

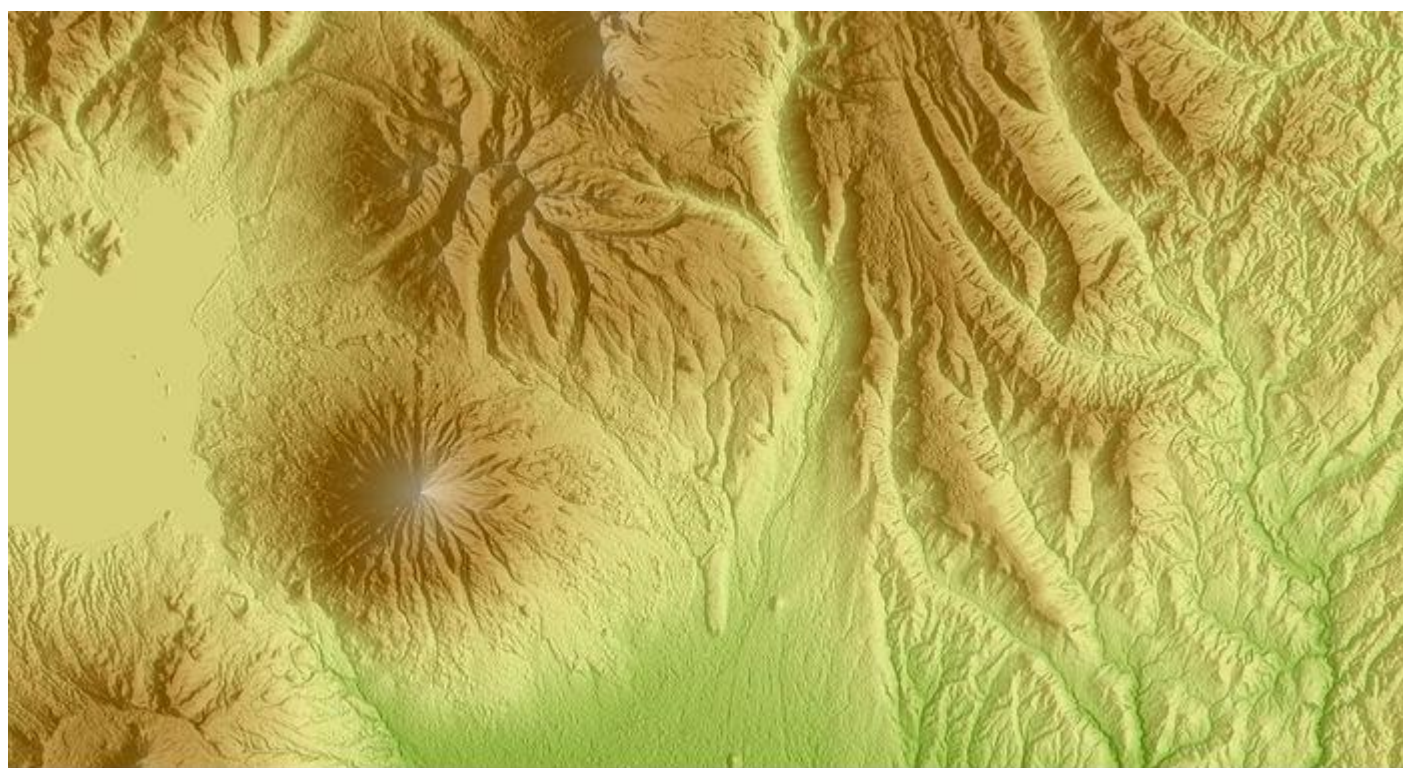
# Комбинирование светотеневой отмывки и цветовой карты рельефа с помощью GDAL

[Обсудить в форуме](#) Комментариев — 9

Эта страница опубликована в основном списке статей сайта по адресу <http://gis-lab.info/qa/gdal-hillshade-colormap.html>

Создание слоя сочетающего теневую отмывку и цветовую ЦМР в нескольких вариантах

Мы все очень любим красивый рельеф. Эта статья рассказывает про то как получить подобный результат в виде *единого* комбинированного слоя с помощью GDAL и вдаётся в некоторые нюансы применения прозрачности.



