



Article

# A Head/Tail Breaks-Based Method for Efficiently Estimating the Absolute Boltzmann Entropy of Numerical Raster Data

Hong Zhang  and Zhiwei Wu \* 

Faculty of Geosciences & Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China; zhangh2011@swjtu.edu.cn

\* Correspondence: WuZhiWei@my.swjtu.edu.cn

Received: 10 December 2019; Accepted: 6 February 2020; Published: 7 February 2020



**Abstract:** Shannon entropy is the most popular method for quantifying information in a system. However, Shannon entropy is considered incapable of quantifying spatial data, such as raster data, hence it has not been applied to such datasets. Recently, a method for calculating the Boltzmann entropy of numerical raster data was proposed, but it is not efficient as it involves a series of numerical processes. We aimed to improve the computational efficiency of this method by borrowing the idea of head and tail breaks. This paper relaxed the condition of head and tail breaks and classified data with a heavy-tailed distribution. The average of the data values in a given class was regarded as its representative value, and this was substituted into a linear function to obtain the full expression of the relationship between classification level and Boltzmann entropy. The function was used to estimate the absolute Boltzmann entropy of the data. Our experimental results show that the proposed method is both practical and efficient; computation time was reduced to about 1% of the original method when dealing with eight  $600 \times 600$  pixel digital elevation models.

**Keywords:** head/tail breaks; numerical raster data; absolute Boltzmann entropy; computational efficiency

## 1. Introduction

Raster data representation offers a number of important advantages over other options, hence the technology predominates in satellite imagery, digital elevation modeling, landscape gradient mapping, and other applications. The advantages of raster data modeling include its simple data structure, the ease with which location-specific data may be collected, and its suitability for representing continuous surfaces [1]. The quantification of information of raster datasets has long been a challenging work and have proven useful in many applications. For example, information held in raster form has been used to evaluate the performance of image fusion [2–4], and it is regarded as an essential reference for band selection in hyperspectral imaging [5,6].

Information production may be quantified through Shannon entropy [7], which is commonly used in many domains, such as computer graphics [3] and landscape ecology [8,9]. However, the application of Shannon entropy to spatial data especially for raster data has been questioned because it is ill-suited to quantifying spatial information that describes the spatial distribution of raster data [10].

Two solutions have been proposed to address this problem: the Shannon entropy model can be improved [11–13] or Boltzmann entropy can be used in its place [14–18]. Boltzmann entropy is the measure of the degree of system disorder, and, in theory, it is capable of quantifying the spatial information of raster data [19]. The first method to calculate the Boltzmann entropy of numerical raster data was proposed recently [18]. This approach defines a proper macrostate and then determines

the number of microstates. By using absolute Boltzmann entropy, which has zero entropy as its point of reference, the proposed method is able to compare different raster data in an absolute sense. The experimental results of Gao et al. [18] showed that their method can capture the spatial information held within a raster data structure. However, this method is not efficient because it includes a series of processes that are computationally intensive and time-consuming.

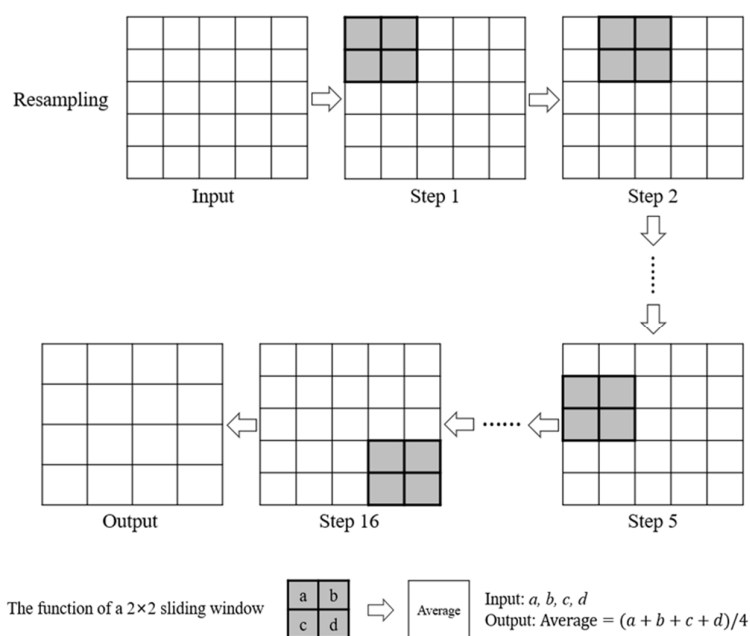
This study aims to develop a method that can estimate the absolute Boltzmann entropy of numerical raster data in a way that is efficient. To do so, we adopted a classification method for a heavy-tailed distribution with a scaling pattern, estimated the relative Boltzmann entropy of each class, and computed the absolute Boltzmann entropy based on these estimates. We tested the performance of this approach with several experiments using numerical raster data.

