shadow: R Package for Geometric Shadow Calculations in an Urban Environment

Michael Dorman, Adi Vulkan, Evyatar Erell, Itai Kloog
Ben-Gurion University of the Negev

2017-07-05
Contents

- Introduction & Motivation
- The shadow package
- Example 1: Shadow height
- Example 2: Shadow footprint
- Example 3: Sky View Factor
- Example 4: Annual radiation on facades
- Summary
Introduction

Urban planning frequently requires estimating whether a given point is shaded or not, given -

- Representation of spatial obstacles (e.g. buildings)
- Solar position

To estimate -

- Shadow impact of new building
- Microclimatic conditions
- Photovoltaic production potential

Figure 1: Shadow analysis

---

1Biljecki et al., 2015 ISPRS Int. J. Geo-Inf.
Introduction

**Urban planning** frequently requires estimating whether a given point is **shaded or not**, given -

- Representation of **spatial obstacles** (e.g. buildings)
- **Solar position**

To estimate -

- Shadow impact of new building
- Microclimatic conditions
- Photovoltaic production potential

---

Shadow analysis tools

**Raster-based 2.5D**

- Digital Elevation Models (DEM) may not be available and are expensive to produce
- Vertical urban elements (e.g. building facades) not represented

**Examples**

- GRASS GIS r.sun
- UMEP QGIS plugin
- R package insol

Figure 2: UMEP QGIS plugin\(^2\)

\(^2\)http://www.urban-climate.net/umep/UMEP_Manual
Shadow analysis tools

Vector-based 3D

- Mostly restricted to proprietary software
- Associated with specific 3D geometric model formats
- Aimed at visualization rather than quantitative analysis

Examples

- ESRI’s ArcScene
- SketchUp

Figure 3: SketchUp³

³http://extensions.sketchup.com/ko/content/shadow-analysis
Motivation

Adi Vulkan’s MA Thesis -

- Modelling the potential for PV installation in residential buildings in dense urban areas

We could not locate -

- Open-source solution
- Easily integrated with analysis workflow in R
- Suitable for building outline and height (i.e. extruded polygons) input

Therefore -

- Eventually implemented our own solution in R
Motivation

Adi Vulkan’s MA Thesis -

- Modelling the potential for PV installation in residential buildings in dense urban areas

We could not locate -

- Open-source solution
- Easily integrated with analysis workflow in R
- Suitable for building outline and height (i.e. extruded polygons) input

Therefore -

- Eventually implemented our own solution in R
Motivation

Adi Vulkan’s MA Thesis -

- Modelling the potential for PV installation in residential buildings in dense urban areas

We could not locate -

- Open-source solution
- Easily integrated with analysis workflow in R
- Suitable for building outline and height (i.e. extruded polygons) input

Therefore -

- Eventually implemented our own solution in R
## The shadow R package - Main functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Location</th>
<th>Obstacles</th>
<th>Sun Pos.</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>shadowHeight</td>
<td>Points</td>
<td>Polygons</td>
<td>Matrix</td>
<td>Numeric</td>
</tr>
<tr>
<td>shadowFootprint</td>
<td>-</td>
<td>Polygons</td>
<td>Matrix</td>
<td>Polygons</td>
</tr>
<tr>
<td>SVF</td>
<td>Points</td>
<td>Polygons</td>
<td>-</td>
<td>Numeric</td>
</tr>
</tbody>
</table>

Table 1: Inputs and outputs for main functions in package `shadow`

- **Location**: `SpatialPoints*` or `RasterLayer`
- **Obstacles**: `SpatialPolygonsDataFrame` with height attribute
- **Sun Position**: matrix with sun azimuth and elevation angles
The shadow R package - Installation & Loading

▶ Installing CRAN release
```r
install.packages("shadow")
```

▶ Installing development version
```r
devtools::install_github("michaeldorman/shadow")
```

▶ Package loading
```r
library(shadow)
library(raster)
library(rgeos)
```
Inputs layers - Map

plot(build, col = "lightgrey")
text(gCentroid(build, byid = TRUE), build$BLDG_HT)
plot(park, col = "lightgreen", add = TRUE)

Figure 4: Buildings (height in m) and green park
Example 1: Shadow height

\[ h_{\text{shadow}} = h_{\text{build}} - \text{dist} \cdot \tan(\alpha) \]
Example 1: Shadow height

```r
# Point
location = rgeos::gCentroid(build)

# Time
time = as.POSIXct(
  "2004-12-24 13:30:00",
  tz = "Asia/Jerusalem"
)
```
Example 1: Shadow height

```cpp
# Location in geographical coordinates
location_geo = sp::spTransform(
    location,
    "+proj=longlat +datum=WGS84"
)

# Solar position
solar_pos = maptools::solarpos(
    crds = location_geo,
    dateTime = time
)
solar_pos

## [,1] [,2]
## [1,] 208.7333 28.79944
```
Example 1: Shadow height

```r
# Shadow height at a single point
h = shadowHeight(
  location = location,
  obstacles = build,
  obstacles_height_field = "BLDG_HT",
  solar_pos = solar_pos
)
```

```r
# Result
h
```

```r
## [,1]
# [1,] 19.86451
```
Example 1: Shadow height