Результат комбинирования цветowego рельефа и теневой отмывки (Кроноцкий заповедник)

Обычно эффект комбинирования в ГИС (таких как QGIS) достигается использованием двух слоёв. Берется сама ЦМР (цифровая модель рельефа) в виде растра, где высоты сгруппированы в пояса и каждому задан свой цвет и берется теневая отмывка рельефа. Слой цветowego рельефа располагается поверх теневого и цветовому рельефу задается определенный процент прозрачности (в примере выше - 30%). В результате получается картинка показанная выше.

Все хорошо и красиво, но есть одна проблема. Этот вариант работает пока вы находитесь в локальной ГИС, умеющей работать со слоями. Что если вам нужен комбинированный рельеф единым слоем? Для использования в веб, графических редакторах или просто для удобства управления.

Разберем как получается изображение выше с помощью GDAL. GDAL - это швейцарский нож специалиста в области ГИС, набор кросс-платформенных консольных утилит позволяющих выполнять большее количество различных операций ([подробнее с примерами](#)).

## Содержание

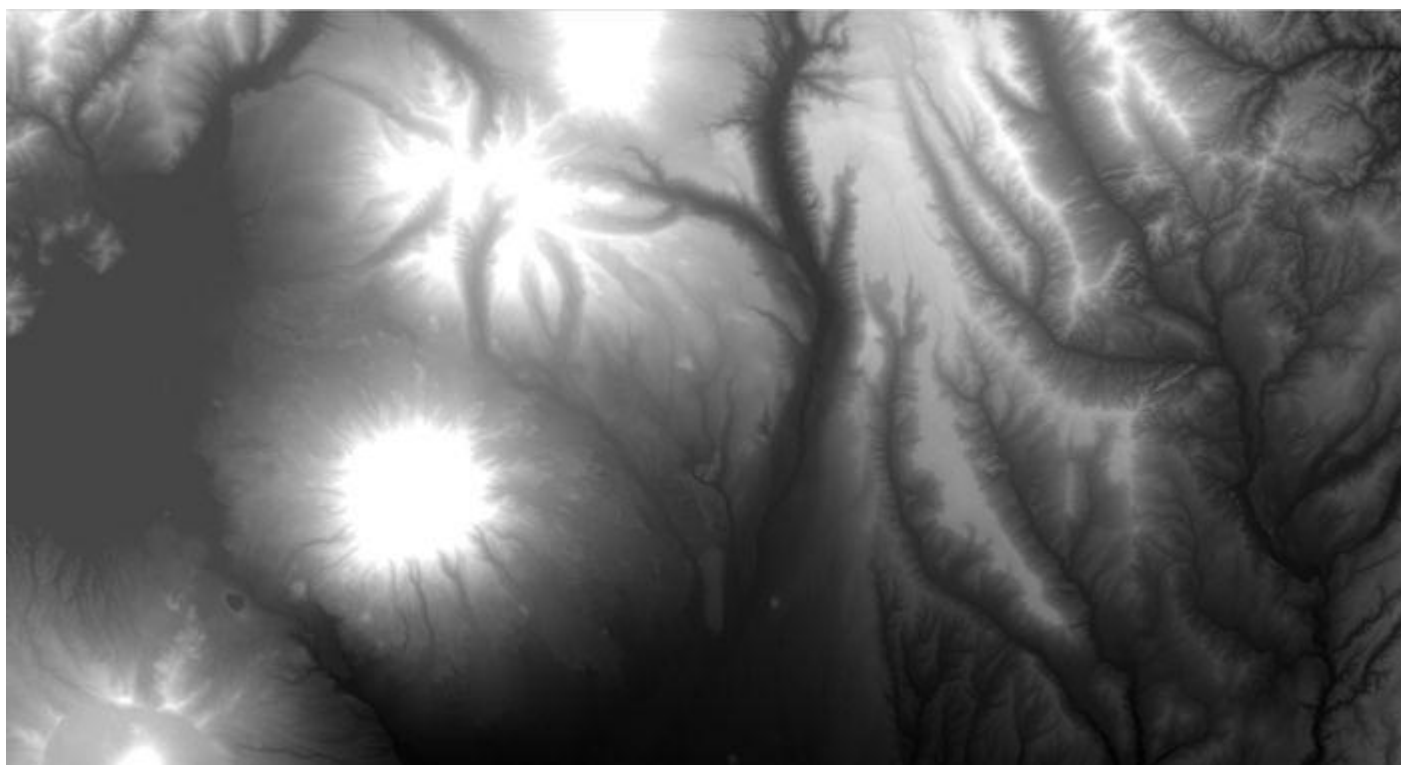
- [1 Исходные данные](#)

