Об одном методе решения обратной RPC-задачи

Лигун А.А., Шумейко А.А., Сотник С.Л.
Днепродзержинский Государственный Технический Университет, Украина

Для решения задачи перевода физических координат объектов в координаты пикселов на спутниковых и аэрофотоснимках, часто используется так называемая RPC(RFM)[1] модель. Она заключается в следующем. Нормализованные (в пределах от -1 до 1) широта, долгота и высота подставляются в функции следующего вида:

$$px_n = \frac{f1(X_n, Y_n, Z_n)}{f2(X_n, Y_n, Z_n)}, \quad py_n = \frac{f3(X_n, Y_n, Z_n)}{f4(X_n, Y_n, Z_n)} \qquad (1)$$

где px, py – нормализованные пиксельные координаты, а X, Y, Z – нормализованные координаты объекта. Функция f имеет следующий вид:

$$\begin{aligned} f(X,Y,Z) = {} & c_1 + c_2 Y + c_3 X + c_4 Z + c_5 XY + c_6 YZ + c_7 XZ + c_8 Y^2 + c_9 X^2 + \\ & + c_{10} Z^2 + c_{11} XYZ + c_{12} Y^3 + c_{13} X^2 Y + c_{14} Z^2 Y + c_{15} Y^2 X + c_{16} X^3 + \\ & + c_{17} XZ^2 + c_{18} Y^2 Z + c_{19} X^2 Z + c_{20} Z^3 \end{aligned} \qquad (2)$$

На практике, у знаменателя коэффициент c_1 фиксируют, делая равным единице, во избежание множественности решений и деления на ноль в случаях, когда значение нормализованных координат близко к нулю.

RPC-коэффициенты приходят клиентскому приложению вместе с изображением интересующей области, и являются частью стандартов для многих форматов передачи космических и аэрофотоснимков – NITF, GeoTIFF. В то же время, при работе с изображениями, часто возникает обратная задача – найти реальные координаты объекта, находящегося в определенном месте изображения. Наиболее распространенными подходами для решения инверсной задачи, являются итеративный и прямой методы.

Итеративный заключается в том, что находятся некоторые начальные значения для решения, после чего каким-либо градиентным методом (например, Ньютона), производится уточнение решения до тех пор, пока не будет достигнута необходимая точность. Основным минусом данной методики являются значительные вычислительные затраты при вычислении каждой точки.

Прямой метод заключается в том, что находится аналитическое решение уравнений (1) относительно переменных X, Y, для константных px, py, Z. Такое решение является весьма сложным и неустойчивым в вычислительном отношении.

---

[1] RPC – Rational Positioning Capability, RFM – Rational Function Model. В контексте данной статьи эти термины могут считаться синонимами.

Мы предлагаем, поскольку прямое решение и так находится с определенной степенью точности, обратную задачу решать в том же виде, что и прямую. А именно, при фиксированном Z найти следующие закономерности:

$$X_i = \frac{R1(px_i, py_i)}{R2(px_i, py_i)},\ Y_i = \frac{R3(px_i, py_i)}{R4(px_i, py_i)}, \tag{3}$$

где

$$\begin{aligned} R_i(px, py) &= d_0^i + d_1^i px + d_2^i py + d_3^i px^2 + d_4^i px\, py + d_5^i py^2 + d_6^i px^3 + \\ &+ d_7^i px^2\, py + d_8^i px\, py^2 + d_9^i py^3 \end{aligned}, \tag{4}$$

коэффициенты которых определяются из условия:

$$\sum_{i=1}^{N}\left[\left(X_i - \frac{R1(px_i, py_i)}{R2(px_i, py_i)}\right)^2 + \left(Y_i - \frac{R3(px_i, py_i)}{R4(px_i, py_i)}\right)^2\right] \to \min, \tag{5}$$

Так как данная задача нелинейная, то прямое применение традиционных алгоритмов МНК невозможно. Для её решения предлагается использовать итерационную модификацию МНК.

Была выполнена программная реализация предлагаемого подхода, с использованием языка C# и платформы .NET Framework. Для тестирования были взяты несколько файлов в формате NITF, содержащих сегмент коэффициентов RPC. В качестве исходных точек, на которых выполнялся поиск коэффициентов (4), брались точки, расположенные на равномерной прямоугольной сетке внутри изображения. Количество точек бралось от 100 и выше.

При проверке в каждом пикселе точности найденного решения, было определено, что ошибка определения координат на исследуемых изображениях лежала в пределах 1/10 пиксела, что вполне достаточно для практического применения. В случае, если точность решения на каких-либо снимках не будет достаточной, достаточно разбить его на несколько областей, для каждой из которых найти свой набор коэффициентов d (4).