallow water temporal regularizer enforcing in a first stage the time consistency of the flow vorticity and divergence according to simplified shallow waters equations. Moreover, in this last version, the user can choose to incorporate correlation-based vector fields as external constraints in a variational scheme which yields in a first stage to the joint estimation of large scale motion based on correlation and optical flow. In a second stage both temporal and correlation-based constraints are removed and the estimation is refined at small scales based only on the ICE data model. The aim of this two-stage scheme is to propose an alternative approach overcoming multiresolution technique limitations and enforcing the time consistency of the estimates. For a detailed description please refer to [2].

Image file (in *.pgm* format) paths and program parameters are grouped in a text file named `param_Meteo_V2` which has necessary to be on the same directory. It comprises on each of its line a character string describing the input. Default inputs are saved in `param_Meteo_V2` so that the program can be directly launched without fixing any options and using predefined images, classifications and correlation-based vector fields (provided by *LMD*). The inputs can be changed preserving the following file structure:

```
Number_of_images_in_the_sequence_(>=2): X
layer_t1_path: X
image_t2_path: X
layer_t2_path: X
image_t3_path: X
.
.
Regularisation_Coefficient: X
Coarsest_Resolution: X
Finest_Resolution: X
Coarsest_Grid: X
Finest_Grid: X
Robust_threshold_data_model: X
Regularization_coefficient_for_Div_and_Curl_smoothing: X
Constrain_velocity_estimates_with_correlation_yes(1)_no(0): X
Weighting_factor_for_correlation_constraints: X
Image_of_latitudes: X
Temporal_regularization_Coefficient: X
Standard_deviation_of_gaussian_extending_the_correlation_influence: X
Offset_in_x_for_reading_correlation_files: X
Offset_in_y_for_reading_correlation_files: X
image_t1_path: X
correlVectors_t1_t2: X
correlVectors_t2_t3: X
.
.
```

The correlation vector fields and the latitude image are text files which have the format produced by the *LMD software.* For a description of the file structure, please refer to [3].

References

[1] T. Corpetti, E. Memin, and P. Perez. **Dense estimation of fluid flows**. IEEE Trans. Pattern Anal. Machine Intell., 24(3):365–380, 2002.

[2] P. Heas, E. Memin, N. Papadakis, A. Szantai, **Layered estimation of atmospheric mesoscale dynamics from satellite imagery**, IEEE Transactions on Geosciences and Remote Sensing (in press.).

[3] A. Szantai and F. Desalmand. **Basic information on msg images**. Fluid project deliverable 1.1, 2005.

[4] T. Corpetti, D. Heitz, G. Arroyo, E. Mémin, A. Santa-Cruz. **Fluid experimental flow estimation based on an optical-flow scheme**. Experiments in fluids, 40(1):80-97, 2006.

[5] D. Heitz, P.Heas,V. Navaza, J. Carlier, E. Mémin, **Spatio-temporal correlation-variational approach for robust optical flow estimation**, symposium on Particle Image Velocimetry (PIV), 2007.