Two-component horizontal motion vectors from scanning eye-safe aerosol lidar

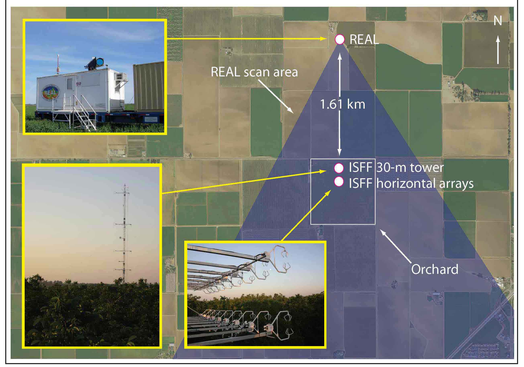
Shane D. Mayor, Department of Physics, California State University Chico, Chico, CA; and P. Dérian, P. Héas, and E. Mémin, INRIA Rennes, France

Objective

Derive the two-component vector motion field from pairs of aerosol backscatter lidar images and compare results with observations from a co-located sonic anemometer. Two methods will be tested: correlation and dense estimation.

Experiment

The REAL was located 1.61 km north of a micrometeorological tower. PPI scans intersected the tower at 12 m AGL. A sonic anemometer at that level provides a reference wind measurement for comparison with the lidar derived velocities.



Conclusions

Correlation Method Strengths

- Each vector is not part of a global solution and therefore each vector is not influenced by information beyond the block size.

Correlation Method Weaknesses

- Lacks constraints on solutions and allows unrealistic vectors.

Dense Estimation Method Strengths

- Produces a vector at every pixel in the image.
- Constraints reduce the number of unrealistic vectors.

Dense Estimation Method Weaknesses

- For the work shown here, it appears the results may be oversmoothed.

Turbulence, or the lack of it, has a large effect on the comparisons.

Acknowledgements

This work was made possible by Grant 0924407 from the U.S. National Science Foundation's Physical and Dynamic Meteorology Program. Jen Lowe assisted with the preparation of graphics.

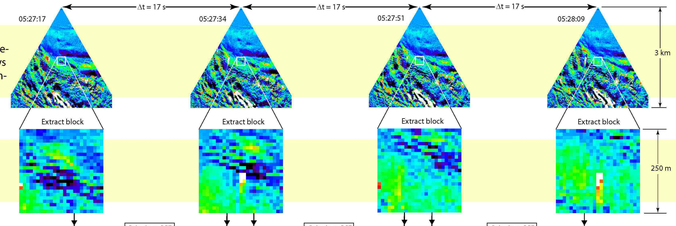
Algorithm: Correlation

SCANS

In this example, a series of PPI scans are used, each resulting from about 150 laser pulses. The radial arrays have been high- and low-pass median filtered and interpolated to a Cartesian grid with 10 m resolution.

BLOCKS

A block can be extracted from anywhere in a gridded scan. In this this example, they are taken from a location centered on the tower.



CROSS-CORRELATION FUNCTIONS (CCFs)

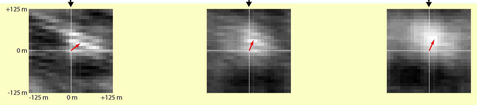
The cross-correlation functions are computed by:

F1 = FFT(block1,-1)

F2 = FFT(block2,-1)

CCF = FFT(F1 * Conj (F2), 1)

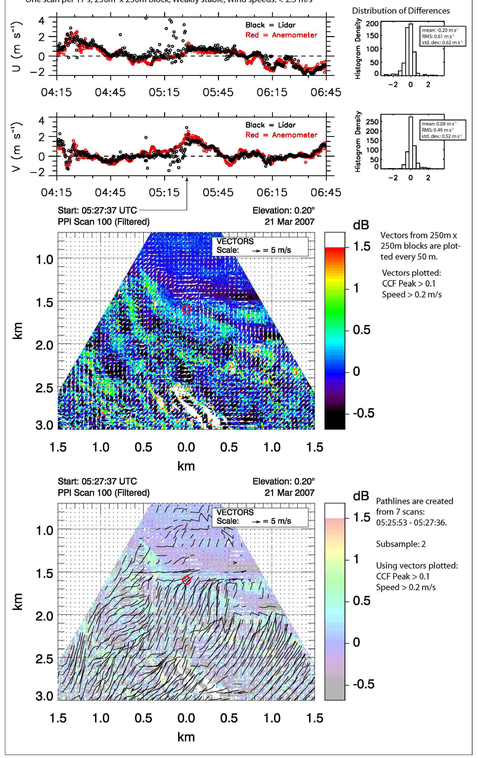
The 9x9 point area surrounding the peak of the CCF is fit with a bicubic spline to provide a finer resolution estimate of the peak location.