## 2. The Absolute Boltzmann Entropy of Numerical Raster Data: Computation and Problem

A method for computing the absolute Boltzmann entropy of a landscape gradient is described in detail in Gao et al. [18]. First, a proper set of microstates must be defined, which can be done by resampling the original landscape gradient with a sliding window of  $2 \times 2$  pixels, as shown in Figure 1. The number of microstates for each sliding window in the original landscape was uniquely determined based on the related macrostate. The relative Boltzmann entropy ( $S_R$ ) of the original landscape can be computed by the Boltzmann entropy equation as

$$S_R = K_B \log\left(\prod_{i=1}^m W_i\right), \quad (1)$$

where  $K_B$  denotes the Boltzmann constant ( $1.38 \times 10^{-23}$ ),  $W_i$  is the number of possible microstates for  $i^{\text{th}}$  sliding window, and  $m$  is the number of sliding windows. The base of the logarithm is usually set to 2, 10, or the natural log base,  $e$ , depending on the context.

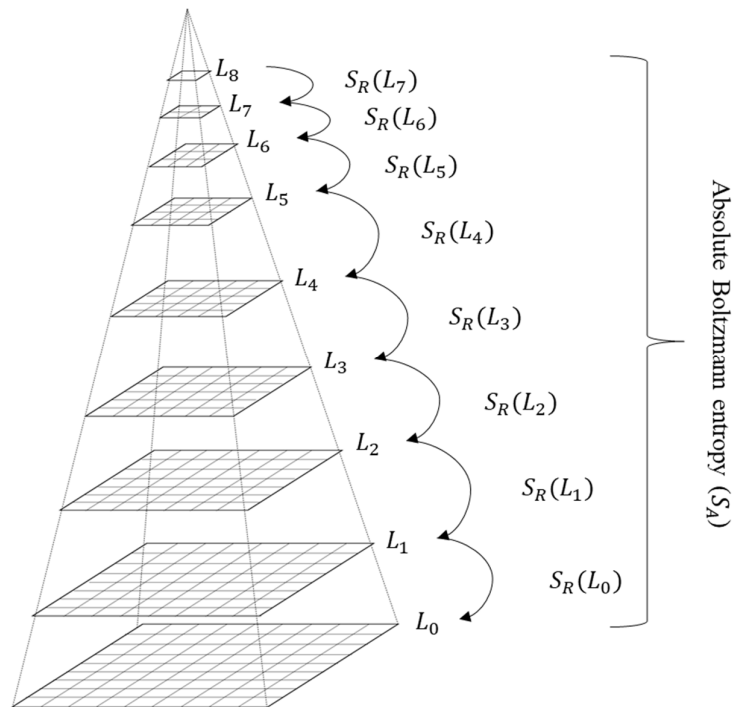


**Figure 1.** A diagram of the resampling technique used to generate a macrostate (output) for a landscape gradient (input; modified from Gao et al. [18]).

A given landscape gradient can be represented with a hierarchy that spans from the most detailed level (i.e., the original landscape) to the most abstract level (i.e., a single pixel), as shown in Figure 2. The sum of all  $S_R$  values is the absolute Boltzmann entropy ( $S_A$ ) of the landscape gradients, as represented in the equation

$$S_A = \sum_{j=1}^n S_R(L_j) \quad (2)$$

where  $L_j$  denotes the  $j^{\text{th}}$  level in the hierarchy,  $S_R(L_j)$  is the relative Boltzmann entropy of  $L_j$ , and  $m$  is the total number of levels. The base of the logarithm is usually set to 2, 10, or  $e$ .



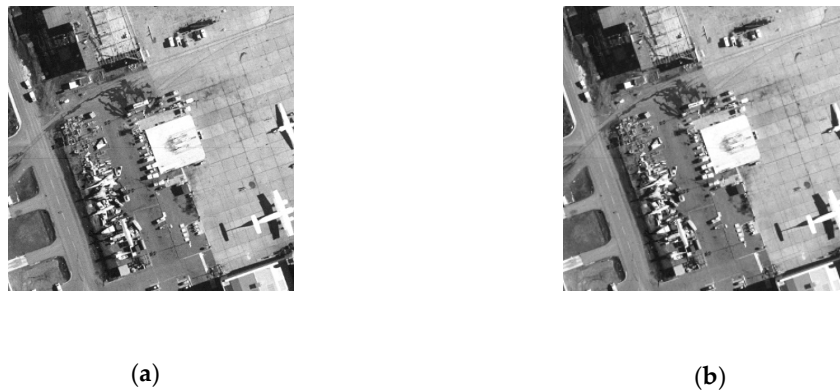
**Figure 2.** A visual representation of summing the relative Boltzmann entropies ( $S_R$ ) of all levels to get the absolute Boltzmann entropy ( $S_A$ ) of a landscape gradient (modified from Gao et al. [18]).

The method used by Gao et al. quantifies the spatial information of raster data, such as a landscape gradient. However, the iterative process for determining the number of microstates for each sliding window is slow and computationally demanding. Moreover, as the dimensions of the original landscape gradient increases, the number of sliding windows increases exponentially. For example, as the size of the original landscape expands from  $10 \times 10$  pixels to  $100 \times 100$  pixels, the number of sliding windows increases from 81 ( $9^2$ ) to 9801 ( $99^2$ ).