- [2 Теневая отмывка](#)
- [3 Цветовой рельеф](#)
- [4 Комбинирование](#)
  - [4.1 Метод 1: "в HSV и обратно"](#)
  - [4.2 Метод 2: "Прозрачность как в ГИС"](#)
  - [4.3 Метод 3: "Взять прозрачность из RGBA"](#)
- [5 Вспомогательные операции](#)
- [6 Ссылки](#)

## Исходные данные

В качестве единственного источника выступает ЦМР, растровая матрица, где каждому пикселю соответствует значение его высоты над уровнем моря. Это может быть [SRTM](#), [ASTER GDEM](#) и другие источники.

Посмотреть на эти матрицы можно в QGIS в сером или цветном представлении.



ЦМР в палитре градаций серого

## Теневая отмывка

Создадим теневую отмывку на основе нашей ЦМР. Для многих операций с данными о рельефе в GDAL используется инструмент [gdaldem](#). Результатом операции является 8-битный растр, где каждому пикселю назначается интенсивность освещенности в зависимости от его положения. Также можно указывать высоту (-alt) солнца над горизонтом 0-90 и угол (-az) освещения солнца 0-360.

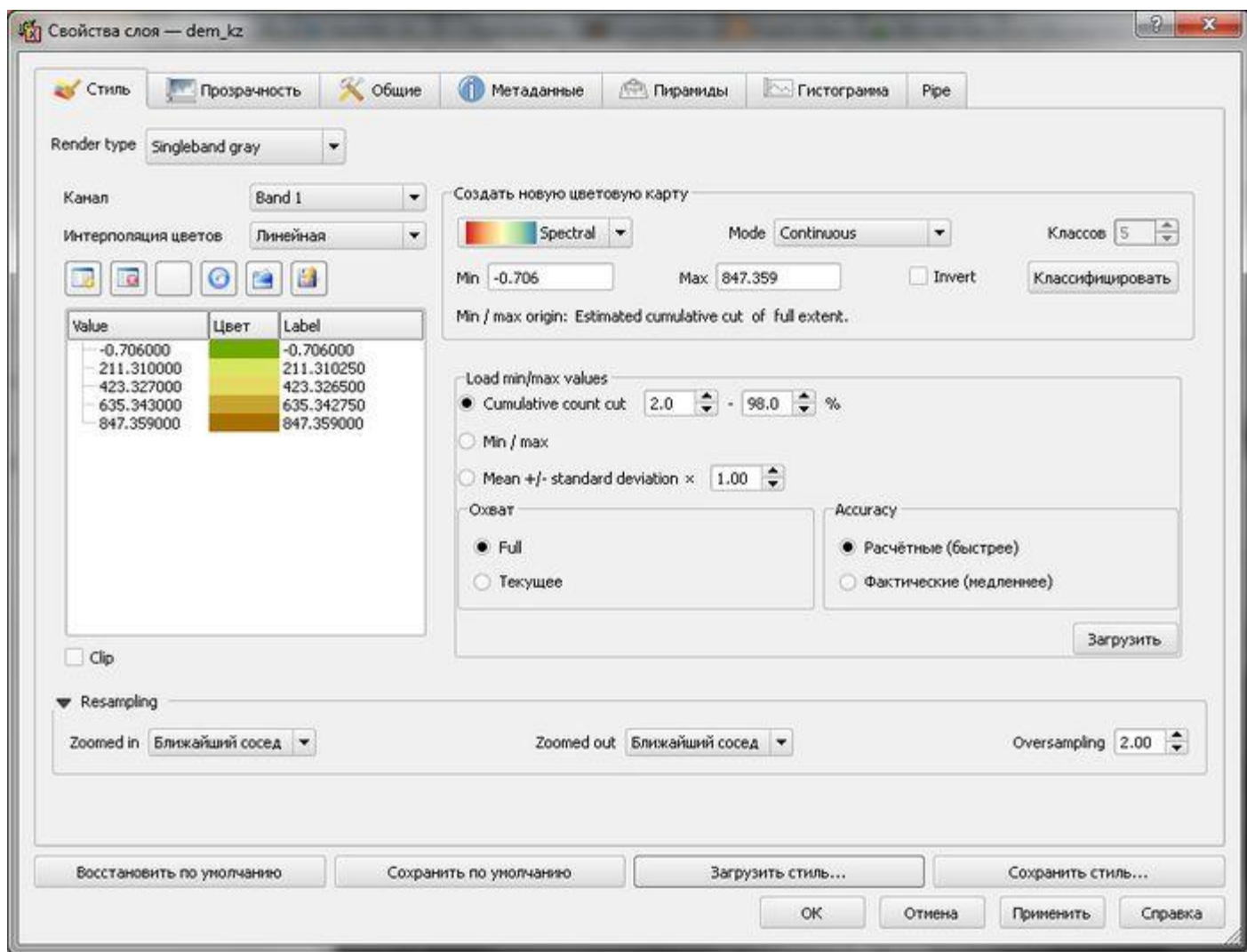
```
gdaldem hillshade dem.tif shade.tif -z 5 -az 90
```