The wind vector (red arrow) is calculated by noting the displacement of the peak of the CCF from the origin and dividing by the time elapsed between frames.

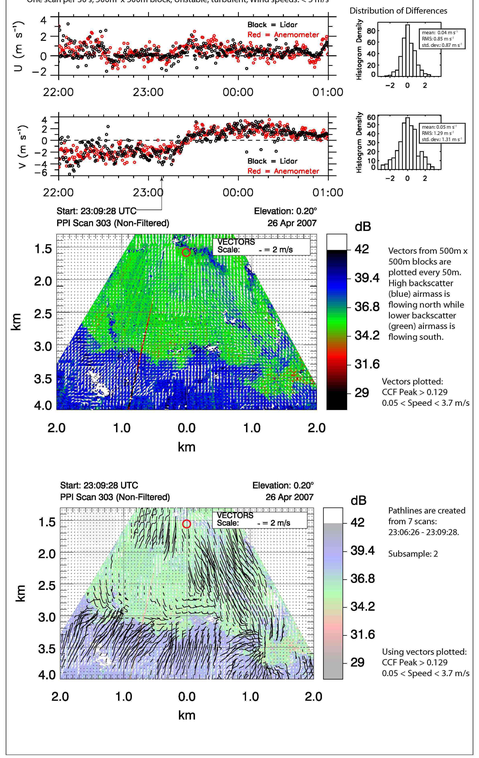
Case 1: Weakly stable, nearly-quiescent evening

21 March 2007
One scan per 17 s; 250m x 250m block; Weakly stable; Wind speeds: < 2.5 m/s



Case 2: Turbulent afternoon with frontal passage

26 April 2007
One scan per 30 s; 500m x 500m block; Unstable, turbulent; Wind speeds: < 5 m/s



Algorithm: Dense Estimation

The complete algorithm is summarized below:

- $\hat{\boldsymbol{d}} \equiv 0$
- From coarsest to finest resolution do[a]
  - $\Delta\boldsymbol{d} \equiv 0$, $\xi \equiv 0$, $\zeta \equiv 0$, $s^d \equiv \tau_1$, $s^\xi \equiv \tau_2$, $s^\zeta \equiv \tau_2$
  - Until convergence do[b,b,c]

$$\Delta\boldsymbol{d} \leftarrow GS\left(\sum_s s_s^d g^2(\Delta\boldsymbol{d})_s + \alpha|\mathrm{div}\hat{\boldsymbol{d}}(s) + \mathrm{div}\Delta\boldsymbol{d}(s) - \xi_s|^2 + \alpha|\mathrm{curl}\hat{\boldsymbol{d}}(s) + \mathrm{curl}\Delta\boldsymbol{d}(s) - \zeta_s|^2\right)$$

$$s_s^d \leftarrow \tau_1 \exp -\tau_1 g^2(\Delta\boldsymbol{d})_s, \forall s \in S$$

$$\xi \leftarrow GS\left(\sum_s |\mathrm{div}\hat{\boldsymbol{d}}(s) + \mathrm{div}\Delta\boldsymbol{d}(s) - \xi_s|^2 + \lambda\sum_{(s,r)} s_{s,r}^\xi |\xi_s - \xi_r|^2\right)$$

$$s_{s,r}^\xi \leftarrow \tau_2 \exp -\tau_2|\xi_s - \xi_r|^2, \forall (s,r) \in C$$

$$\zeta \leftarrow GS\left(\sum_s |\mathrm{curl}\hat{\boldsymbol{d}}(s) + \mathrm{curl}\Delta\boldsymbol{d}(s) - \zeta_s|^2 + \lambda\sum_{(s,r)} s_{s,r}^\zeta |\zeta_s - \zeta_r|^2\right)$$

$$s_{s,r}^\zeta \leftarrow \tau_2 \exp -\tau_2|\zeta_s - \zeta_r|^2, \forall (s,r) \in C$$

- If not finest resolution, pass $\hat{\boldsymbol{d}} \leftarrow \hat{\boldsymbol{d}} + \Delta\boldsymbol{d}$ to next resolution.
- If finest resolution, $\boldsymbol{d} \leftarrow \hat{\boldsymbol{d}} + \Delta\boldsymbol{d}$.