Computing the absolute Boltzmann entropy is also a process of iteration because each level is derived from the level below it through resampling (e.g.,  $L_1$  is derived from  $L_0$ ). Each pixel in  $L_1$  can be generated from four related pixels in  $L_0$  by this method. For example, the average of the four pixels in  $L_0$  is the value of the corresponding pixel in  $L_1$ . Further, the value of all pixels in  $L_1$  can be obtained. By using “resample” tools with bilinear interpolation option in ArcGIS software, we changed the image shown in Figure 3a from  $512 \times 512$  pixels to  $256 \times 256$  pixels (Figure 3b).

We established a baseline comparison by computing the absolute Boltzmann entropy for two images of different sizes (Figure 3) in the same operating environment—MATLAB running in 64-bit Windows 10 with an Intel Core i7-8750H CPU (2.20 GHz, 12.00 GB RAM). We recorded the time taken to compute the number of microstates, as well as the total computation time. The results of the baseline computation are shown in Table 1. We found that the time required to compute the number of microstates accounted for a large proportion of the total computation time. In addition, the larger

image ( $512 \times 512$  pixels, 1218.5s) required about seven times as much computation time as the smaller image ( $256 \times 256$  pixels, 173.5s), even though its area was only four times larger.



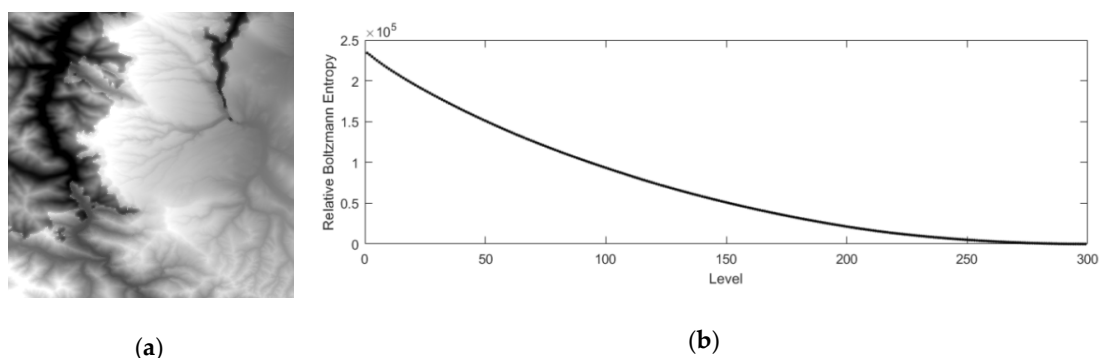
**Figure 3.** Two images with different resolutions: (a)  $512 \times 512$  pixels and (b)  $256 \times 256$  pixels.

**Table 1.** The time taken to compute the microstates and to calculate the absolute Boltzmann entropy for two images of different sizes, showing the proportion of time spent on the microstates.

Image	Time of Computing the Number of Microstates (s)	Total Time (s)	Time of Computing the Number of Microstates/Total Time (%)
(a)	1002.5	1218.5	82.3
(b)	148.1	173.5	85.4

### 3. A Strategy for Efficiently Estimating the Absolute Boltzmann Entropy

Since the sum of all relative Boltzmann entropies is the absolute Boltzmann entropy, we simplified our approach by assuming that the relative Boltzmann entropy of each level could be plotted against its hierarchy level to generate a curve. The value of the variable *Level* stands for the level in the hierarchy of a landscape gradient. An example of this process is shown in Figure 4. By extrapolating this curve, the complex process of computing the absolute Boltzmann entropy was changed into the relatively easy process of adding the values of all points on the curve.



**Figure 4.** An illustration of (a) a Digital Elevation Model (DEM) obtained from the Geospatial Data Cloud site (<http://www.gscloud.cn>) with a size of  $300 \times 300$  pixels, and (b) a curve established between the hierarchy level of the DEM and its relative Boltzmann entropy.

The tail of the curve always asymptotically approaches the x-axis in the positive direction, and most points have small relative Boltzmann entropy values. This curve is considered to have a heavy-tailed distribution, which means that it is significantly right-skewed and can be described as having a “scaling” or “hierarchical” pattern, with “far more smaller things than larger ones” [20].

By assuming that the shape of the curve, and therefore its function, can be estimated from several points, we were able to predict the values of the other points and use these to estimate the absolute Boltzmann entropy of the numerical raster data. The points that we used to estimate the shape of the curve were close to the y-axis. This was the result of a fundamental limitation of the iterative process, which always starts with the original gradient and proceeds in sequence. These low-level points have a high impact on the trajectory of the curve and on the sum of the entropies, which makes them more critical than those points with smaller values. It was reasonable to estimate the shape of the curve from these critical points. This solution was developed to avoid having to compute the relative Boltzmann entropies at all levels, which should improve the computational efficiency of the entire process.

