Simulating Rainfall, Water Evaporation and Groundwater Flow in Three-Dimensional Satellite Images with Cellular Automata

Moisés Espínola[a], José A. Piedra-Fernández[a], Rosa Ayala[a], Luis Iribarne[a], Saturnino Leguizamón[b], James Z. Wang[c]

[a] *Applied Computing Group, University of Almería, Spain*
[b] *Regional Faculty, National Technological University, Mendoza, Argentina*
[c] *The Pennsylvania State University, University Park, Pennsylvania, USA*

Abstract

Remote sensing has been used in numerous environmental simulations with the aim of solving and improving many different kinds of problems, e.g., meteorology applications, soil quality studies, water resource exploration, and environmental protection. Cellular automata have been widely used in the field of remote sensing for simulating natural phenomena over two-dimensional satellite images. However, simulations on Digital Elevation Model (DEM), or three-dimensional (3D) satellite images, are rare. This paper presents a study of modeling and simulation of the weather phenomena of rainfall, water evaporation and groundwater flow in 3D satellite images through a new algorithm named RACA (RAinfall with Cellular Automata). The aim of RACA is to obtain, from the simulation, numerical and 3D results related to the water level that allow us to make decisions on important issues such as preventing the destruction of human life and property from future natural disasters in urban areas with probability of flooding, establishing future construction projects and urban planning away from locations with high probability of flooding, or estimating the future water supply of a population.

Keywords:
Cellular Automata, Remote Sensing, DEM Satellite Images, Water Simulation

1. Introduction

This paper presents research applying the methodology of cellular automata to model and simulate the meteorological phenomena of rainfall, water evaporation, and groundwater flow to Digital Elevation Model (DEM) satellite images provided by NASA, showing hydrological results in both numerical and three-dimensional of flow remaining water. Remote sensing [18] allows us to acquire information about the surface of the land and environmental information values [2] without having actual contact with the area being observed [5]. Examples of remotely-sensed applications include searching for water resources, ascertaining soil quality, addressing environmental protection, and creating meteorology simulations, among many others [11].

The weather phenomena have been extensively modeled and simulated by the worldwide scientific community. A deep investigation into the nature of these processes allows us to perceive and, above all, to predict the effects they have on the Earth's surface. Such predictions empower us to anticipate the effects of weather events in advance. For instance, at the present time we can foresee many consequences of climate change. Or, we can save lives by providing early warnings to the population before a hurricane hits.

One of the most studied and arguably the most important weather phenomenon is the process of rainfall. Its virtual modeling is a key factor in the estimation of various environmental changes caused by this phenomenon, e.g., variation of the amount of water remaining in a lake, formation of new rivers, or soil erosion process. The phenomenon of rainfall has been investigated from various scientific perspectives using diverse methods. The mathematical simulation of rainfall physical properties is a fairly complex process with many variables to consider.

Cellular automata [21] have been widely used in the last decades for environmental simulations such as modeling snow-cover dynamics [15], simulating land features dynamics [14], characterization of natural textures [13], modeling vegetation systems dynamics [1], detecting Vibrio cholerae by indirect measurement [16], simulating forest fire spread [17], modeling lava flows [20], understanding of urban growth [3], projecting population percentages infected by periodic plague [9], and simulating species competition and evolution [4].

This paper focus on RACA (RAinfall with Cellular Automata), an algorithm that simulates the complete rainfall process in DEM satellite images. RACA is to achieve several objectives, including the following three highlighted:

1. *Natural disaster prevention in urban areas with probability of flooding.* With the assistance of cellular automata, we can calculate the index ratio of flood disasters and streams in urban areas using the graphs of rainfall recorded historically in that region. In the visual results provided by the algorithm, we can clearly see which regions of the 3D satellite image will be covered by water after performing the corresponding simulations. Therefore, if an urbanized area appears to have a very high flood risk, the authorities can act to avoid loss of human life.
2. *Projection of future construction projects and urban planning.* We can identify safe regions for development around geographical areas at risk of flooding. Given all the graphs of rainfall recorded historically in a particular region, we can calculate the maximum height reached by the water at a given location in the 3D satellite image. Taking into account the results, we must avoid creating urban areas at altitudes lower than the height risk for certain areas with probability of flood in order to prevent further loss of life.
3. *Estimation of future water supply of a population.* Taking into account the rainfall charts for the type of climate of a particular area, we can make very rough estimates of the approximate cumulative water flow in a region that has a constructed wetland. RACA also considers the initial amount of water that the lake could have (initial conditions before simulation). In addition, we can calculate the most appropriate areas for the construction of a dam to optimize the process of accumulation of water with a minimum wall built.

