A NEW METHOD FOR THE CALCULATION OF THE SKY VIEW FACTOR FOR NON-RECTANGULAR SURROUNDINGS

Matthias Gladt[1] and Thomas Bednar[1]
[1]Vienna University of Technology, Vienna Austria

ABSTRACT

The sky view factor (SVF) represents the relationship between the visible portion of the sky and the portion covered by the surrounding objects.

It is significant e.g. in urban zones where the nightly cooling of a building has an intense correlation with the long-wave radiation emitted by the surrounding objects.

The methods of calculating the SVF comprise fisheye-lens photographs analysis as well as mathematical models or image processing. A well-known method for the calculation is based on a vector-based presentation of the model. The projection of the model onto the hemisphere is divided into slices to approximate the calculation of the resulting polygon with the calculation of the SVF for the rectangles resulting from the slices of the previous step.

The approach has two major drawbacks: The process of discretization slows the calculation down and the accuracy of the result depends on the particularity of discretization. This paper presents a method where no more discretization is needed which results in lower calculation times and more accurate results.

INTRODUCTION

Especially in urban areas the nightly cooling of the buildings is limited because the densely built-up area prevents an effective heat rejection as it takes place in rural areas. Buildings act as heat accumulators but whereas in rural areas they practically only give off heat in urban areas they also receive heat from their neighboring buildings. Gál et al. (2007) coin the term, "urban heat island" to clarify the relevance of surrounding objects for the heat balance of a particular building. A common way of calculating the sky view factor (SVF) which is elaborated in many publications is based on graphical methods and in particular on hemispherical photographs with a fisheye photographic lens. (E.g. Watson and Johnson, 1988)

However Matuschek and Matzarakis (2010) point out that the process of taking a picture is error-prone and can usually only be completed if an existing environment shall be evaluated. So it may not be possible to run simulations in an early phase of a project where only computer models exist. They evaluate "SkyHelios", a software tool, which can be used to calculate the SVF. It's based on digital elevation models of the relevant scenario and returns promising results.

Rakovec and Zakšek (2012) take into account the diffuse tilt factor in addition to the SVF. They present a method for the calculation of the complete diffuse radiation but they restrain to a flat slope rather than to projections from arbitrary objects.

Bradley, Thornes and Chapman (2001) focus on calculating the SVF value from hemispherical photographs with a fisheye photographic lens. They suggest using GPS in connection with it to avoid errors when building up a SVF database. They also take into account the density of the urban development to make out differences in the SVF value for particular classes of urban regions.

Gál et al. compare the SVF values received by a fisheye-lens photograph analysis with values calculated from an urban vector database. Their method of calculating the SVF from a vector database is analyzed in detail on the following pages. The fisheye-lens photographs are divided into a number of concentric annuli of equal width, each representing an interval of zenith angles. The results of both methods deviate by an average value of 0.106 (regarding the relevant SVF values) from each other which is mainly owed to the fact that vegetation is not in the database used as input for the calculations.

Matzarakis and Matuschek (2007) present "Ray-Man", a model used for the calculation of short- and long-wave radiation fluxes on the human body. RayMan also provides an interface for free drawings and fish-eye photographs to calculate the SVF.

Johnson and Watson (1984) present a method where they set up a polar coordinate system to calculate SVF values for finite and infinite canyons enclosed by walls of constant height. The method outlined in what follows is partly related to their approach but is based on a different coordinate system. Besides also walls with varying height with respect to the longitudinal axis are considered.

Källblad (1999) presents a generic function to calculate the SVF value and Oke (1987) who defines the sky view factor (SVF) as the ratio of the amount of the sky "seen" from a given point on a surface to that potentially available also gives values for commonly occurring geometric arrangements like valleys or slopes or a basin of a constant radius and of constant height. Based on these values the SVF can also be used to quantify the characteristics of a building's surroundings which are significant for the amount of heat that is received.

