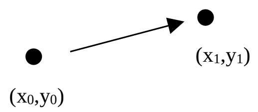
Features of transformation options

Plane transformations.

A ***plane transformation*** is a formula that maps (moves) every point in a two-dimensional plane to another:



The most widely used plane transformations are ***polynomial transformations*** because they are relatively simple to calculate and provide a good approximation even to more complex transformations. They are always defined by a pair of equations that only involve sums of powers of x and y, for example:

$$x_1 = 5x_0^2 + 3x_0y_0 + 2y_0^2 + 5x_0 + 1$$
$$y_1 = 3x_0^2 + 6x_0y_0 + 5y_0^2 + 7y_0 + 6$$

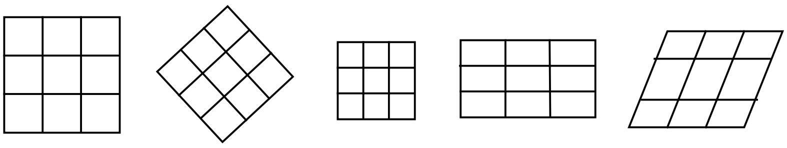
You obtain the transformed coordinates of any point simply by plugging the original coordinates into the right-hand side of these equations.

Affine Transformations

A special subset of polynomial transformations are ***affine transformations*** (sometimes loosely called *linear* transformations) that only involve x and y and constant terms (that is, no higher powers of x and y) which means all equations of the form:

$$x_1 = Ax_0 + By_0 + C$$
$$y_1 = Dx_0 + Ey_0 + F$$

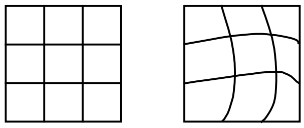
where A, B, C, D, E, F (the ***transformation parameters***) are just numbers, possibly zero. Affine transformations include translation (offset), rotation, scaling and shearing operations and all combinations of these, for example:



Notice that while lengths and angles can change, parallel lines remain parallel: this is always true of affine transformations. They are used in GIS to convert map coordinates into display co-ordinates for output to computer screens and plotters, and sometimes to correct minor distortions in scanned or digitised documents.

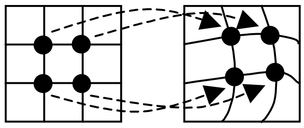
Higher-order polynomial transformations

Higher-order (*non-linear*) polynomial transformations – that is, those with equations involving higher powers of x and y, can introduce much more complex local distortions, for example:



The amount of distortion possible increases as the order of the equations increases, but the transformations are always smooth – they never introduce *tears*. Transformations that behave like this are often called ***rubber-sheet*** transformations and are used extensively in GIS and remote sensing, for example to correct camera-lens distortions in aerial imagery or to approximate the projection transformations from ellipsoidal co-ordinates to map co-ordinates.

Very often the exact parameters for a polynomial transformation are not known, but a set of ***control points*** is available for which both the untransformed coordinates and transformed co-ordinates are known. Provided there are enough of them, you can work backwards and produce a set of parameters for an approximate transformation that fits the control points well enough, and can be reliably applied to all other points as well. This set of parameters is usually termed a ***solution*** for that particular set of control points and a particular definition of *well enough*.



The more points, you have the higher the order of transformation you can generate, which allows you to model increasingly complex distortions. *In general, however,* you should *keep the order of transformation as low as possible* (second-order polynomials are often good enough) and accept some discrepancy between the actual and calculated co-ordinates of the control points. This avoids introducing artificial distortions into the data. The differences between the actual and calculated coordinates of the control points are called ***residuals***. If you have more control points than are required to calculate a set of transformation parameters, you can use the redundancy to pick parameters that produce a *best fit* to all of the data. Many systems use a mathematical technique called ***least squares*** to work out the best fit.

However the parameters are derived, you should *always check the residuals are acceptably small*. If not, you need either to revisit the control points, or consider whether a single polynomial transformation is actually appropriate for your data. If, for example, you are trying to model some sort of discontinuity (some kind of *tear*) you may need to apply different transformations on either side of the discontinuity – a *piecewise linear* transformation is a set of affine transformations joined together in this way. Extreme cases have been solved, when a large number of control points were available, by *triangulating* the control points and applying a different polynomial transformation in each triangle.