Результат теневой отмывки рельефа

## Цветовой рельеф

Теперь необходимо сделать еще одну матрицу, где назначить каждому ее пикселю определенный цвет. Это очень удобно сделать в пользовательской ГИС, где посредством интерфейса можно задать количество высотных поясов, выбрать цвет, оценить результат.



Настройка цветового рельефа в QGIS

Для GDAL результат такого раскрашивания нужно преобразовать в текстовый файл следующего формата, пример:

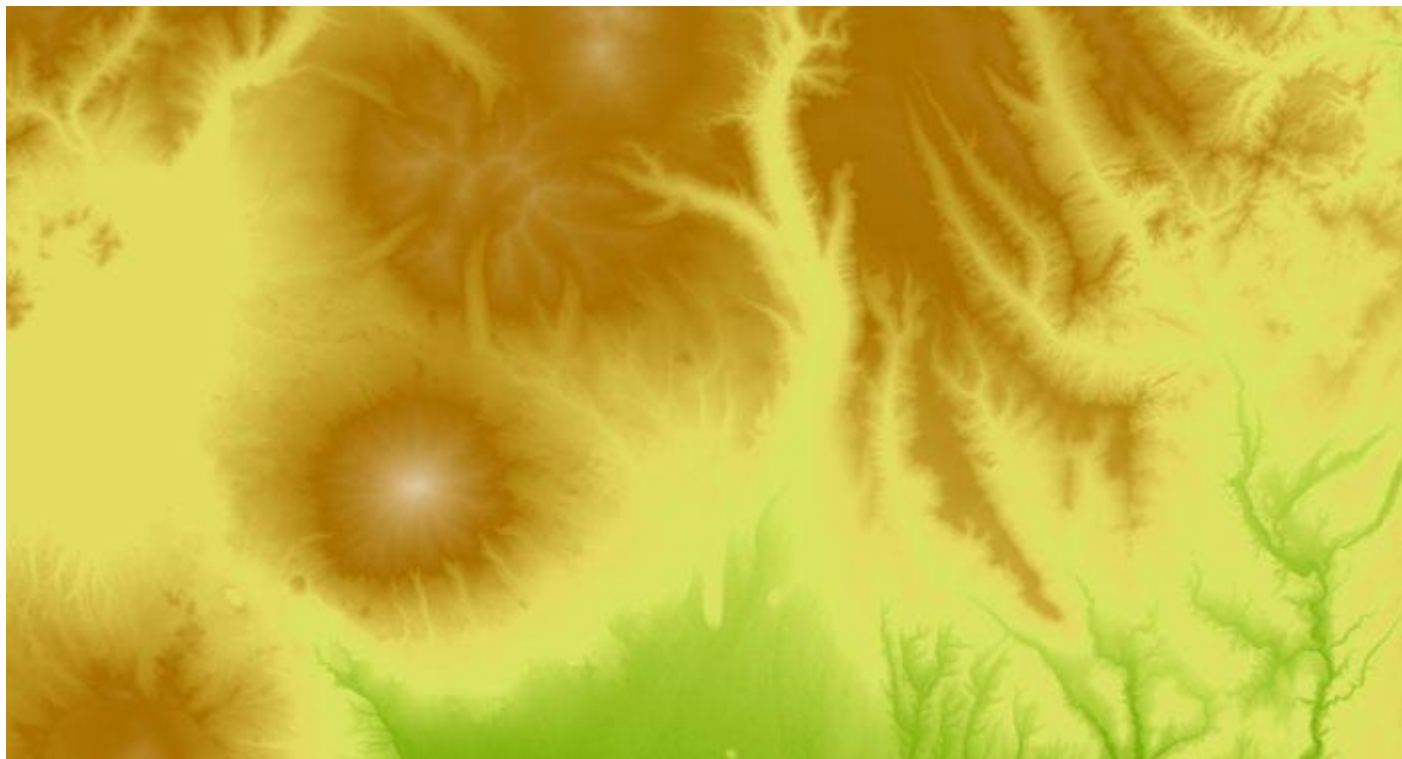
```
5000 255 255 255
1000 168 112 0
650 198 165 48
400 229 218 97
200 218 229 97
0 112 168 0
```