However, estimating the function of a curve based on several points is challenging. As the shape of the curve becomes more complex, more data is required to represent it accurately [21]. When the available data is insufficient, breaking the curve into stepwise classifications is the best way to reduce its complexity. For example, MacDonald et al. [22] used three different classes to simplify the curve, a lower tail, a plateau, and an upper tail. We identified an appropriate classification method to break up the curve, with the additional objective of extracting the essential points with larger relative Boltzmann entropy values. A representative value was chosen to represent all points in a given class, and several representative values were substituted into a function to obtain the full expression of that function, which was then used to estimate the relative Boltzmann entropies of other classes. Finally, the absolute Boltzmann entropy of the numerical raster data was computed based on these estimated relative Boltzmann entropies.

Following this line of thought, this paper proposes a strategy for efficiently estimating the absolute Boltzmann entropy as follows:

- Analyze the data distribution and adopt a suitable classification method.
- Estimate a function through a representative value of the most essential class and obtain the estimated relative Boltzmann entropies of other classes through this function.
- Compute the absolute Boltzmann entropy of the numerical raster data based on the estimated relative Boltzmann entropies.

#### 4. The Calculation of Absolute Boltzmann Entropy Based on Head/Tail Breaks

In this section, the approach of head/tail breaking has been used for estimating the relative Boltzmann entropies of each class for a given numerical raster dataset. The total absolute Boltzmann entropy can be obtained by summing up the estimation values of relative Boltzmann entropy of each class.

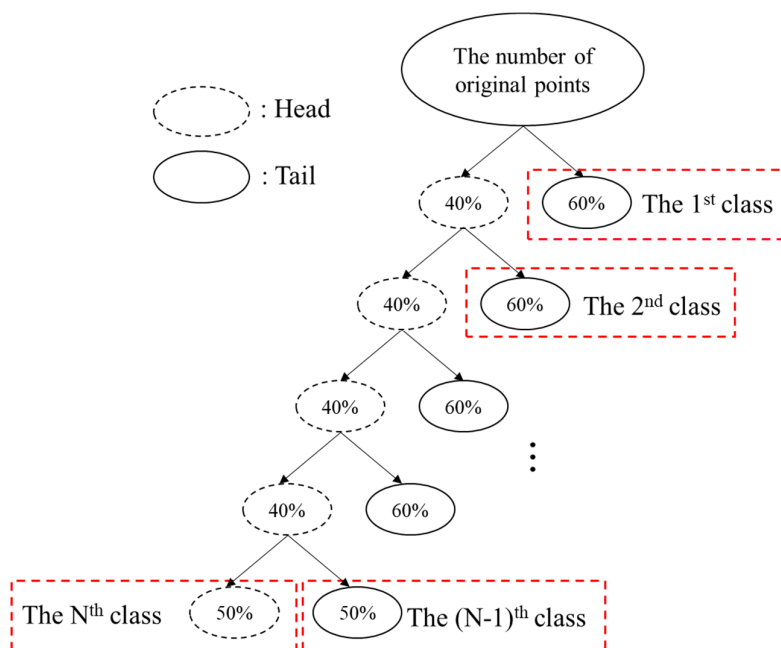
##### 4.1. Adopting a Classification Method for a Heavy-Tailed Distribution with a Scaling Pattern

Good classification results can reflect the patterns of data. The results of classification will be entirely dependent on the classification method that is used to divide the curve, so it is crucial to adopt an appropriate classification method.

A classification method for data with a heavy-tailed distribution has been proposed recently [20]. The method sorts the data into two imbalanced parts based on the arithmetic mean—a head (higher than the mean) and a tail (lower than the mean). Given that this imbalanced structure may repeat in the head, these points are further divided into two parts until the points remaining in the head do not follow a heavy-tailed distribution. The final division of data points between the head and tail is generally about 20% and 80%, respectively [23]. However, “this condition can be relaxed for many geographic features, such as 50 percent or even more” [24].

To estimate the number of points contained in the head and tail of each class, this paper used this idea of head and tail breaks and set a target of 40% of points in the head and 60% in the tail. For example, for a numerical raster that was  $100 \times 80$  pixels, we obtained 80 levels through resampling, but the 80<sup>th</sup> level consisted of only one unit, which did not meet the calculation conditions established by

Gao et al. Therefore, the final number of points on the curve was 79, of which 31 were sorted into the head (decimals were rounded down). This division was repeated until the number of points in the head equaled the number of points in the tail, or the number of points in the head reached 1. Since the heavy-tailed distribution considered in this paper had a scaling pattern, it contained a small number of high relative Boltzmann entropies. The tail from each step was considered a class, as was the head in the last step, as shown in Figure 5.



**Figure 5.** The process of classification, which uses the concept of head and tail breaks.

The classification method used in this study not only classifies all points into one of several classes, but it reflects the patterns of data with a heavy-tailed distribution. More importantly, it extracts the most important points—those with the largest relative Boltzmann entropy values.

#### 4.2. Estimating the Relative Boltzmann Entropies of Each Class

Since the range of values in a given same class is small, this paper chose to use the average of the points in each class as its relative Boltzmann entropy value.

The points in each classification were derived from the head of the upper classification, and the points contained in the last classification should have the highest relative Boltzmann entropies. We computed the relative Boltzmann entropies of these points, which was expedited through the use of the method borrowed from Gao et al. [18]. The average of these accurate relative Boltzmann entropies is the relative Boltzmann entropy of the last class. In the same way, the accurate relative Boltzmann entropies of the  $(N - 1)^{th}$  class and the  $(N - 2)^{th}$  class can be computed, in theory. However, if we were to compute the accurate relative Boltzmann entropy of every class, then we would not have solved the inefficiency problem.