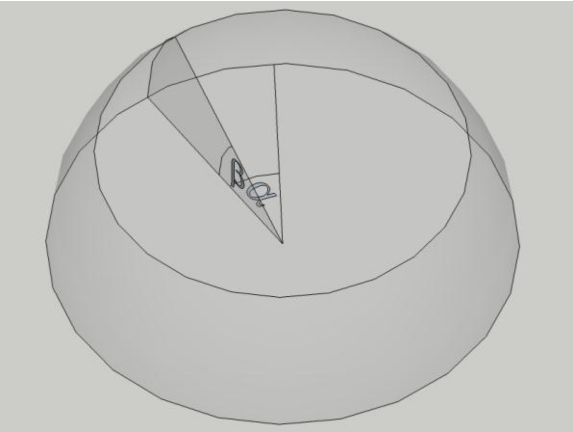

Figure 1 Basin with the opening angle of β representing the projection of a virtual surrounding of constant height on the hemisphere

The SVF for the whole basin (Figure 1) given by Oke is:

$$\psi_{sky} = \cos^2 \beta \tag{1}$$

So the SVF in this case represents the area of the opening of the basin to the area of the middle circle quotient. If only the fraction represented by α shall be taken into account according to Gál et al. (2007) the SVF results to:

$$\psi_{sky} = \cos^2 \beta \left(\frac{\alpha}{2\pi} \right) \tag{2}$$

The calculation methods presented in the paper at hand are partly based on Equation 1 and 2.

A solution for the calculation of the SVF for generic surroundings in terms of geometry is outlined. Gál et al. also base their approach of calculating the SVF from a vector database on Oke's value of the SVF for the basin mentioned above by dividing the hemisphere into slices and calculating the SVF for each slice. The drawbacks of their solution may regard performance issues and accuracy depending on the chosen level of discretization. With the raise of resolution of discretization not only the accuracy of the result may improve but also the amount of time consumed by a single simulation run may go up. The solution in the paper at hand avoids these drawbacks by presenting a new calculation method where no more discretization has to be made.

METHOD

Transformation of the scenery into a new coordinate system

For demonstration purposes we chose a simple scenario with two faces representing two of the enveloping surfaces of an object. They can also be interpreted e.g. as the facades of two buildings.

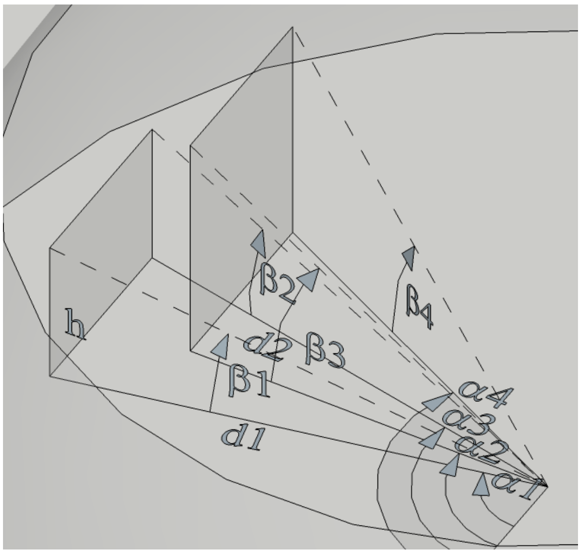

Figure 2 Two faces in relation to a thought part of the hemi-sphere. Vertical angles are labeled with β_{1-4}, horizontal ones with α_{1-4}.

The calculation of the SVF is based on the central projection of the relevant scenario. The origin is set at the point that the SVF shall be referred to. The projection of the surrounding buildings has to be specified by a function of the horizontal angle, α, returning the vertical angle β rather than by a set of Cartesian coordinates. The transformation from the Cartesian coordinate system into a new α, β coordinate system brings along a distortion of originally straight lines.

In the example of Figure 2 vertical edges remain vertical in the projection, horizontal edges, however, must be specified by the tangent of the height to distance quotient. (Figure 3)

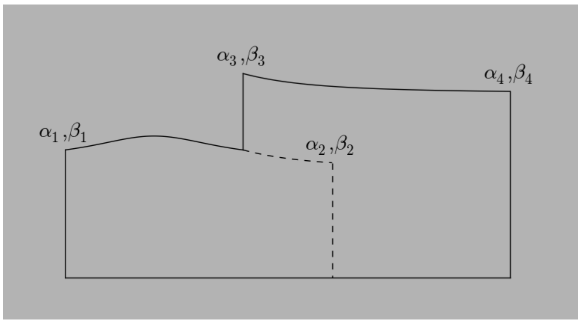

Figure 3 Schematic representation of the central projection of the same scenario from the position of the origin as a function of the horizontal angle α.