Где первая цифра - высота, а 2-3-4, ее цвет в RGB. В данном примере, все пиксели от 0 до 200 будут закрашены одним цветом, от 200 до 400 другим и так далее. Сохраним файл под названием ramp.txt

Дальше, выполним команду, которая на основе ЦМР создаст RGB изображение (информация о высотах таким образом будет потеряна, останется только цвет):

```
gdaldem color-relief dem.tif ramp.txt relief.tif
```





Настройка цветового рельефа в QGIS

## Комбинирование

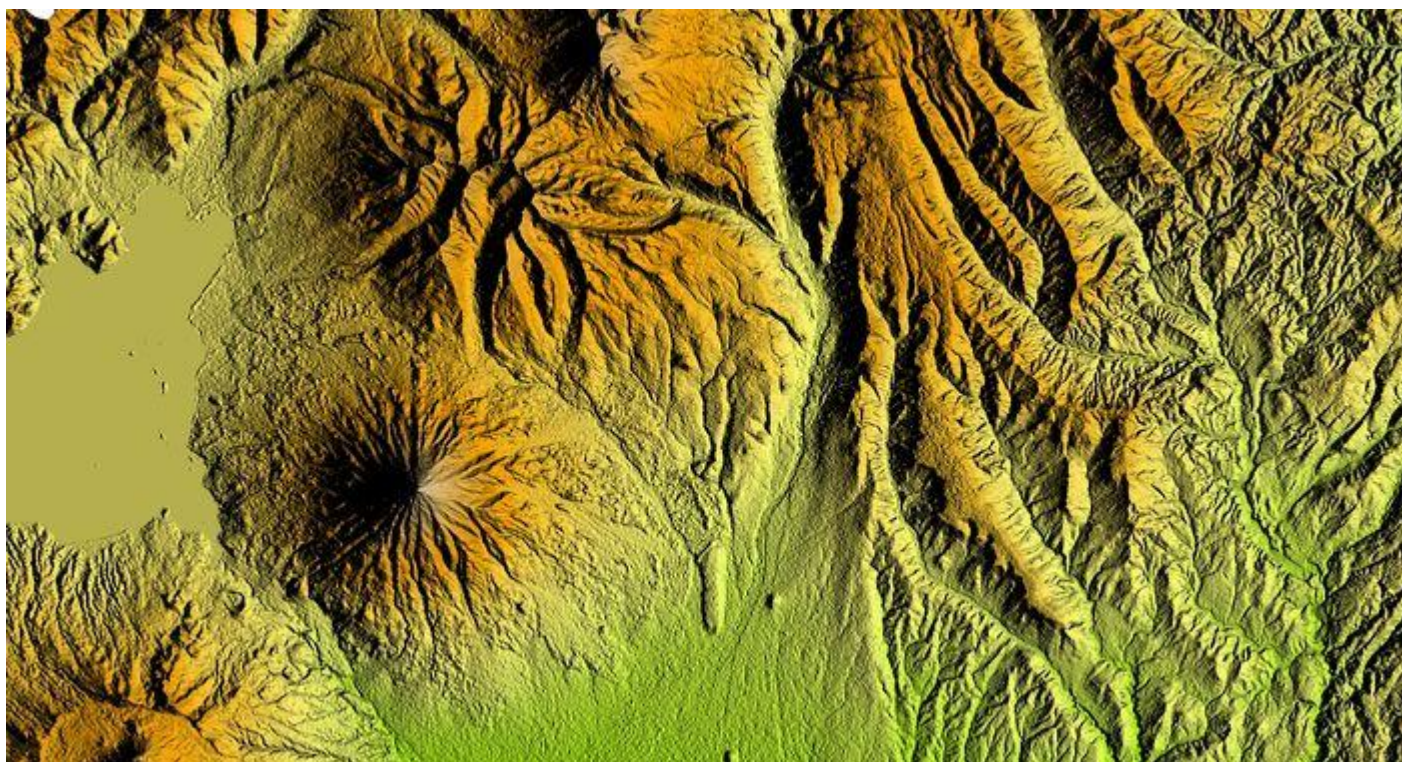
### Метод 1: "в HSV и обратно"

