J12.4 An Object-Oriented Approach to the Verification of Quantitative Precipitation Forecasts: Part I - Methodology

Randy Bullock, (NCAR, Boulder, 80307) Barbara G. Brown
Chris A. Davis, Michael Chapman, Kevin W. Manning and Rebecca Morss

Author Contact Information: Randy Bullock, NCAR, PO Box 3000, Boulder CO 80307
Phone: (303) 497-8444; e-mail: `bullock@ucar.edu`

1. Introduction

The main goal of our project is to develop and test advanced diagnostic methods for the evaluation of quantitative precipitation forecasts. To that end, an "object-oriented" strategy has been adopted. Rather than score the precipitation field on a gridpoint-by-gridpoint basis, both the forecast and observed fields are resolved into *objects* (or *regions of interest*), and forecast and observed objects are then compared. The procedure is outlined in this section and a step-by-step example is presented in Sec 2.

The setup is as follows. We have a two-dimensional grid of points, say

$$G = \{(x, y) | 1 \le x \le N, 1 \le y \le N\}$$

and a scalar field f defined on G. The first step in resolving objects in this field is performing a convolution filtering of the original data field. Defining another scalar field on G (the *convolving function* or *filter*) by

$$\phi(x, y) = \begin{cases} H & \text{if } \sqrt{x^2 + y^2} \le R \\ 0 & \text{otherwise,} \end{cases}$$

then the convolved field g is given by

$$g(x, y) = \sum_{u=-R}^{R} \sum_{v=-R}^{R} \phi(u, v) f(x - u, y - v).$$

The parameters H and R define the filter. Usually R is taken to be anywhere from a few to a dozen or so grid-units and H is chosen so that the volume under the graph of ϕ is equal to 1, *i.e.,*

$$\pi R^2 H = 1.$$

A "circular" filter such as defined here is by no means the only possible choice. We have experimented with some other filters and have found the end results do not change appreciably with these more complicated filters. Thus we go with the simple filter above.

The second step is to threshold the convolved field. Why don't we just threshold the original field? Remember our goal is to resolve objects in the original field. As is shown in the example in Sec. 2, thresholding data that has high spatial frequncy components (*e.g.* precipitation data) will not produce representative objects. The convolution process gets around that difficulty. It can be thought of as a simple low-pass spatial filter that attenuates high spatial frequencies in the data. Any points in the grid where the value of the convolved field is greater than the threshold are set to one—any points below the threshold are set to zero. This gives a zero/one field that can be used as a mask.

Finally, the mask field is applied to the original data field, preserving data that are inside detected objects and zeroing out the rest of the field.

The rest of this paper is devoted to illustrating and extending these ideas. Section 2 gives an example using real-world precipitation forecast data. Sections 3 and 4 discuss what to do with objects once you've got them. Section 5 gives a summary and indication of future work. References are given in Section 6.

2. Example

All this should be made clearer by an example. We will look at the various stages of this process as applied to a real data field. CONUS WRF precipitation data from the summer of 2001 will be used.