Instead, we used the accurate relative Boltzmann entropies of several classes to estimate the relative Boltzmann entropies of other classes. This is valid because these relative Boltzmann entropies can be described approximately through a linear function. This paper chose the linear function  $y = a \times x$ . Recall that the first class through the  $(N - 1)^{th}$  class all belong to the tail, while the last class belongs to the head. We substituted the accurate relative Boltzmann entropy of the  $(N - 1)^{th}$  class into the linear function (i.e., the independent variable is  $[N - 1]$ , and the dependent variable is the accurate relative Boltzmann entropy of the  $[N - 1]^{th}$  class). We inversely solved the parameter  $\alpha$  and derived

the expression of the function. Then, we computed the relative Boltzmann entropies of the remaining classes, which belonged to the tails.

#### 4.3. Computation of the Absolute Boltzmann Entropy Based on Estimates

Thus far, we have obtained the number of classes, the number of points contained in each class, and the relative Boltzmann entropies of each class. The absolute Boltzmann entropy of the numerical raster dataset of interest can be computed as

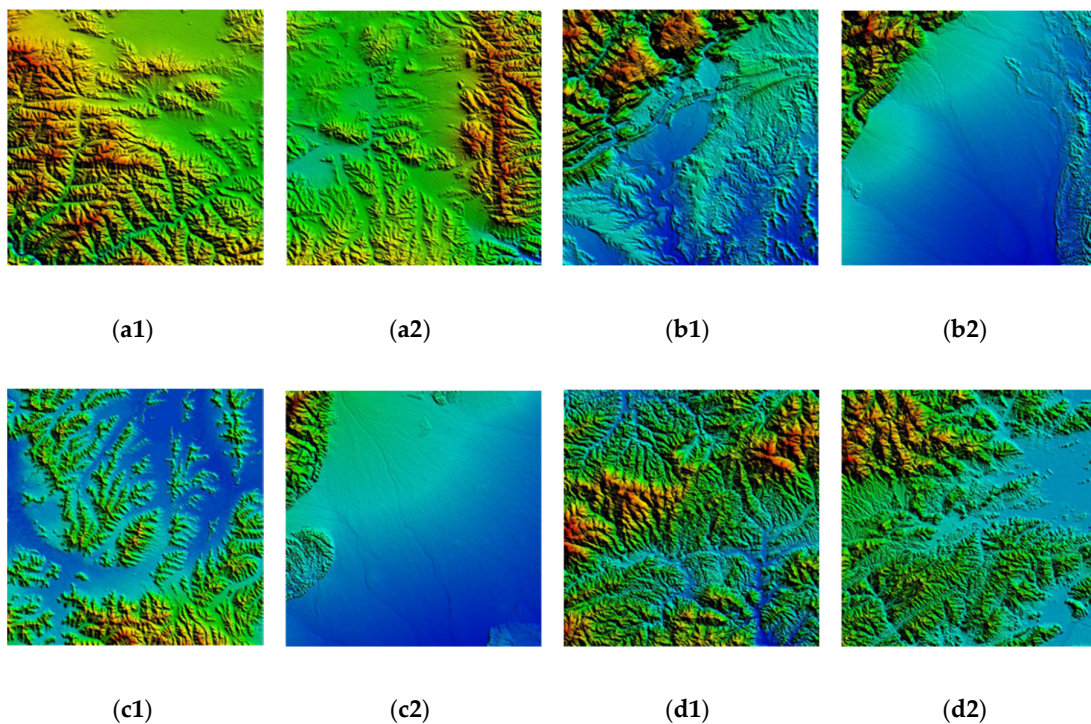
$$\overline{S}_A = \sum_{c=1}^k \overline{S}_R(c) \times n_c \quad (3)$$

where  $\overline{S}_A$  denotes the estimated absolute Boltzmann entropy of the numerical raster data,  $\overline{S}_R(c)$  is the estimated relative Boltzmann entropy of class  $c$ ,  $n_c$  is the number of points in class  $c$ , and  $k$  is the number of classes.

For example,  $\overline{S}_R(1)$  and  $\overline{S}_R(2)$  are the averages of all points in the  $N^{\text{th}}$  class and the  $(N - 1)^{\text{th}}$  class, respectively. They can be computed precisely through the Gao et al. method [18].

### 5. Experimental Testing

We conducted several tests of the proposed method. As shown in Figure 6, the experimental data, i.e., four pairs of digital elevation models (DEMs), was the same as that used by Gao et al. [18]. The size of each DEM was  $600 \times 600$  pixels, with each pixel representing the elevation in an area. The DEMs in Figure 6b2,c2 are smoother than the other DEMs, indicating a relatively low degree of heterogeneity. They contain fewer impossible microstates, which lowers their absolute Boltzmann entropy. To better analyze our results, we compared the results of these experiments with those of Gao et al. [18], as shown in Table 2.