There exist several previous studies related to water process phenomenon simulation with cellular automata: a model for soil erosion by water [7], an algorithm for simulation of surface flows in large plains [19], algorithms for drainage network extraction and rainfall data assimilation [6], and developments of a flood inundation model based on the cellular automata approach [8]. All these research works obtained realistic results because they use a large number of states and rules in the cellular automata. However, they have the drawback that it is necessary to know a priori a large number of parameters to configure the performance of cellular automata, and therefore, it is necessary to perform a preliminary study of the simulation region. In many cases there is no such information to perform a simulation, or it is hard to get it in terms of time. However, RACA uses very few states and rules of cellular automata in order to simplify the simulation process, implementing only the gravity and providing numerical and 3D quick view results on the final water level. Besides, RACA offers the possibility of making a preliminary field work of the study area in order to improve the basic results, adding water evaporation and groundwater flow to the simulation.

2. Cellular Automata

A cellular automaton is a mathematical model which consists of a set of cells usually distributed in a matrix form [12]. In recent years, cellular automata have become a powerful tool applied in remote sensing especially to implement any kind of simulation processes in satellite images. We include a short description of the model [10] here for the sake of completeness. From a mathematical point of view a cellular automaton is a set of six components, as shown in the following expression:

$$CA = (d, r, Q, \#, V, f) \tag{1}$$

where:

- $d|d > 0$: spatial dimension of the cellular automaton. The position of each cell is shown by a vector of Z^d. Given $d = 1$ it is a one-dimensional cellular automaton with cells position in Z; given $d = 2$ it is a bi-dimensional cellular automaton with cells position in $Z \times Z$; given $d = 3$ it is a tri-dimensional cellular automaton with cells position in $Z \times Z \times Z$, and so on.
- r: an index that shows the neighborhood dimension, that is, how many neighbours interact with each cell of the cellular automaton.
- Q: a set of states per cell. The set of states is finite, equal for all the cells of the cellular automaton and it can not be changed during the cellular automaton application process.
- $\#$: state called quiescent. This state shows inactivity in the cells of the cellular automaton and it is often used as the initial state of the cells.
- V: neighborhood vector which has r different elements from Z^d. The most common types of neighborhood in a cellular automaton are 4 neighbours (von Neumann neighborhood), 8 neighbours (Moore neighborhood) and 24 neighbours (extended Moore neighborhood). Figure

1 shows the most common types of neighborhood that we can find in a regular cellular automata. The neighborhood vector V is a subset of Z^d, as shown in the following expression:

$$V \subset (Z^d)^r \tag{2}$$

- f: cellular automaton transition function. It takes as input arguments the states of the current cell and its neighborhood, and returns a new state for the current cell. The transition function f uses a set of rules that specifies the changes of the cellular automaton cell states, and it is applied to each cell through a finite number of iterations, as shown in the following expression:

$$f : Q^{r+1} \rightarrow Q \tag{3}$$

$$q_i(t) = f(q_{i-r}(t-1), q_{i-r+1}(t-1), ..., q_{i+r}(t-1))$$

  where $q_i(t)$ is the state of the cell i at time t. The changes in cells states of the cellular automaton occur in discrete time form. In each iteration, the whole cells stored in Z^d are checked and the rules are applied through the transition function f to each cell taking into account the neighborhood V to change its state Q to its corresponding state.

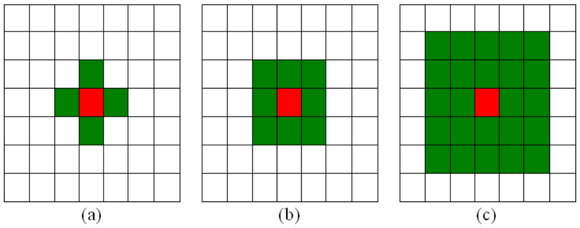


Figure 1: (a) Von Neumann neighborhood (b) Moore neighborhood (c) Extended Moore neighborhood.

When we work with satellite images, we usually consider each pixel of the image as a cell of a bi-dimensional cellular automaton ($d = 2$), we normally take the Moore neighborhood ($r = 8$), we assign to each cell a defined set of states to perform the corresponding simulation ($Q = q_1, q_2, ..., q_n$) and we apply the different rules ($R = r_1, r_2, ..., r_m$) in each iteration i through the transition function f. Therefore, cellular automata have an evolution process because the cells are always changing their states through the different iterations. From this point of view, cellular automata have become a powerful tool to simulate environmental processes in satellite images [10].

3. Simulating Rainfall with Cellular Automata

This paper presents the work of modeling and simulation of lakes and rivers flow, based on cellular automata, considering a volume of rainfall in a given time interval. To accomplish that goal, elevation data provided by the Shuttle Radar Topographic Mission (SRTM, NASA) were used. Each pixel of the DEM satellite image corresponds to a square shaped cell with size of one square meter.