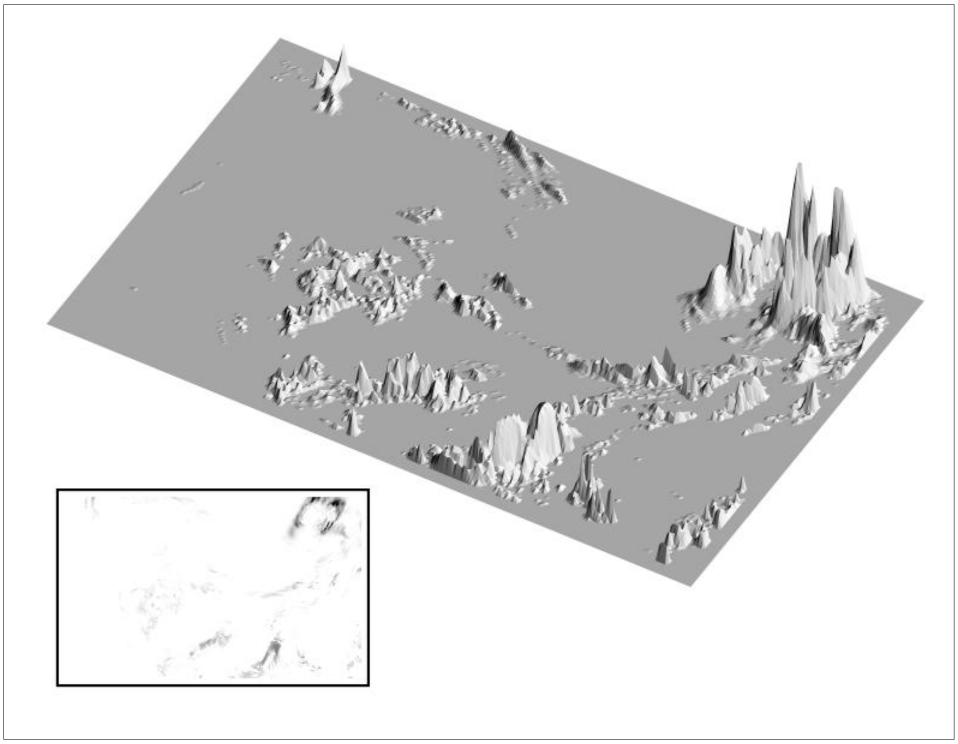

Fig. 1 Raw Precipitation Field

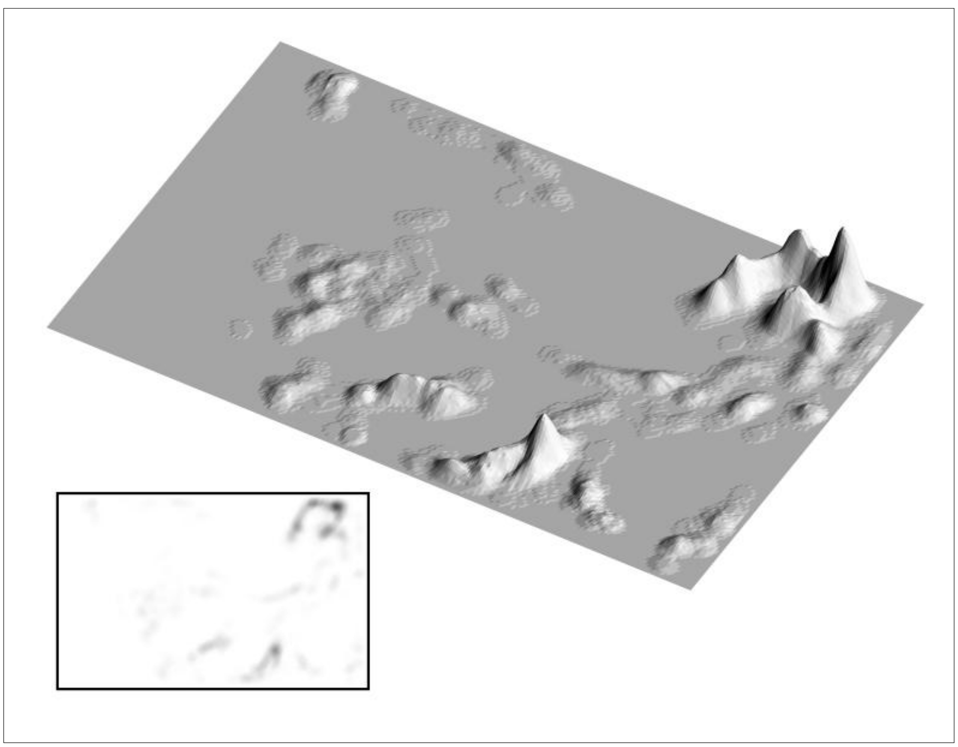

Fig. 2 Convolved Precipitation Field

In **Fig. 1** we show a raw precipitation field, in both a 3-D and a 2-D rendering. The various steps involved in the process are easier to see in a 3-D view. (*Note:* for clarity, state outlines have been left out.) Note that the original data are so uneven that a simple thresholding of the raw data at any intermediate value would not result in a small number of representative objects. It has been our experience that most 2-D plots of scalar fields on a grid fail to give the viewer an appreciation of just how "raw" the raw data often is.