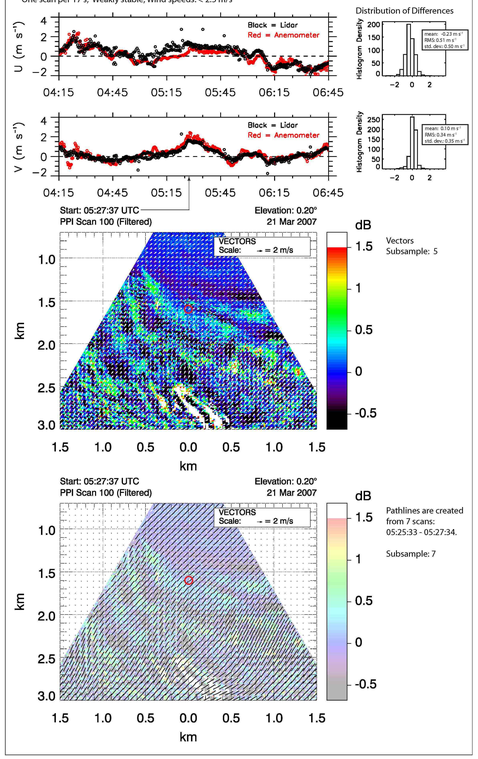
[a] Convergence is reached when the relative change in the L_2-norm of $\Delta\boldsymbol{d}$ is below 3%.
[b] $g(\Delta\boldsymbol{d})_s$ denotes $\exp(\mathrm{div}\hat{\boldsymbol{d}}(s))((E(s)\nabla\mathrm{div}\hat{\boldsymbol{d}}(s) + \nabla E(s)) \cdot \Delta\boldsymbol{d}(s) + \hat{E}(s)) - E(s)$.
[c] $GS(\cdot)^*$ denotes one Gauss-Seidel iteration on the quadratic form under consideration.

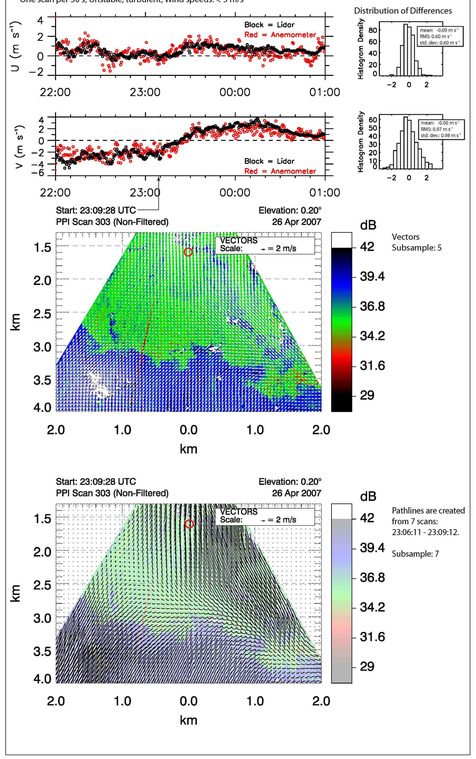
Dense estimation is a technique to derive the apparent motion of features in a series of images based on spatial and temporal derivatives of pixel brightness. It is based on a pair of images through the solution of a partial differential equation. However, this equation does not enable a unique solution since there is only one scalar constraint (the PDF) and two unknowns (the velocity components). To close the problem, additional constraints are applied. In the work here, we apply the algorithm of Corpetti et al. (2002).

Corpetti, T., Mémin, É., and Pérez, P. 2002: Dense estimation of fluid flows. IEEE Transactions on pattern analysis and machine intelligence, pp. 365–380.

Case 1: Weakly stable, nearly-quiescent evening

21 March 2007
One scan per 17 s; Weakly stable; Wind speeds: < 2.5 m/s



Case 2: Turbulent afternoon with frontal passage

26 April 2007
One scan per 30 s; Unstable, turbulent; Wind speeds: < 5 m/s