Один из методов сложения цветового рельефа и теневой отмывки - преобразование через HSV и обратно. В рамках этого метода, цветовой рельеф из RGB (red-green-blue) преобразуется в HSV (hue-saturation-value), получившаяся компонента V заменяется целиком на теневую отмывку, результирующий HSV преобразуется обратно в RGB.

Осуществить такое преобразование можно с помощью скрипта [hsv\\_merge.py](#) использующего возможности GDAL (автор: Frank Warmerdam, [подробнее о скрипте](#)).

Скрипт запускается следующим образом:

```
hsv_merge.py relief.tif shade.tif colour_shade.tif
```



Результат комбинирования теневой отмывки и цветового рельефа через HSV

## Метод 2: "Прозрачность как в ГИС"

Как можно видеть результат предыдущего метода и то, что мы получаем в пользовательских ГИС (самая первая иллюстрация) несколько различается. Мы также не можем в предыдущем методе менять прозрачность, чтобы достигнуть необходимого результата.

Чтобы получить точно такой же результат, нужно конвертировать не через HSV, а действовать простым сложением компонент по следующему принципу (подсмотрено на [Stackoverflow](https://stackoverflow.com), спасибо Дмитрию Барышникову за наводку!):

$$\begin{aligned} R &= r * (1 - \alpha) + h * \alpha \\ G &= g * (1 - \alpha) + h * \alpha \\ B &= b * (1 - \alpha) + h * \alpha \end{aligned}$$