The next step is convolving. The particular parameters used for the convolving function aren't important for this example. As explained earlier, this step is a particular form of low-pass filtering. High spatial frequencies are greatly attenuated while low spatial frequencies survive relatively intact. **Fig. 2** shows the result.

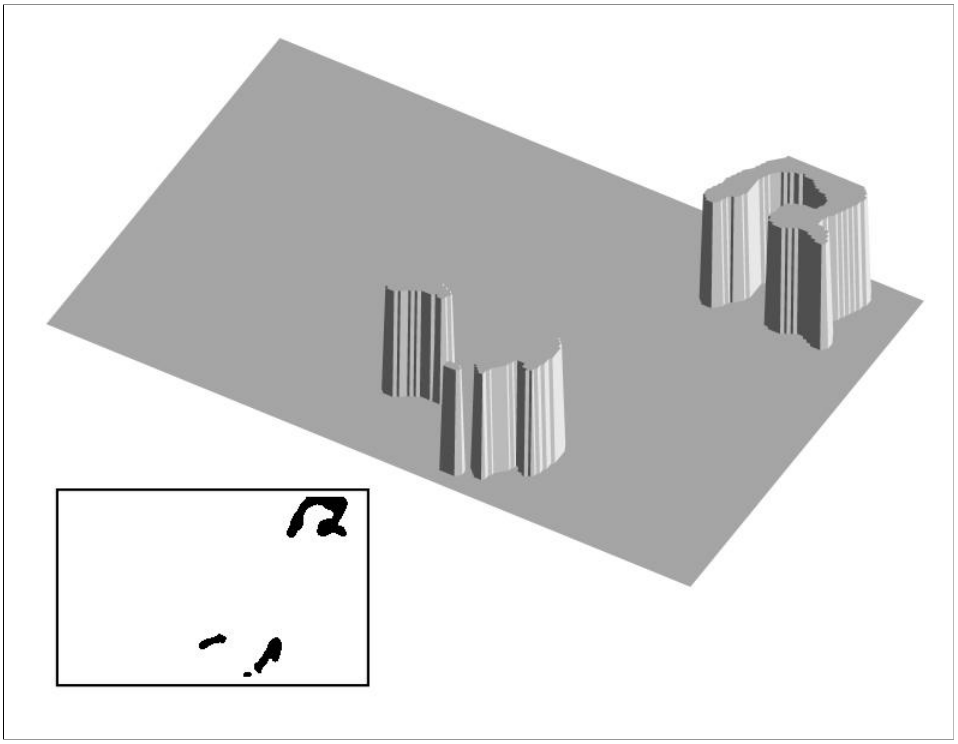

Fig. 3 Zero/One Mask Field

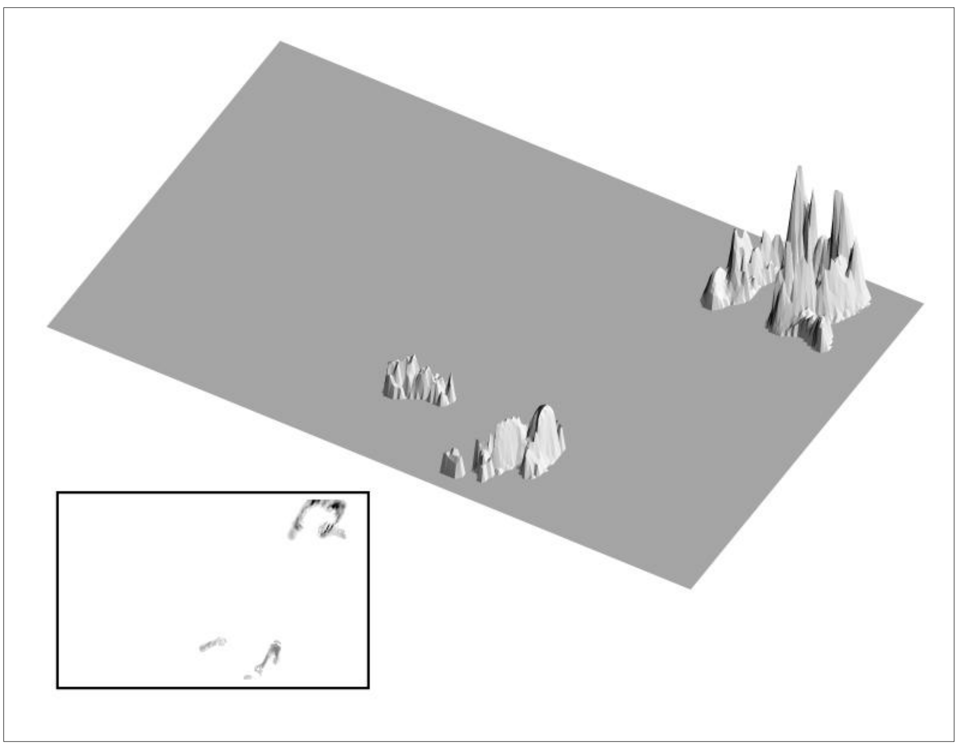

Fig. 4 Original Field Resolved into Objects

If this convolved field is thresholded at some intermediate value, a relatively small number of larger connected shapes will result—rather than the large number of smaller disconnected shapes which would have resulted if the raw field were thresholded. If we now perform this thresholding, and replace all grid values that were above the threshold by one, and all values below the threshold by zero, a zero/one field is obtained that can be used as a mask for the original data. The result is shown in **Fig. 3**. It's important to understand that the convolved or "filtered" data are used for detecting object boundaries and for *nothing else*. Once the objects have been resolved, we then apply this mask field gridpoint-by-gridpoint to the original data—in other words, at every gridpoint where the mask field is zero, the original data field is set to zero. At every grid point where the mask field is one, the original data field is left unchanged. We then have our original data field resolved into objects and the rest of the data field outside these regions of interest has been zeroed out. See **Fig. 4**.

At this point a number of things can be done. Distributions of data values inside objects can be calculated, and various object attributes such as centroid, axis angle, moments and curvature can be calculated. In addition, the objects can be replaced by fitted shapes with similar attributes chosen to represent the objects in simplified or schematic form.