Figure 6: Shadow height at a single point
Example 1: Shadow height

```r
# Raster template
ext = as(raster::extent(build)+50, "SpatialPolygons")
r = raster::raster(ext, res = 2)
proj4string(r) = proj4string(build)

# Shadow height surface
height_surface = shadowHeight(
  location = r,
  obstacles = build,
  obstacles_height_field = "BLDG_HT",
  solar_pos = solar_pos,
  parallel = 2
)
```
Example 1: Shadow height

Figure 7: Shadow height (m) surface
Example 1: Shadow height

Figure 8: Shadow height (m) surface with individual point
Example 2: Shadow footprint

Figure 9: Shadow footprint of building ‘1’ at different times on 2004-06-24
Example 2: Shadow footprint

```r
# Another solar position

time2 = as.POSIXct(
  "2004-06-24 09:30:00",
  tz = "Asia/Jerusalem"
)
solar_pos2 = maptools::solarpos(
  crds = location_geo,
  dateTime = time2
)
solar_pos2

## [,1] [,2]
## [1,] 88.83113 46.724
```

```r
```
Example 2: Shadow footprint

```r
# Shadow footprint polygons
footprint = shadowFootprint(
  obstacles = build,
  obstacles_height_field = "BLDG_HT",
  solar_pos = solar_pos2
)

# Intersection with park
park_shadow = rgeos::gIntersection(park, footprint)
shade_prop =
  rgeos::gArea(park_shadow) / rgeos::gArea(park)
shade_prop
```

```r
# [1] 0.3447709
```
Example 2: Shadow footprint

Figure 10: Shaded park proportion on 2004-06-24 09:30:00
Example 2: Shadow footprint

Figure 11: Shaded park proportion at hourly intervals on 2004-06-24
Example 3: Sky View Factor

Figure 12: SVF = Fraction of sky visible from the ground up \([0 – 1]\)
# Sky View Factor (SVF) of a single point

```python
s = SVF(
    location = location,
    obstacles = build,
    obstacles_height_field = "BLDG_HT"
)
```

# Result

```r
s
```

```r
## [1] 0.4390933
```
Example 3: Sky View Factor

Figure 13: Sky View Factor (SVF) of a single point
Example 3: Sky View Factor

# Sky View Factor (SVF) surface
svf_surface = SVF(
    location = r,
    obstacles = build,
    obstacles_height_field = "BLDG_HT",
    parallel = 2
)
Example 3: Sky View Factor

Figure 14: Sky View Factor (SVF) surface
Example 3: Sky View Factor

Figure 15: Sky View Factor (SVF) surface with individual point
Example 4: Annual radiation on facades

Package shadow was used to estimate facade shading on -

- A neighborhood of 410 buildings
  - 506,567 pixels of $1 \cdot 1m^2$ covering facade area
  - 162,249 pixels of $1 \cdot 1m^2$ covering roof area

- Entire year
  - 8760 hours covering a Typical Meteorological Year
Example 4: Annual radiation on facades

Figure 16: Annual radiation on $1 \cdot 1m^2$ pixels ($kWh/m^2/yr$)
Summary

The `shadow` package offers -

- **Strightforward** estimation of shading and SVF in urban environments
- Based on **generally available** polygonal building data
- Easy **integration** with general spatial analysis workflow in the R ecosystem

Can be used for -

- Determining whether a given location is shaded or not
- Calculating shaded surface area
- Estimating Sky View Factor of ground locations
Summary

The shadow package offers -

- **Strightforward** estimation of shading and SVF in urban environments
- Based on **generally available** polygonal building data
- Easy **integration** with general spatial analysis workflow in the R ecosystem

Can be used for -

- Determining whether a given location is shaded or not
- Calculating shaded surface area
- Estimating Sky View Factor of ground locations
Limitations

The model assumes -

- Flat terrain
- Polygon-extrusion obstacles

This may not be suitable for all cases -

- Complex shape of buildings or other obstacles (e.g. trees)
- Non-flat topography
Limitations

The model assumes -

▶ Flat terrain
▶ Polygon-extrusion obstacles

This may not be suitable for all cases -

▶ Complex shape of buildings or other obstacles (e.g. trees)
▶ Non-flat topography
Thank you for listening!

**Funding**

- Ministry of National Infrastructures, Energy and Water Resources

**Comments and suggestions**

- Michael Dorman (dorman@post.bgu.ac.il)
- https://cran.r-project.org/package=shadow
- https://github.com/michaeldorman/shadow