The exact shape of non-vertical edges depends on the geometric circumstances. In the example at hand the calculation of β_1 and β_2 is trivial:

$$\beta_1 = \arctan\left(\frac{h_1}{d_1}\right) \tag{3}$$

and

$$\beta_2 = \arctan\left(\frac{h_1 + \Delta h}{d_2}\right) \tag{4}$$

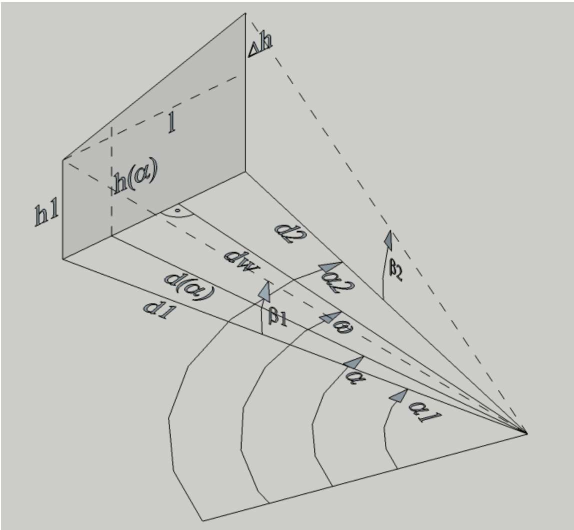

Figure 4 Geometry of one of the faces of the length, l, related to the point that the SVF refers to. d_w denotes the distance of the face from the origin and ω the angle of the line orthogonal to the face.

From Figure 4 it can be seen that the function describing β between β_1 and β_2 results to:

$$\beta(\alpha) = \arctan\left(\frac{h(\alpha)}{d(\alpha)}\right) \tag{5}$$

and

$$d(\alpha) = \frac{d_w}{\cos(\omega - \alpha)} \tag{6}$$

$$h(\alpha) = h_1 + \Delta h \cdot \frac{d_1 \sin(\omega - \alpha_1) - d(\alpha)\sin(\omega - \alpha)}{l} \tag{7}$$

So the originally horizontal straight edges of the two faces become curves in the projection illustrated in Figure 3.

Since only the difference between two angles is used in the calculations any reference axis can be chosen for all relevant angles.

Discretization approach

The projection resulting from the previous example is displayed in Figure 5.

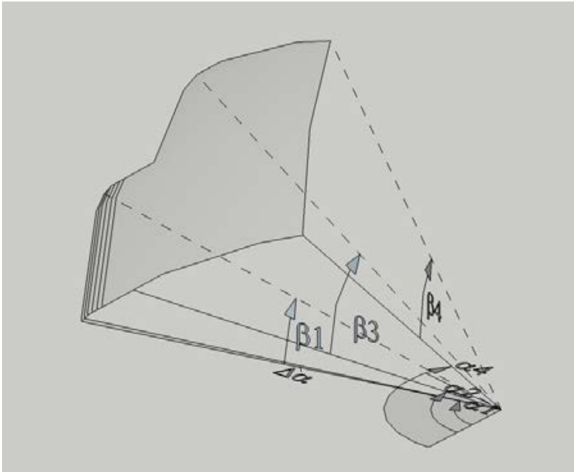

Figure 5: Resulting projection of two faces onto the hemisphere discretized by dividing it into slices of equal width.

Once the projection on the hemisphere as a function of the horizontal angle, α, is complete the SVF can be calculated based on Equation 2.

Figure 5 illustrates how Gál et al. (2007) divide the projection into slices, each covering the same fraction of the angle α, $\Delta\alpha$. They choose the middle points of the intervals as values for β. However under some unfavorable conditions the middle point may not present the average value of β. (Figure 6) A higher level of discretization can improve the accuracy of the result but will have a negative impact on the performance.

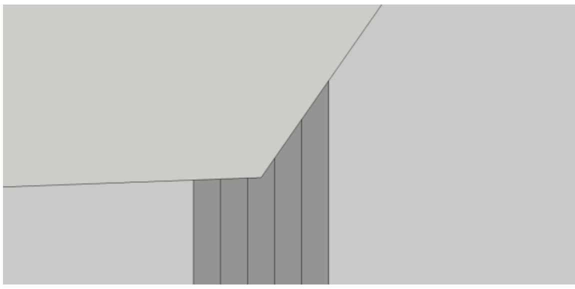

Figure 6: Discretization matches well left and right of the corner of the model. The slice in the middle produces an inaccurate result because its middle value is too low.