Before the simulation process, RACA loads the DEM satellite image and the initial conditions, and takes as input arguments the samples of rainfall measured in liters per square meter of different times, which may be days, weeks, months, years, etc. The configuration of the cellular automaton is also loaded before the simulation process. Then, RACA applies the rules of its cellular automaton on the DEM satellite image performing a complete simulation for each of the samples of rainfall, using as many iterations as necessary for water to fall and accumulate in the lower levels of the 3D satellite image. Each cell of cellular automaton has two states, the original altitude of the corresponding DEM pixel ($state_1$) and the level reached by the water volume ($state_2$). The total height of a pixel will be the sum of both states. As boundary conditions, RACA simulates a vertical wall on the borders of the image. The results are presented not only at the end of the whole simulation process, but also after each iteration of the cellular automaton. Therefore, RACA has the following input-output components:

- *DEM satellite image loading.* $state_1$ of RACA is the terrain altitude obtained from the DEM satellite image. The value of this state in the cells does not vary along the simulation process.

- *Initial conditions loading.* Before applying the 3D simulation, RACA starts with the initial conditions: existing water level in the lake/river. These values are stored in $state_2$ of RACA and its values vary along the simulation process.

- *Precipitation graphs.* To carry out the simulation, the volume of precipitation must be measured in liters per square meter for each corresponding time-step (day, week, month, year, etc.).

- *Temperature and permeability.* RACA can take into account the air temperature and the soil permeability in order to simulate the water evaporation and the groundwater flow in the DEM satellite image. This parameters improve the simulation, and their use is optional. So, if we have field work of the study area, we can use them.

- *Configuration of cellular automata.* The transition function, neighborhood and maximum iterations number of RACA must be established before starting the simulation.

- *RACA: rainfall with cellular automata.* Each input argument (precipitation graph samples) that falls over each image pixel DEM will be used to calculate the volume of liters per square meter, and a full simulation of the water addition and outflows in the image will be established. The transition function f, through cellular automaton rules, will pass the corresponding volume of water from each cell to its neighbor with lower elevation height considering the total height $(state_1 + state_2)$, as the water goes down. When the water is reaching lower pixels that have neighboring cells with elevation greater than or equal to their own, the flow will increase on that surface: the pixel altitude $(state_1)$ plus volume of water level accumulated on it $(state_2)$. This step will be repeated in each iteration of the cellular automaton until all water of the image stagnates and stops moving between cells or until the maximum number of iterations. At that moment, the simulation of the current input argument ends, and the simulation of the next sample begins.

- *Results.* Finally, RACA shows the accumulated flow after each iteration of cellular automaton as well as the total water level after the complete process simulation. The quantity of water that exists on a given DEM image pixel is calculated by multiplying $state_2$ of the each cell (level of water above the pixel) by 1000 because, as each DEM image pixel has a spatial resolution of 1 m^2, then each three-dimensional cube of virtual representation has a resolution of 1 m^3, and each cubic meter has a volume of 1000 liters. To calculate the total flow of the geographical area, RACA adds the flow stored on each of the pixels of the DEM image. This flow is shown both numerically and in 3D virtual view. Figure 2 shows the general RACA architecture with all the input-output components.

The cellular automaton of RACA can be defined mathematically by the following expression:

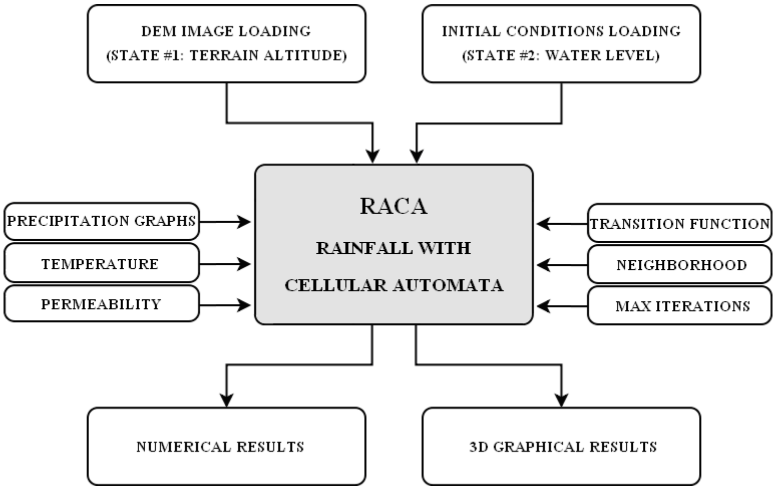


Figure 2: General RACA architecture.

$$RACA = (d, r, Q, \#, V, f) = \tag{4}$$
$$(3, \{4|8\}, [q_{terrain\_alt}, q_{water\_level}], [DEM\_alt, initial\_water], V, f)$$

where:

- $d = 3$: the spatial dimension of the RACA cellular automaton is 3, so cells are distributed in 3D form.
- $r = \{4|8\}$: the neighborhood dimension may consist of the von Neumann or Moore neighborhood.
- $Q = [q_{terrain\_alt}, q_{water\_level}]$: each cell has a set of two states, the terrain altitude and the water level.
- $\# = [DEM\_alt, initial\_water]$: the quiescent state, or initial value. This value is DEM_alt for $q_{terrain\_alt}$ and $initial\_water$ for q_{water_level}.
- V: the neighborhood vector is configurable to 4 or 8 surrounding neighbours of each one of the cells.
- f: the transition function applies the rules to each one of the cells along the different iterations in order to change their states taking into account the neighborhood chosen, as shown in the following expression:

$$f : Q^{\{4|8\}+1} \to Q \tag{5}$$

The transition function applies the following rules:

- Rule #1. If $water\_flow < q_{(i)water\_level}$:

  $q_{(i)water\_level}$ = $q_{(i)water\_level}$-$water\_flow$
  $q_{(j)water\_level}$ = $q_{(j)water\_level}$+$water\_flow$

- Rule #2. If $water\_flow \geq q_{(i)water\_level}$:

  $q_{(i)water\_level} = 0$
  $q_{(j)water\_level}$ = $q_{(j)water\_level}$+$q_{(i)water\_level}$

where $q_{(i)}$ is the current cell, $q_{(j)}$ is the neighboring cell with lower height (if there are several neighboring cells with the same height, we take a random one) and $water\_flow$ is defined by the following expression:

$$water\_flow = \tag{6}$$

$$\frac{(q_{(i)terrain\_alt}+q_{(i)water\_level})-(q_{(j)terrain\_alt}+q_{(j)water\_level})}{2}$$

RACA also uses the formula of Visentini to simulate the water evaporation in lakes and rivers, based on the annual average temperature of the study area. The empirical formulas proposed by Visentini are the following:

$$E = \begin{cases} 75t, & \text{for water level less than 200 meters above sea level} \\ 90t, & \text{for water level between 200 and 500 meters above sea level} \\ 90t + 300, & \text{for water level higher than 500 meters above sea level} \end{cases}$$

where E is the annual evaporation (in mm) and t is the annual average temperature in degrees Celsius.

Finally, RACA uses the soil permeability coefficient with the aim of estimating the hydraulic conductivity and simulate the groundwater flow, which is specific to the material through the water is flowing. Table 1 shows the typical ranges of permeability coefficient for different types of soil.

Figure 3 shows the water evaporation and groundwater flow processes that RACA has into account in order to simulate the decreasing of water level.

Type of soil	**Permeability** $k(cm^2)$
Well sorted gravel	$10^{-3}/10^{-4}$
Highly fractured rocks	$10^{-3}/10^{-6}$
Sand and gravel	$10^{-5}/10^{-7}$
Peat	$10^{-7}/10^{-8}$
Oil reservoir rocks	$10^{-7}/10^{-9}$
Very fine sand	$10^{-8}/10^{-11}$
Layered clay	$10^{-9}/10^{-11}$
Fresh sandstone	$10^{-10}/10^{-11}$
Fresh limestone or dolomite	$10^{-12}/10^{-13}$
Unweathered clay	$10^{-12}/10^{-15}$
Fresh granite	$10^{-14}/10^{-15}$

Table 1: Permeability coefficients for different types of soil.