**Figure 6.** Eight digital elevation models, paired by letter, taken from the experimental data of Gao et al. [18].

**Table 2.** The absolute Boltzmann entropy and its computation time, as calculated by the traditional method used by Gao et al. [18] and our proposed method.

DEMs	Computation Time (s)		The Absolute Boltzmann Entropy		
	Gao et al.'s Research	The Proposed Method	Accurate Values	Estimated Values	Relative Error (%)
a1	2842.2	6.9	$1.2 \times 10^8$	$1.0 \times 10^8$	16.7
a2	2271.1	6.7	$1.0 \times 10^8$	$1.0 \times 10^8$	0.0
b1	2807.9	6.8	$9.3 \times 10^7$	$10.0 \times 10^7$	7.5
b2	1482.9	7.8	$4.3 \times 10^7$	$6.1 \times 10^7$	41.9
c1	2132.8	7.2	$8.5 \times 10^7$	$8.1 \times 10^7$	4.7
c2	1294.9	6.7	$3.1 \times 10^7$	$4.9 \times 10^7$	58.1
d1	3717.7	6.8	$1.4 \times 10^8$	$1.2 \times 10^8$	14.3
d2	2924.9	6.9	$1.1 \times 10^8$	$1.1 \times 10^8$	0.0

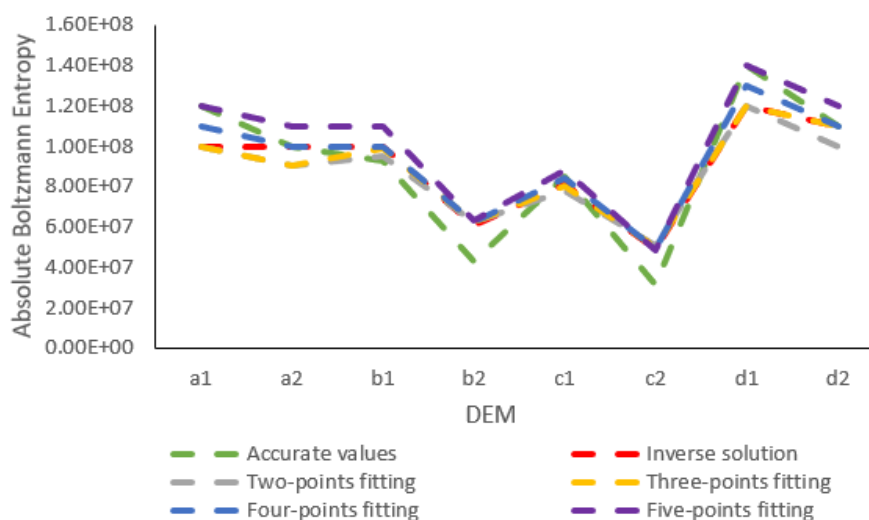
Note:  $Relative\ error = \frac{|Accurate\ values - Estimated\ values|}{Accurate\ values} \times 100\%$

Table 2 reveals that the use of our proposed method dramatically reduced the computation time by between 99.5% (from 1482.9 to 7.8 s) and 99.8% (from 2842.2 to 6.9 s). However, their relative errors were unstable with the smallest at 0% and the largest reaching 58.1%.

## 6. Discussion

In the process of estimating the relative Boltzmann entropy of each class, we chose a linear function ( $y = a \times x$ ) and substituted the accurate relative Boltzmann entropy of the  $(N - 1)^{th}$  class into the linear function to inversely solve the parameter  $\alpha$  (inverse solution). This allowed us to obtain the expression of the function. We tried other function models, such as quadratic functions, power law functions, and exponential functions, but linear functions yielded results closest to the actual values.

We used the accurate relative Boltzmann entropies of several classes to generate a fitted linear function ( $y = a \times x$ ), which has been used to compute the relative Boltzmann entropies of the remaining classes. To identify the most efficient number of points (classes) to use, we modeled the entropies based on two, three, four, and five points respectively. Supposing  $x$  is the number of points, the classes involved will range from the  $N^{th}$  class to the  $(N - (x - 1))^{th}$  class. When applied to the experimental DEM, the results of absolute Boltzmann entropies obtained by models with varying number of points are shown in Figure 7, with the relative errors tabulated in Table 3 and the computation times displayed in Figure 8 and Table 4.