Functional based approach

An alternative approach is based directly on Equation 2. The SVF of a fraction of the basin can also be interpreted as the result of an integral:

$$\psi_{sky} = \cos^2\beta \frac{\alpha_2 - \alpha_1}{2\pi} = \frac{\alpha_2 - \alpha_1}{2\pi} - \frac{1}{2\pi}\int_{\alpha_1}^{\alpha_2} \sin^2\beta(\alpha)\,\mathrm{d}\alpha \tag{8}$$

The "terrain view factor" is defined as $1 - \psi_{sky}$.

$$\psi_{terrain} = 1 - \left(\frac{\alpha_2 - \alpha_1}{2\pi} + \frac{1}{2\pi} \int_{\alpha_1}^{\alpha_2} \sin^2 \beta(\alpha)\, \mathrm{d}\alpha \right) \qquad (9)$$

$\beta(\alpha)$ can be obtained from Equation 5. Its form is independent of the inclination of an edge so no additional complexity occurs with real scenarios. So what remains to be done is to accomplish the integration of $\sin^2\beta(\alpha)$.

There is no symbolic solution for it but the calculation of the numerical solution is no problem with common math programs.

The relation between $h(\alpha)$ and $d(\alpha)$ has the most significant impact on the SVF. The point of reference that the SVF is calculated for is marked POR in Figure 7.

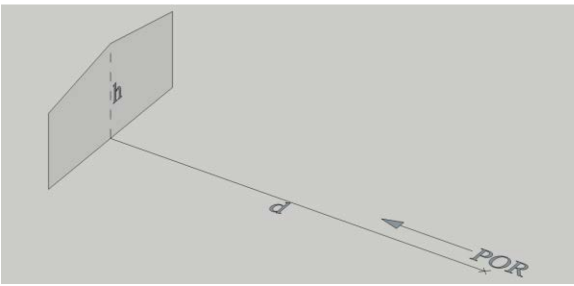

Figure 7 The h to d quotient is a critical factor for the calculation of the SVF.

The h to d quotient rises as the distance between the point of reference (POR) and the relevant building declines. Figure 8 illustrates the run of the SVF-curve for an h to d quotient starting with 0 representing a building in infinite distance up to 10 representing a building with the top edge at a vertical angle of about 85 degrees. As the h to d quotient grows also the horizontal opening angle $\alpha_2 - \alpha_1$ grows approximately at the same rate as the vertical angle.

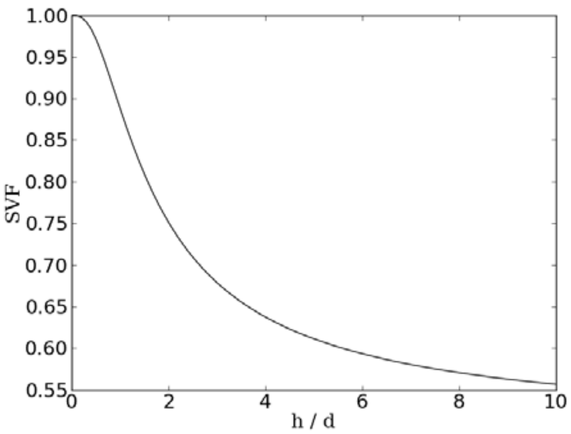

Figure 8 The SVF depends on the relation between h and d. At an h to d quotient between 3 and 5 the SVF curve starts to flatten.

From Figure 8 it can be seen that the exact calculation of the SVF is less important for h to d quotients above about 6. However for low h to d quotients also small inaccuracies can cause large errors.

EVALUATION

Whereas the performance and the accuracy of the calculation of the SVF depend on the number of slices in case of the discretization approach there is no more need for discretization in case of the functional-based approach.

Accuracy

The accuracy in case of discretization depends mainly on two parameters:

- The geometry of objects which are part of the scenario to be calculated.
- The width of the slices and the level of discretization respectively.

The calculation of the SVF of a particular slice is based on the middle value of β which is used as an average value for the height of the whole slice. So in case of straight edges in the α-β-projection there is no error at all because the middle value exactly matches the average value.