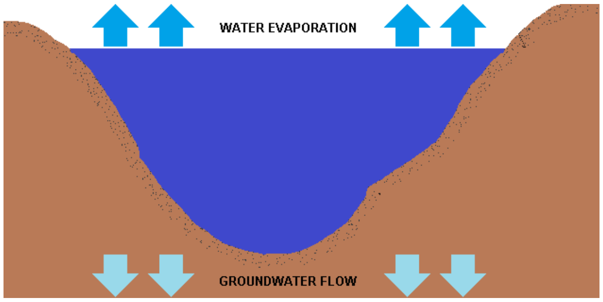


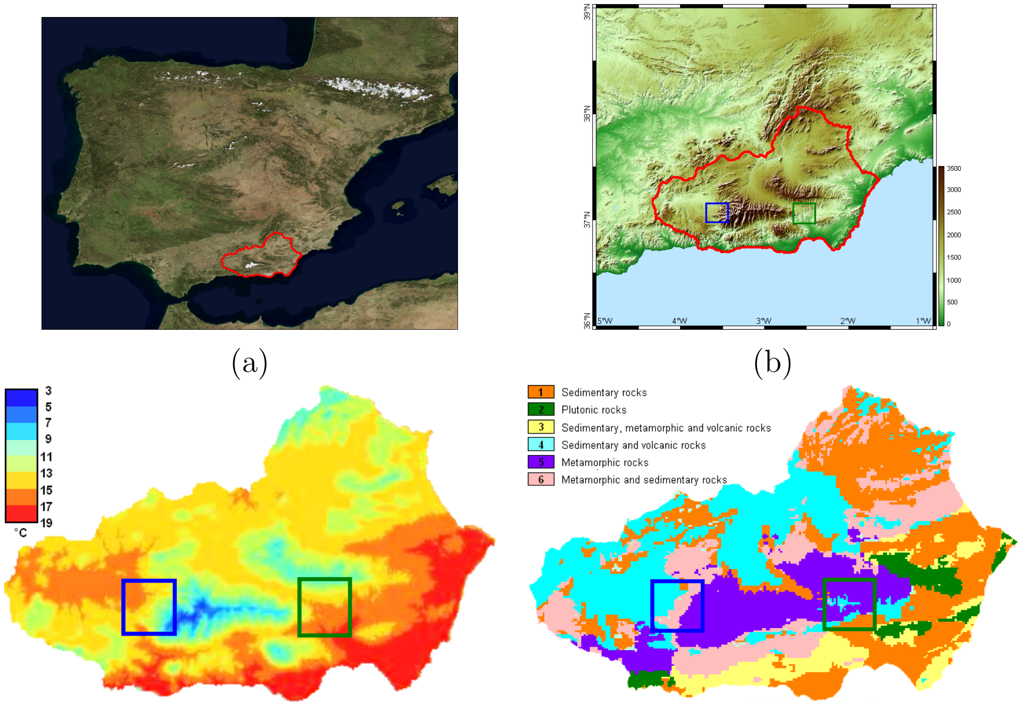
Figure 3: Water evaporation and groundwater flow.

4. Experimental Features

The DEM satellite images, provided by SRTM (NASA), correspond to two regions in Almería and Granada, southeast Spain: Sierra Nevada, a cold zone, with the geographical coordinates North 37 ° and West 3.5 °, and Desierto de Tabernas, an arid desert, with the geographical coordinates North 37 ° and West 2.5 °, as shown in Figure 4 (a)(b).

Each DEM image has 1200×1200 pixels, each pixel has a spatial resolution of 1 m^2, with an altitude ranging from 236 to 3460 m in Sierra Nevada, and ranging from 213 to 2602 m in Desierto de Tabernas. Temperature and soil in southeast Spain are extremely diverse. Figure 4 (c) shows the temperature areas in Almería and Granada with the study areas marked by the squares, and Figure 4 (d) shows the six soil areas. The permeability coefficient has been estimated for each area: 1 (10^{-13}), 2 (10^{-15}), 3 (10^{-12}), 4 (10^{-14}), 5 (10^{-10}) and 6 (10^{-11}), having into account the rock type composition of the soil. Specifically, Sierra Nevada is characterized by soil areas 1, 4, 5 and 6, and Desierto de Tabernas is characterized by soil areas 1, 4 and 5.

In order to compare the results provided by different types of climate, the rainfall simulations with RACA have been carried out using four different annual precipitation graphs (annual mean precipitation of last 30 years), each one corresponding to a different climate of Spain: sub-desert climate, continental climate, mediterranean climate and oceanic climate (source AEMET). Figure 5 shows the graphs of annual precipitation for the four different types of climates. As can be seen, the annual precipitation graphs are arranged in growing order, taking as reference the total number of liters per square meter per year. The first graph corresponds to the drier climate and the last graph corresponds to the wetter climate in which precipitation is recorded during all months of the year.



(a)

(b)

(c)

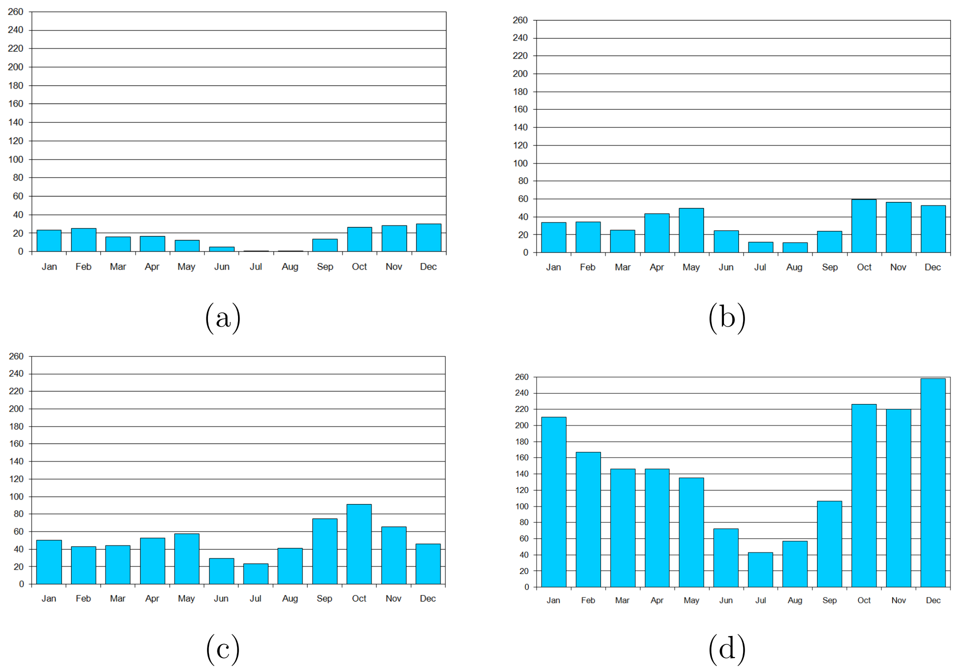
(d)

Figure 4: Experimental features of the DEM satellite images: (a) Almería and Granada provinces, situated in southeast Spain; (b) Sierra Nevada (blue square), Desierto de Tabernas (green); (c) temperature areas; (d) soil areas.