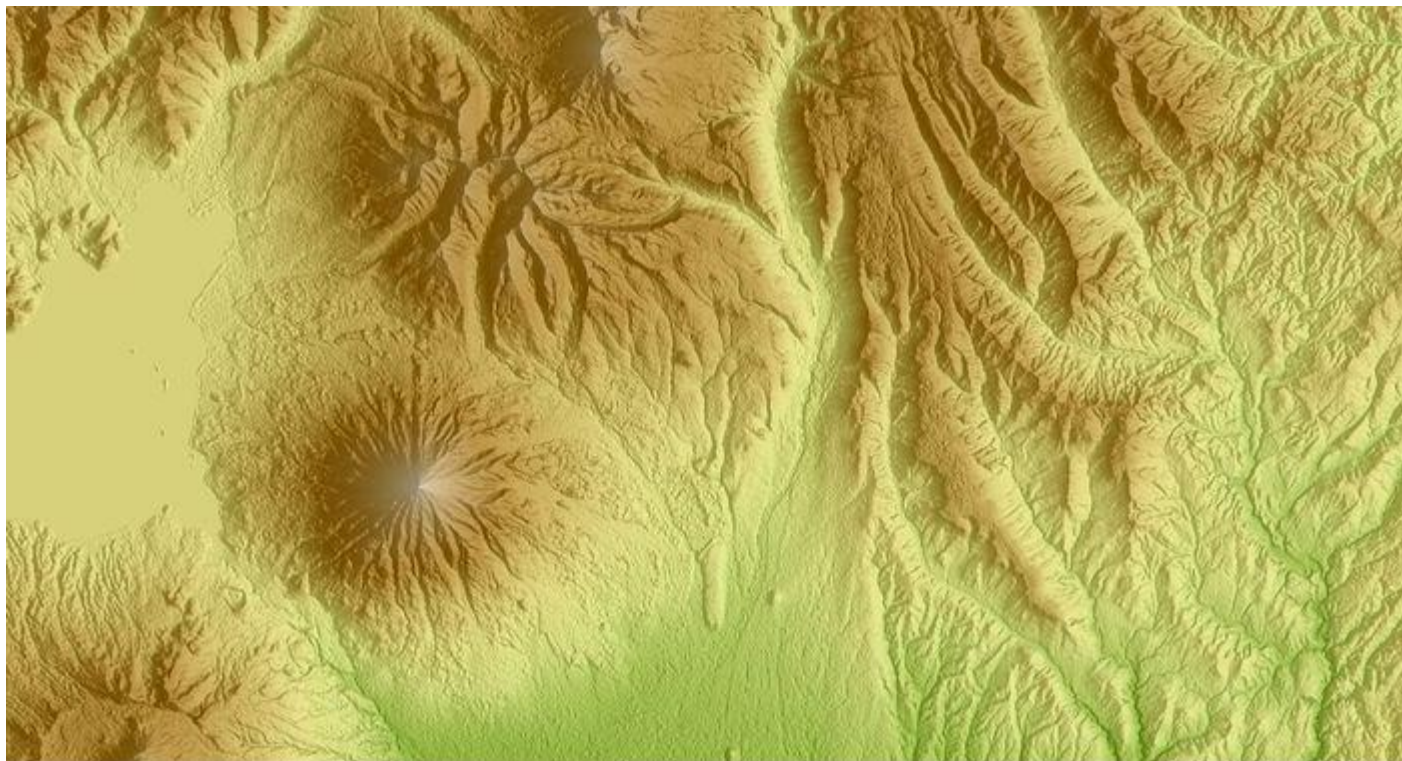
Где,  $r, g, b$  - каналы исходного изображения,  $h$  - растр теневой отмывки, а  $R, G, B$  - каналы результирующего изображения,  $\alpha$  - значение прозрачности от 0 до 1, 0 - верхний растр непрозрачен (нижний, отмывка, просто не будет виден), 1 - верхний растр полностью прозрачен, результат будет идентичен слою теневой отмывки.

Выполнить эти манипуляции можно с помощью скрипта `transparent_merge.py` (адаптирован автором статьи, [скачать](#)), следующим образом:

```
transparent_merge.py -alpha 0.3 relief.tif shade.tif colour_shade.tif
```

Таким образом, синтаксис у скрипта такой же, как и у `hsv_merge.py`, за исключением добавления параметра `alpha`, который указывает, сколько прозрачность добавить к цветовому рельефу (`relief.tif`). Результат будет такой же, как и наложение двух растров в пользовательской ГИС:



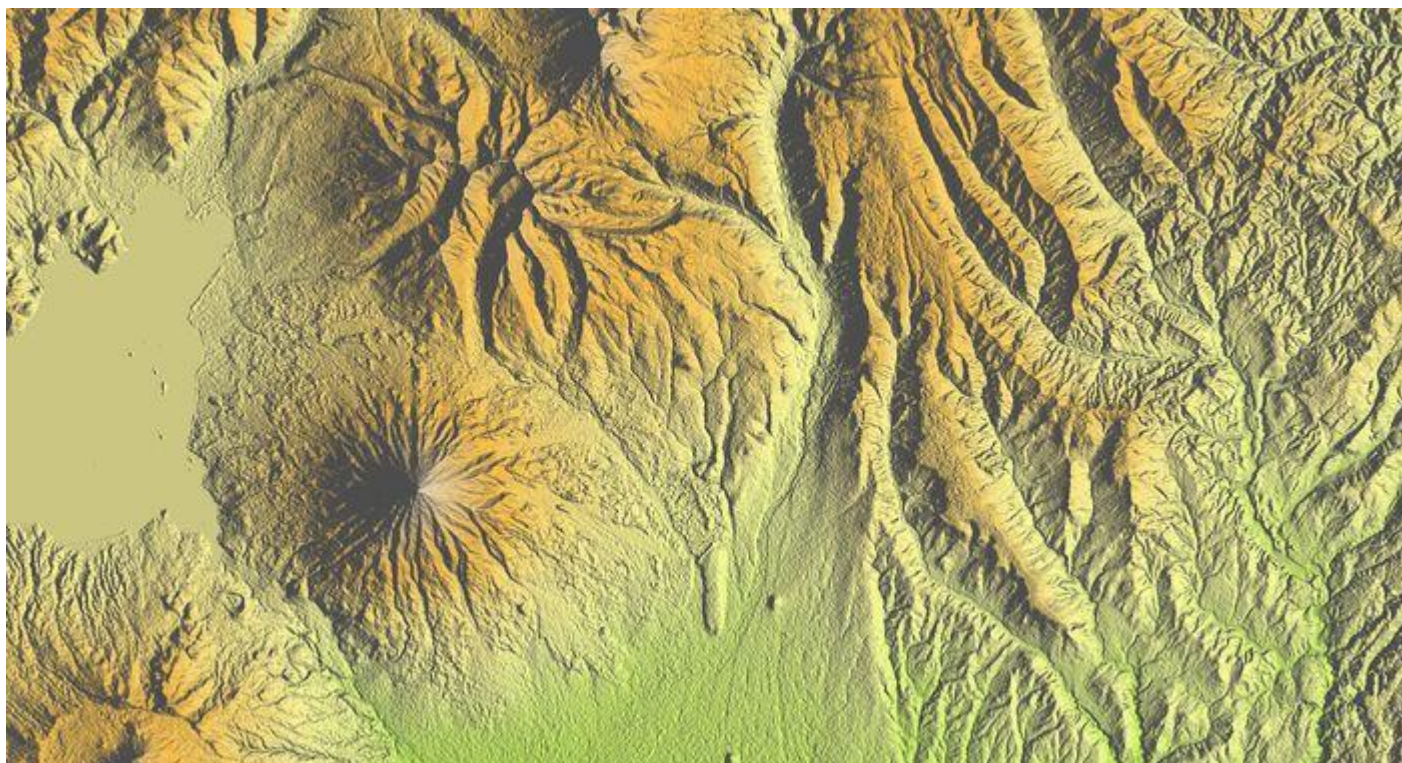


Результат комбинирования цветового рельефа и теневой отмывки с помощью transparent\_merge.py

### Метод 3: "Взять прозрачность из RGBA"

Исследование исходников hsv\_merge.py показывает, что в качестве исходного растра может использовать не только RGB, но и RGBA, т.е. растр с заданной прозрачностью. Хотя это похоже на то, что нам нужно, на самом деле это не оно, так как никакого учета канала прозрачности при цветовой трансформации не происходит, а канал прозрачности (альфа-канал) просто переносится с исходного изображения на результат перевода через HSV и обратно.

Чтобы воспользоваться этим методом, нужно сначала создать из RGB - RGBA, самый простой способ это сделать (не разобрался как это сделать в GDAL) следующий, нужно просто загрузить растр цветового рельефа в QGIS, задать ему прозрачность и щелкнув правой кнопки мыши по нему выбрать "Сохранить как" и далее "Rendered image". Растр сохранится с учетом прозрачности в RGBA.



Результат hsv\_merge.py с RGBA растром на входе (30% прозрачности)

## Вспомогательные операции

Часто встречающиеся задачи по доделке результата - обрезка по контуру и создание альфа-канала.

Обрезка по существующему сложному (необязательно прямоугольному контуру) может быть выполнена с помощью следующей команды:

```
gdalwarp -dstnodata 255 -cutline clip.shp -crop_to_cutline colour_shade.tif  
colour_shade_clip.tif
```

Создать и назначить определенное значение альфа-каналу можно с помощью следующей команды:

```
gdalwarp -dstalpha -srcnodata 255 colour_shade_clip.tif colour_shade_clip_alpha.tif
```

## Ссылки

- [A workflow for creating beautiful relief shaded dems using GDAL](#)
- [Working with terrain data](#)

[Обсудить в форуме](#) Комментариев — 9

Последнее обновление: 2014-05-15 01:46

Дата создания: 19.03.2013

Автор(ы): [Максим Дубинин](#)