3. Shape Fitting

It is often desirable to replace an object by something simpler—something incorporating the important characteristics of the object but (for many purposes) easier to work with. This is analagous to summarizing a probability distribution by giving its mean and first few central moments. We distill pertinent object characteristics into several descriptive attributes, and consider simpler geometric shapes that have the same or similar attributes. Several such schematic representations are used.

First, the *centroid*. This is the geometric center of the object, and as such it provides an easy way to assign simple locations to extended shapes. See the upper left illustration in **Fig. 5**.

Let the object be represented by

$$f(x, y) = \begin{cases} 1 & \text{if } (x, y) \text{ is inside the object} \\ 0 & \text{otherwise,} \end{cases}$$

where x and y take integer values in some finite range, say from 1 to N, then the coordinates $(\overline{x}, \overline{y})$ of the centroid are given by

$$\overline{x} = \frac{1}{A} \sum_{x=1}^{N} \sum_{y=1}^{N} x f(x, y)$$

and

$$\overline{y} = \frac{1}{A} \sum_{x=1}^{N} \sum_{y=1}^{N} y f(x, y)$$

where A is the area (expressed as number of grid squares) of the object. This should strongly remind the reader of the mean of a two-dimensional probability distribution. We shall be needing higher moments of this "distribution" in a minute.

Next, the *axis*. See the upper middle illustration in **Fig. 5** for an example. Note that this is not an axis of symmetry—indeed few objects will even *have* an axis of symmetry. Neither is it the usual least-squares approach to fitting a line to a set of data points., It is not, however, an entirely new approach—see the book by Ritter and Wilson in the references.