For each graph of annual precipitation, 12 simulations have been made, one for each month of the year. Therefore, the results of each DEM satellite image were obtained after making a total of 48 simulations. For each simulation, RACA used as many iterations of cellular automata as necessary until the water stopped moving. In the first few months, in which water levels are lower, the number of iterations of the cellular automata has been greater than in the subsequent months in which water levels are higher, consequently the water has less height to descend the mountain in those early months. To simplify the large number of results offered by all the simulations, we used as a reference only the annual final result of each type of climate after the simulation of all the months of the year.

5. Results and Conclusions

After performing the simulations without water evaporation and groundwater flow processes, the sub-desert climate records a total annual precipi-



(a)

(b)

(c)

(d)

Figure 5: Graphs of annual precipitation for four types of climate in Spain: (a) sub-desert climate, (b) continental climate, (c) Mediterranean climate, and (d) oceanic climate.

tation of 199 liters per square meter. The total cumulative flow in this type of climate is 286 million liters. With that amount of annual rainfall, the water reaches a maximum height level of 282 meters above sea level in Sierra Nevada and 265 meters above sea level in Desierto de Tabernas.

In the second place, the continental climate records a total annual precipitation of 427 liters per square meter. The total cumulative flow is 614 million liters, with the water reaching a maximum height of 298 meters above sea level in Sierra Nevada and 297 meters above sea level in Desierto de Tabernas.

In the third place, the mediterranean climate records a total annual precipitation of 621 liters per square meter. The total cumulative flow is 893 million liters, reaching a maximum water level height of 308 meters above sea level in Sierra Nevada and 316 meters above sea level in Desierto de Tabernas.

Finally, the oceanic climate records a total annual precipitation of 1787 liters per square meter. The cumulative flow is 2571 million liters, reaching the maximum water level height of 345 meters above sea level in Sierra Nevada and 379 meters above sea level in Desierto de Tabernas. Table 2 shows a summary of the numerical annual results obtained after conducting simulations of the four climate types in the Sierra Nevada and Desierto de Tabernas.

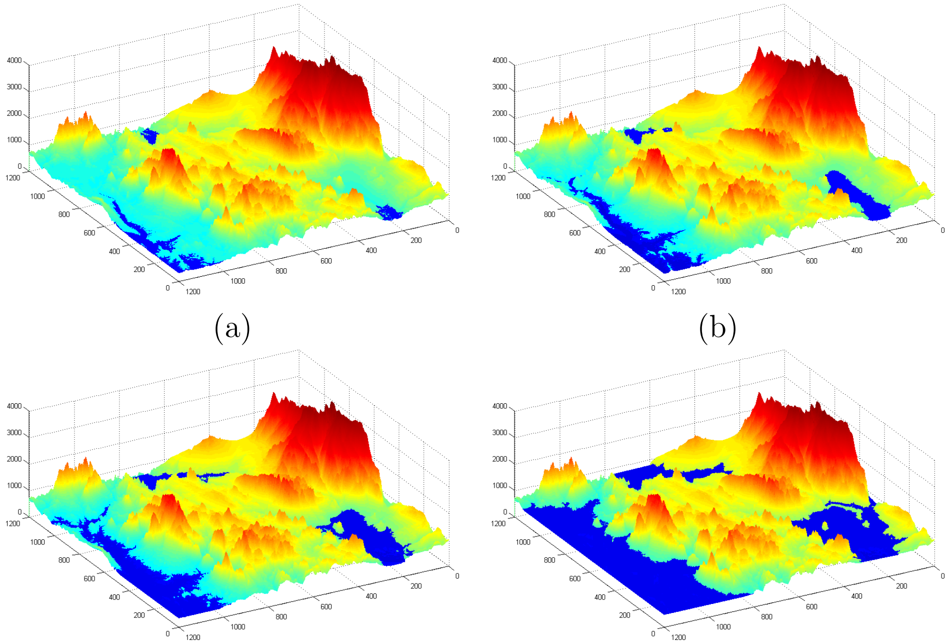
Type of weather	**ATP**	**TCF**	**MLW1**	**MLW2**
(a) Sub-desert climate	199 liters/m^2	286×10^6 liters	282 m	265 m
(b) Continental climate	427 liters/m^2	614×10^6 liters	298 m	297 m
(c) Mediterranean climate	621 liters/m^2	893×10^6 liters	308 m	316 m
(d) Oceanic climate	1787 liters/m^2	2571×10^6 liters	345 m	379 m

Table 2: Numerical results obtained in the four types of weather (where ATP=Annual Total Precipitation, TCF=Total Cumulative Flow, MLW1=Maximum Level of Water in Sierra Nevada and MLW2=Maximum Level of Water in Desierto de Tabernas).