However if there is a vertex causing an offset in the projection that does not coincide with the border of a zone an error is caused. (Figure 9)

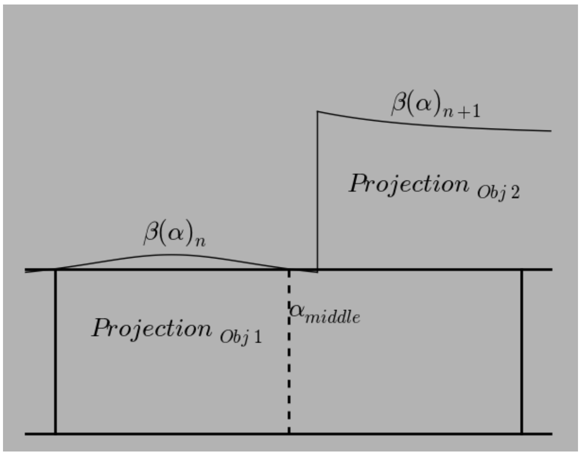

Figure 9 The middle line of a slice is close to an offset in the projection of two objects. Thus the error in the region of the slice becomes significant.

The size of the error depends on how much the average value differs from the middle value of the relevant slice. For a particular case the error was calculated for different h to d quotients.

In the following example a set of two buildings with a total opening angle of 6 degrees is investigated. Since the calculation of the SVF comprises a calculation for 360 degrees the scenario is assumed to be repeated for a complete rotation.

Table 1: Error in the calculation of the SVF for the scenario illustrated in Figure 9. There are two types of objects, one with a horizontal opening angle, α_2 - α_1, of 3.01 degrees, the other one with 2.99 degrees. The h to d quotient of the second object is kept constant at a value of 8.0. The width of the slices used to discretize the scene was chosen with 2.0 degrees.

H / D (OBJ 1)	H / D (OBJ 2)	ERROR BETWEEN FUNCTIONAL AND DISCRETIZED CALCULATION IN %
0.5	8.0	31.7
1.0	8.0	30.9
3.0	8.0	24.2
4.0	8.0	19.3
5.0	8.0	14.1
6.0	8.0	9.1
7.0	8.0	4.3
8.0	8.0	0.0

The figures in Table 1 clarify that the discretized calculation matches the exact one well for small differences between the relevant *h* to *d* quotients. Also rather large opening angles of the slices in relation to the opening angles of the buildings lead to good results if the edge of slice happens to be close to the edge of a projected building. However the size of the error remains difficult to predict.

Accordingly a high number of offsets (e.g. caused by many different buildings being projected on the hemisphere) increase the risk of slices causing an error.

To avoid for sure as unfavorable conditions as represented in the top part of Table 1 slices with an opening angle clearly below the opening angle of the relevant buildings should be chosen.

Performance

At the same time as the opening angle of the slices is decreased the number of calculation steps is increased. With the functional-based approach one calculation step is needed for every edge of the projection.

Table 2: Number of calculation steps of a complete SVF calculation (360 degrees) with the functional based approach compared with the discretization approach depending on the horizontal opening angle, α_2 - α_1, of the slices and the number of projected objects.

NUMBER OF PROJECTED OBJECTS	OPENING ANGLE OF A SLICE, α_2 - α_1	NUMBER OF CALCULATION STEPS FOR DIFFERENT CALCULATION METHODS	
		FUNCTIO-NAL	DISCRE-TIZED
15	2.0	15	180
25	2.0	25	180
25	4.0	25	90
50	2.0	50	180
100	2.0	100	180
100	4.0	100	90

Obviously the number of calculation steps for the functional based approach depends uniquely on the number of objects that are taken into regard whereas for the discretization approach it depends on the opening angle of a single slice.

For testing purposes sample scenarios were created containing a number of objects depending on a random number representing the opening angle of an object. The upper bound of the opening angle was set to 90 and the lower bound to 25 degrees resulting into 3 to 5 objects. The h / d quotients were also generated based on a random number with an upper bound according to the figures in Table 3 and a lower bound of 0. The opening angle for the slices was set to 2.0 degrees. The simulation was run 100 times with each set of parameters. Table 3 contains the average values for all 100 test runs of each test run.