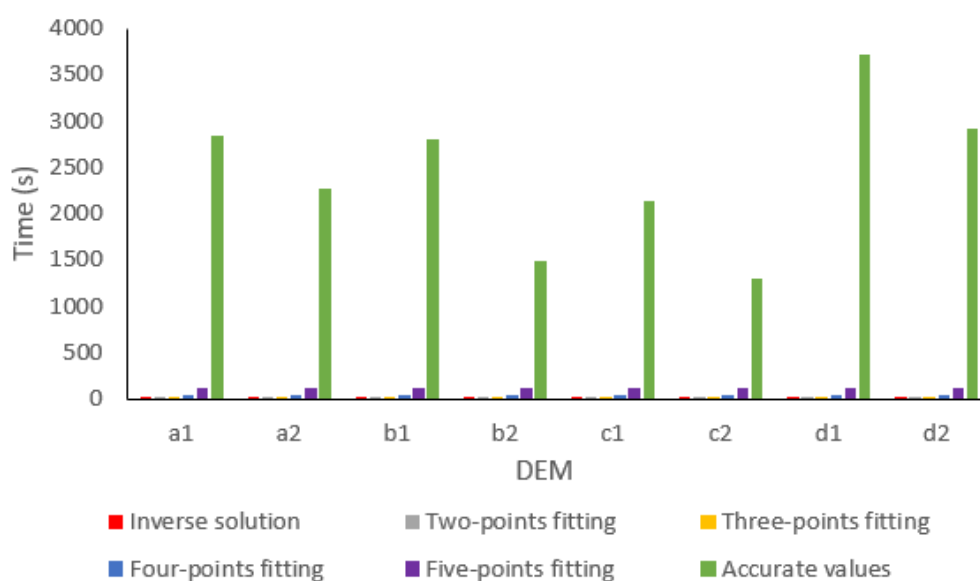
**Figure 7.** A comparison of the absolute Boltzmann entropies of four pairs of DEMs from accurate values and estimates produced by five methods.



**Table 3.** The relative errors (%) of the absolute Boltzmann entropy calculated using different methods for eight DEMs.

Method	a1	a2	b1	b2	c1	c2	d1	d2
Inverse solution	16.7	0.0	7.5	41.9	4.7	58.1	14.3	0.0
Two-points fitting	16.7	10.0	2.2	44.2	8.2	64.5	14.3	9.1
Three-points fitting	16.7	10.0	6.5	44.2	5.9	61.3	14.3	0.0
Four-points fitting	8.3	0.0	7.5	44.2	1.2	58.1	7.1	0.0
Five-points fitting	0.0	10.0	18.3	44.2	3.5	54.8	7.1	0.0

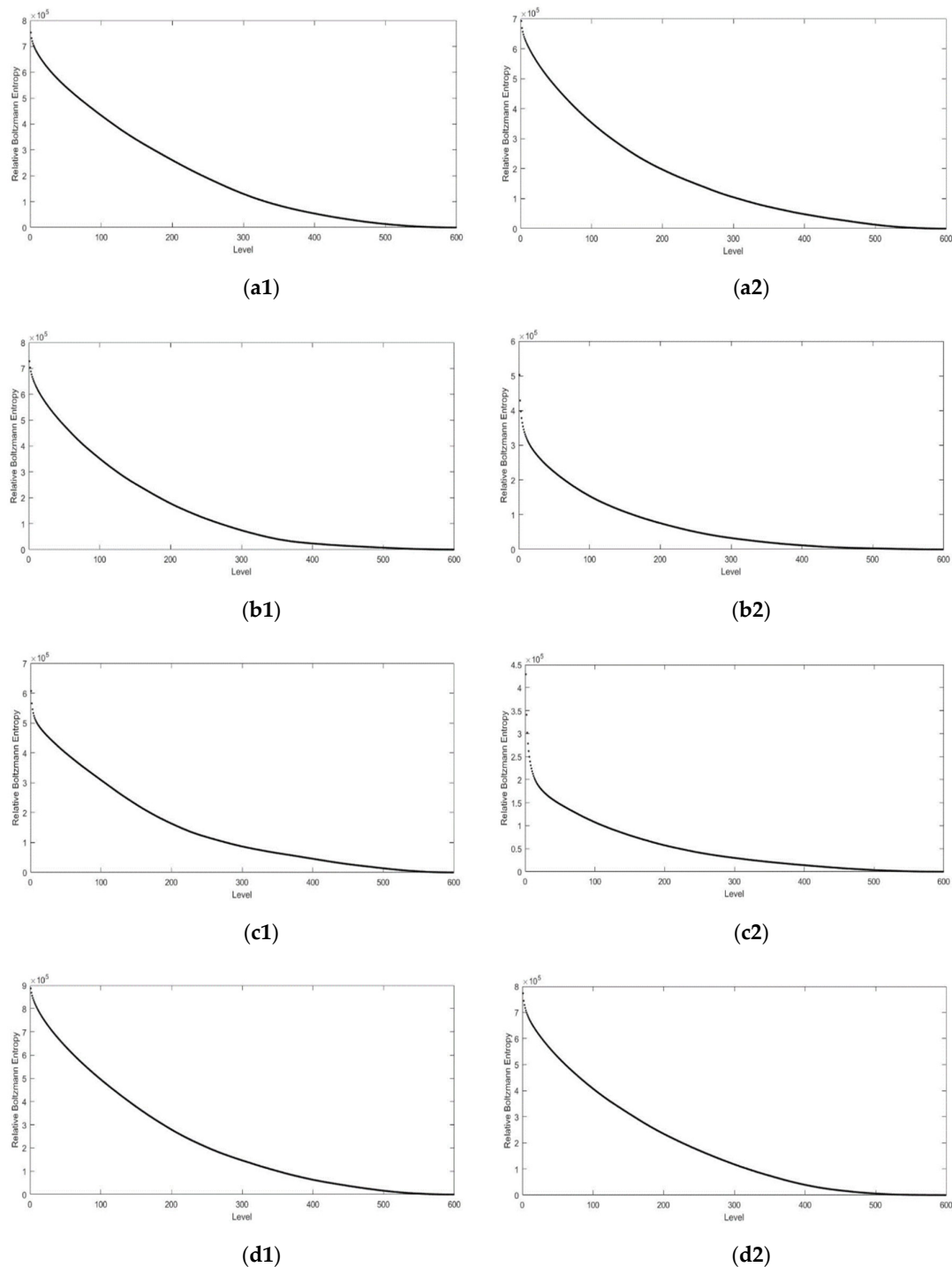
Note:  $Relative\ error = \frac{|Accurate\ values - Estimated\ values|}{Accurate\ values} \times 100\%$