After performing the simulations having into account the water evaporation and groundwater flow processes, the water disapears in the simulations of all the climates, except in the last one: oceanic climate. In this case, the total cumulative flow in Sierra Nevada is 618 million liters, reaching a maximum water level height of 299 meters above sea level, and the total cumulative flow in Desierto de Tabernas is 64 million liters, reaching a maximum water level height of 234 meters above sea level. The total cumulative flow of Sierra Nevada and Desierto de Tabernas changes because the temperature and soils are different in these two regions.

Figure 6 shows the 3D graphic results obtained in Sierra Nevada and Figure 7 in Desierto de Tabernas, without having into account the water evaporation and groundwater flow processes. Figure 8 shows the 3D graphic results obtained in Sierra Nevada and Desierto de Tabernas in oceanic climate, having into account the water evaporation and groundwater flow processes. In the 3D satellite images, the water is represented by dark blue, the lower areas with light colors (blue or light green), and the higher areas with darker colors (like orange or dark red). These 3D graphic results can be configured to provide the best possible view by changing the position of the camera through the keyboard, not only to view the image from the desired perspective but also to navigate freely across the surface of the DEM satellite image with the objective of seeing the most interesting details up close. This technique offers a multitude of practical applications, such as checking the flood risk in villages near wetlands depending on the amount of rainfall in a year in that geographic region.

Future simulation opportunities of cellular automata are diverse and pragmatic in remote sensing. For example, cellular automata could be used to simulate the process of soil erosion after the fall of rain for a long period of time to see how the passage of water modifies the appearance of the landscape.



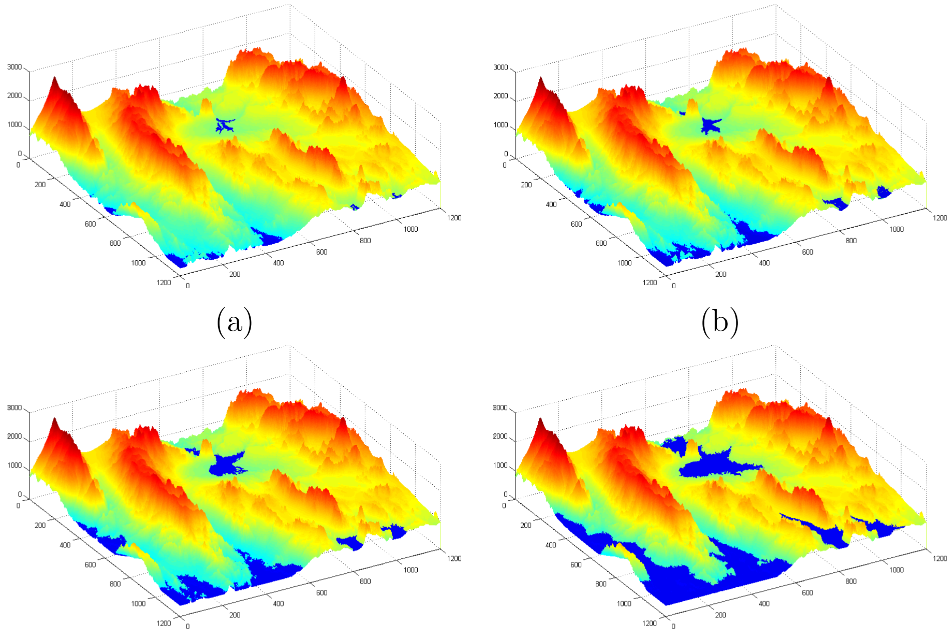
(a)

(b)

(c)

(d)

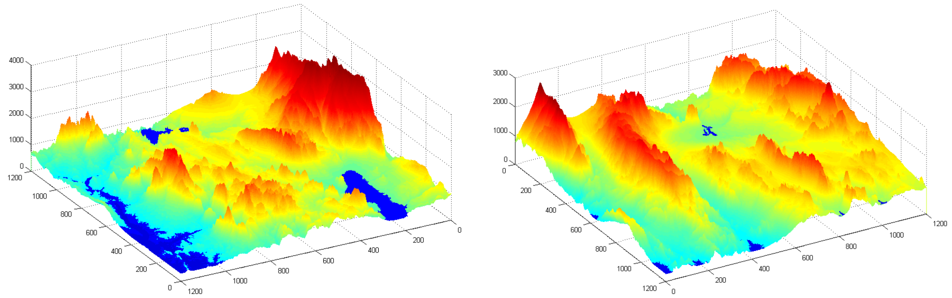
Figure 6: Graphic results obtained in Sierra Nevada: (a) sub-desert climate, (b) continental climate, (c) Mediterranean climate, and (d) oceanic climate.