Table 3: Error in the calculation of the SVF for random test scenarios. Maximum h / d values were varied between 0.5 and 15

MAXIMUM H / D	ERROR BETWEEN FUNCTIONAL AND DISCRETIZED CALCULATION IN %
0.5	1.6
1.0	2.1
3.0	2.4
4.0	2.3
5.0	2.5
6.0	2.0
8.0	2.4
10.0	2.0
15.0	1.7

CONCLUSION

A new method of calculating the SVF has been presented which is based on the calculation of the vertical angle β as function of the horizontal angle α. With the input of the geometry in a vectorial form the calculation of the function $\beta(\alpha)$ is possible for horizontal edges as well as for inclined ones. The geometric circumstances have been illustrated in order to set up the function that needs to be integrated to calculate the "terrain view factor" which is defined as *1-SVF*. The integral that needs to be calculated for the terrain view factor cannot be obtained in a symbolic form anymore but common math programs like Wolfram's Mathematica or MathWorks' Matlab or Python's SciPy can provide numerical solutions in real time.

Other approaches exist, like the ones presented by Gál, Rzepa, Gromek and Unger 2007 based on an urban vector database. They divide the hemisphere into slices representing equally sized parts of the hemisphere. Oke gives SVF values for commonly occurring geometric arrangements (1987), one of them a basin of constant height. The calculation based on the division of the hemisphere into slices as well as the functional based calculation use Oke's SVF value for the basin.

The calculation based on the division of the hemisphere is simple and works well for most scenarios. In some rare cases however a significant error may occur in particular if the opening angle of the slices is close to the horizontal opening angle of the surrounding objects.

With the functional based approach the exact calculation of the SVF can be guaranteed at any time.

Scalability of solutions provided for building simulation tools is substantial for the integration of a particular component within the whole architecture of a tool. With the functional based approach the computational effort is proportional to the number of objects to be taken into account. The implementation is complicated by the transformation of the Cartesian coordinate system into an α, β system. Thus straight lines are specified by a set of trigonometric functions which complicates the calculation of intersections of different objects' projections onto the hemisphere.

The computational effort for calculating the SVF with slices of the hemisphere depends on the level of discretization. A drawback of the approach is that the accuracy of the calculation grows at the same rate as the performance drops. The error depends mainly on two parameters: The dimensions of the objects to be taken into account and the level of discretization. Even though there is no fail-safe way of quantifying it, it hardly ever exceeded a remarkable degree in the random test scenarios even with an opening angle above 2 degrees.

Acknowledgements

Research funded by the Zentrum für Innovation und Technologie GmbH (ZIT) and the Federal Ministry for Transport, Innovation and Technology.

REFERENCES

Bradley A. V., Thornes J. E., Chapman L. 2001. Variation and prediction of urban canyon geometry from sky-view factor transects. Atmos. Science Letters pp 1?11 doi:10.1006/asle2001.0031.

Gál T., Rzepa M., Gromek B. and Unger J., 2007. Comparison Between Sky view factor values computed by two different methods in an urban environment. Acta Climatologica et Chorlogica, Universitatis Szegediensis, Tomus 40-41, 17-26, http://www2.sci.u-szeged.hu/eghajlattan/akta07/017-026.pdf

Johnson G. D., Watson I. D., 1984. The determination of view-factors in urban canyons. J. Clim. Appl. Meteorol. 23, 329?335.

Källblad K., 1999. A method to estimate the shading of solar radiation theory and implementation in a computer program. Proceedings of the 6th International IBPSA Conference: Building Simulation '99, International Building Performance Simulation Association, Kyoto (1999), pp. 595–601

MathWorks Matlab. http://www.mathworks.de/products/matlab/

Matuschek O., Matzarakis A., 2010. Estimation of sky view factor in complex environment as a tool for applied climatological studies. Berichte des Meteorologischen Instituts der Albert-Ludwigs-Universität Freiburg

Matzarakis A., Rutz F., Mayer H., 2007. Modelling Radiation fluxes in simple and complex environments. Application of the RayMan model. Int. J. Biometeorol. 51, 323-334

Oke, T. R., 1987. *Boundary layer climates*, Routledge, London and New York, 1987

Python Programming Language – Official Website, http://www.python.org/

Rakovec J., Zakšek K., 2012. On the proper analytical expression for the sky-view factor and the diffuse irradiation of a slope for the isotropic sky. Renewable Energy, 0960-1481 0960 1481

SciPy, package for scientific computing with Python [online] http://numpy.scipy.org/ [Accessed 4 November 2012]

Watson I. D., Johnson, G. T., 1988. Estimating person view factors from fish-eye lens photographs. Int. J. Biometeorol

Wolfram Mathematica. http://www.wolfram.com/mathematica/