The axis angle is obtained from a two-step procedure: First, the coordinate system is translated so that the centroid is at the origin. If the original coordinate system is (x_g, y_g) and the translated coordinate system is (x, y), then we have $x = x_g - \overline{x}$ and $y = y_g - \overline{y}$. Second, the (x, y) coordinate system is rotated about its origin (giving new coordinates (u, v), say) so that the second moment of u about the new origin is maximized. The u-axis thus obtained is our shape axis. Note that this axis neccesarily goes through the object centroid.

Bandaids. A bandaid shape is a rectangle with semi-circles on a pair of opposite sides. See the upper-right and lower-left illustrations in **Fig. 5** for examples. While the centroid and axis are simply object attributes, here we are really fitting a shape to an object. How is this fitting to be done?

First we require the object and bandaid to have the same centroid and axis. It would be natural to next require that the bandaid have the same second moments as the object, but this turns out to be too many conditions to satisfy. Instead, we regard the ratio of second moments along the axis to second moments orthogonal to the axis as a kind of "aspect ratio" for the object, and require the bandaid to have this same aspect ratio.

This determines the position, orientation, and aspect ratio for the bandaid. What about its overall size? Two criteria can be used. One is to require the bandaid to enclose the same area (*i.e.* number of grid squares). The other is to require the smallest size that entirely encloses the object. Both of these approaches can be seen in **Fig. 5**.