(a) (b)

(c) (d)

Figure 7: Graphic results obtained in Desierto de Tabernas: (a) sub-desert climate, (b) continental climate, (c) Mediterranean climate, and (d) oceanic climate.



(a) (b)

Figure 8: Graphic results having into account the water evaporation and groundwater flow processes in oceanic climate: (a) Sierra Nevada, and (b) Desierto de Tabernas.

Acknowledgments

This work was funded by the EU ERDF and the Spanish Ministry of Economy and Competitiveness (MINECO) under Project TIN2013-41576-R, and the Andalusian Regional Government (Spain) under Project P10-TIC-6114. This work was also supported by the CEiA3 and CEIMAR consortiums. J. Z. Wang has been funded by the National Science Foundation under Grant No. 1027854.

[1] Balzter H, Braun P and Kühler W (1998) "Cellular automata models for vegetation dynamics", *Ecological Modelling*, 107: 113-125.

[2] Barret EC and Curtis LF (1999) "Introduction to environmental remote sensing", *Cheltenham Stanley Thornes Publishers Ltd.*

[3] Cheng J and Masser I (2002) "Cellular automata based temporal process understanding of urban growth", *ACRI 2002*, LNCS 2493: 325-336.

[4] Chopard B and Lagrava D (2006) "A cellular automata model for species competition and evolution", *ACRI 2006*, LNCS 4173: 277-286.

[5] Chuvieco E and Huete A (2010) "Fundamentals of satellite remote sensing", *CRC Press.*

[6] Coppola E, Tomassetti B, Mariotti L, Verdecchia M and Visconti G (2007) "Cellular automata algorithms for drainage network extraction and rainfall data assimilation", *Hydrological Sciences-Journal-des Sciences Hydrologiques*, 52(3): 579-592.

[7] D´ Ambrosio D, Di Gregorio S, Gabriele S and Gaudio R (2001) "A cellular automata model for soil erosion by water", *Physics and Chemistry of the Earth (B)*, 26(1): 33-39.

[8] Dottori F and Todini E (2011) "Developments of a flood inundation model based on the cellular automata approach: testing different methods to improve model performance", *Physics and Chemistry of the Earth 36*, 266-280.

[9] Dzwinel W (2004) "A cellular automata model of population infected by periodic plague", *ACRI 2004*, LNCS 3305: 464-473.

[10] Espinola M, Piedra-Fernandez JA, Ayala R, Iribarne L and Wang JZ (2015) "Contextual and hierarchical classification of satellite images based on cellular automata," *IEEE Transactions on Geoscience and Remote Sensing*, 53(2): 795-809.

[11] Guzman OF and Gómez ID (2007) "Fundamentos físicos de teledetección", *Instituto Geográfico Agustin Codazzi IGAC.*

[12] Kari J (2005) "Theory of cellular automata: a survey", *Theoretical Computer Science*, 334: 3-33.

[13] Leguizamón S, Espínola M, Ayala R, Iribarne L and Menenti M (2010) "Characterization of texture in images by using a cellular automata approach", CCIS 112(2): 522-533.

[14] Leguizamón S (2006) "Modelling land features dynamics by using cellular automata techniques", *Proceedings of the ISPR Technical Comission 7 Mid-Term Symposium "From pixels to Processes"*, 497-501.

[15] Leguizamón S (2005) "Simulation of snow-cover dynamics using the cellular automata approach", *Proc.of the 8th Int. Symposium on High Mountain Remote Sensing Cartography*, 87-91.

[16] Lobitz B, Beck L, Huq A, Woods B, Fuchs G, Faruque A and Colwell R (2000) "Climate and infectious disease: use of remote sensing for detection of Vibrio cholerae by indirect measurement", *Proc. National Academic Sci. USA*, 97(4): 1438-43.

[17] Muzy A, Innocenti E, Aiello A, Santucci JF, Santonio PA and Hill D (2005) "Modelling and simulation of ecological propagation processes: application to fire spread", *Environmental Modelling and Software*, 20: 827-842.

[18] Rees WG (2001) "Physical principles of remote sensing", *Cambridge University Press 2nd Edition.*

[19] Rinaldi P, Dalponte D, Vénere M and Clausse A (2007) "Cellular automata algorithm for simulation of surface flows in large plains", *Simulation Modelling Practice and Theory 15*, 315-327.

[20] Spataro W, D Ambrosio D, Rongo R and Trunfio GA (2004) "An evolutionary approach for modelling lava flows through cellular automata", *ACRI 2004*, LNCS 3305: 725-734.

[21] Wolfram S (2002) "A new kind of science", *Wolfram Media Inc.*