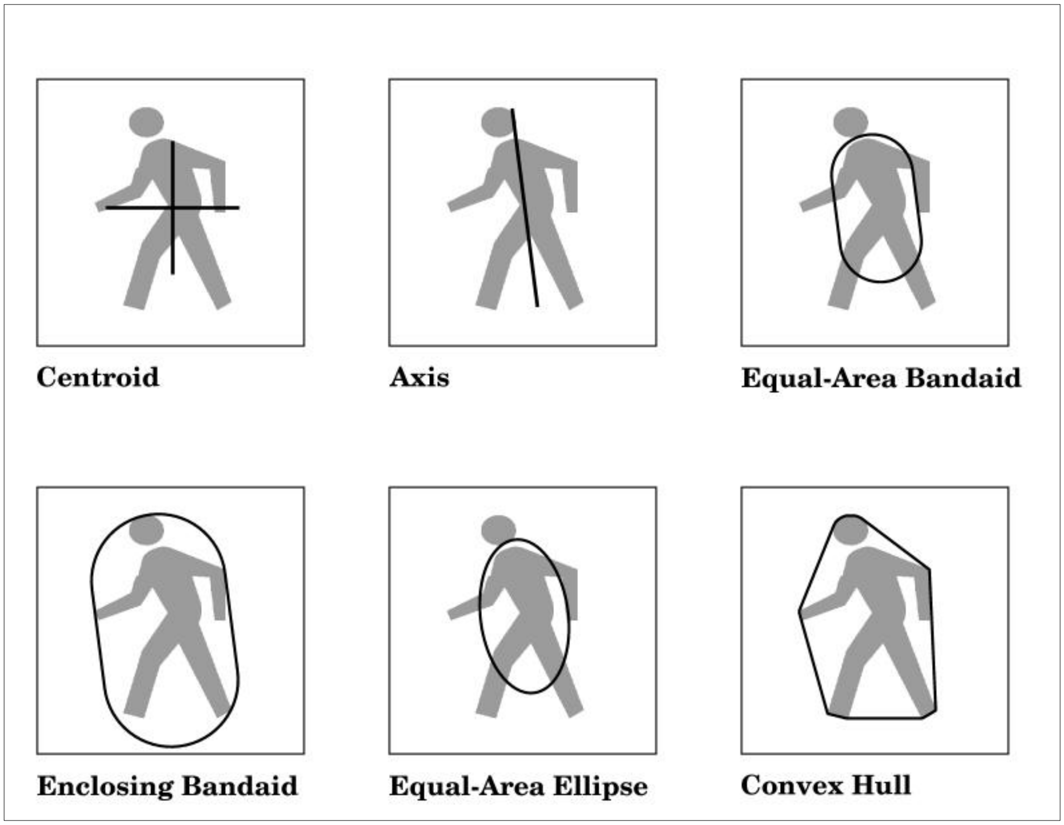

Fig. 5

Bandaids aren't the only simple geometrical shapes that can be fitted to an object. We can also use *ellipses*. Here the same criteria for fitting are used. We require that the fitted ellipse have the same centroid, axis and aspect ratio as the given object. And again, one can either have an equal-area ellipse or an enclosing ellipse (though only one of these is shown in **Fig. 5**).

Convex hull. A convex set is one that has the property that whenever two points are in the set, the straight line segment joining the two points lies entirely in the set. The convex hull of a set is the smallest convex set that contains the given set.

The convex hull uses none of the previously mentioned shape attributes. Also, it is often the case that quite a few points are required in the closed polyline that gives the hull, so the convex hull is not really a very economical representation of an object. However, it does have a sort of intuitive appeal. Again, see **Fig. 5** for an illustration.

4. Matching Shapes

Rules for matching forecast and observed objects can incorporate several criteria. First and simplest is the vector difference between their centroids. Treating this separation as a vector instead of a simple scalar distance is useful because the vector incorporates information on direction as well. Knowing that forecasts tend to be incorrectly placed in a certian direction (*e.g.* northeast) can be very useful to modelers and algorithm developers.

Also, axis information can be used. Forecast objects that have different axis orientations than observed objects are another category of forecast error.

In general, the greater the number of forecast object attributes that match (or nearly so) observed object attributes, the better the forecast can be said to be. We thus report the quality of a forecast by giving summaries of object attribute differences.

Object matching can be done as well on only a single field, rather than between a forecast and observed field. Two separate objects that have similar attributes, *e.g.* small separation and similar axes, can be considered to be parts of the same feature. None of the object attributes discussed in Sec. 3 require that the given object be in one connected piece. Notice that the figure of the walking man in our oft-referred-to **Fig. 5** is in two pieces. Centroid, axis, and other attributes can then be recalculated for this composite object, if desired.

5. Conclusions & Further Work

This paper has reported our attempts to develop forecast verification methods that move beyond traditional gridpoint-by-gridpoint techniques. Objects can be detected in data fields by a simple convolution-filter based approach. Simple geometric shapes can be fitted to objects to represent them in a simplified or schematic fashion. Objects can be matched or merged according to closeness of attributes. Forecast quality can be expressed by giving summary statistics of object attribute differences.

Future work will expand and extend these methods. Object matching, for example, can be done in the time domain as well as spatially, giving a method for tracking individual objects over their lifetime. Time tracking of objects is made difficult however, by the fact that objects can split into pieces or merge.

Improved methods of object matching will be pursued, hopefully leading to more sophisticated rule sets, perhaps including object histories. Object attributes that are meaningful for forecast developers and other users will be identified.

6. Acknowledgments

This research was supported by the U. S. Weather Research Program (USWRP). NCAR is sponsored by the National Science Foundation.

7. References

Articles & Papers

Brown, B. G., Mahoney, J. L., Davis, C. A., Bullock, R., *and* Mueller, C.
Improved Approaches for Measuring the Quality of Convective Weather Forecasts
Preprints, 16th Conference on Probability and Statistics in the Atmospheric Sciences, Orlando 2002
American Meteorological Society (Boston), 20-25

Chapman, M., Bullock, R. G., Brown, B. G., Davis, C. A., *and* Manning, K. W.
An Object-oriented Approach to the Verification of Quantitative Precipitation Forecasts: Part II - Examples
This Conference

Books

De Berg, M., van Kreveld, M., Overmars, M., *and* Schwarzkopf, O.
Computational Geometry, 2nd. Ed.
Springer, 2000

Ritter, G. X., *and* Wilson, J. N.
Computer Vision Algorithms in Image Algebra, 2nd. Ed.
CRC Press, 2001