**Figure 8.** A comparison of the computation time required for four pairs of DEMs from accurate values and estimates produced by five methods.**Table 4.** The computation times (s) of the absolute Boltzmann entropy calculated using different methods for eight DEMs.

Method	a1	a2	b1	b2	c1	c2	d1	d2
Inverse solution	6.9	6.7	6.8	7.8	7.2	6.7	6.8	6.9
Two-points fitting	7.0	6.8	7.1	7.4	7.0	6.7	6.9	7.0
Three-points fitting	19.8	19.3	19.3	19.7	19.1	19.2	19.4	19.7
Four-points fitting	47.9	51.6	48.7	47.1	48.2	49.8	48.2	49.6
Five-points fitting	123.2	122.0	115.6	121.5	116.2	117.4	114.2	120.8

We found that all models yielded estimates similar to the accurate values, and we were unable to identify a consistent relationship between the estimates of different methods for a given DEM. As more points were used to fit the linear function, the computation time increased. The relative errors using a fitting with four points were minimal and the computation time was acceptable. Hence, we argue that four-point fitting was the most suitable solution. As the size of a numerical raster becomes larger, it may not prove adequate to substitute only one point into the linear function to obtain the final curve. For any raster, there is an optimal number of points used to generate a fitted linear function that maximizes accuracy and also of high calculation efficiency.

Table 3 reveals the systematic failure of the models to estimate the absolute Boltzmann entropies of DEM b2 and DEM c2. In order to identify the possible reasons for this consistently high error, we plotted level against relative Boltzmann entropy for all eight DEMs, as shown in Figure 9.

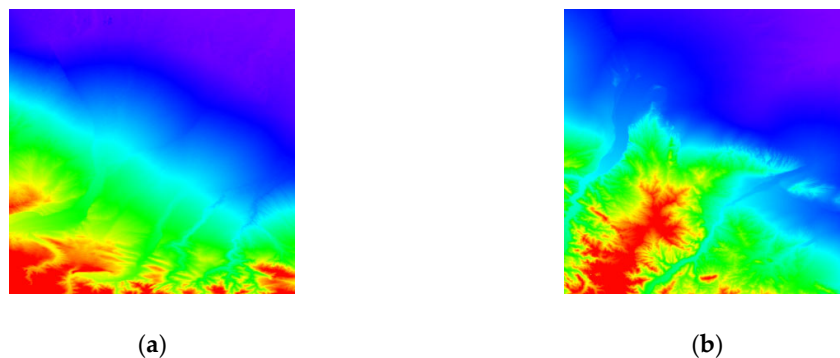


**Figure 9.** Plots of level against relative Boltzmann entropy for each of the eight DEMs. (a1–b1), (c1) and (d1–d2) Six plots of level against relative Boltzmann entropy for DEMs with high heterogeneity; (b2) and (c2) Two plots of level against relative Boltzmann entropy for DEMs with low heterogeneity.

The negative slopes of DEM b2 and DEM c2 are steeper than those of the other DEMs. From Figure 6, it is clear that DEM b2 and DEM c2 began with smoother topography than the other DEMs, indicating low heterogeneity. This may be exacerbated by the process of reducing the resolution through the resampling technique, which tends to decrease heterogeneity even further for those images with lower heterogeneity. In other words, with the decrease of spatial heterogeneity, the abundance of impossible microstates diminishes, which led the relative Boltzmann entropy of each level to shrink.

Since the heterogeneity of DEM b2 and DEM c2 started off small, the relative Boltzmann entropy of each level decreased faster than for the other DEMs, and the number of points with small values in the tail was more than for other DEMs. For DEM b2 and DEM c2, the average of all points in the first class were meager, but the estimations of the first class through a linear function will be much higher than the accurate average.

Gao and Li [17] found that their resampling-based method paid more attention to the central locations than to the edge ones when computing the absolute Boltzmann entropy of a landscape gradient, which is not consistent with thermodynamics. To address this problem, they proposed an aggregation-based method for computing the absolute Boltzmann entropy. This method is similar to the resampling-based method, and a proper macrostate can be produced by aggregating the original landscape gradient with a sliding window whose size is  $2 \times 2$  pixels. However, this approach removes the overlap while sliding the window, as described in full by Gao and Li [17]. The method only works if both the length and the width of the landscape gradient are the  $2^n$ , where  $n$  is a positive integer. Two DEMs obtained from the Geospatial Data Cloud (<http://www.gscloud.cn>), with dimensions of  $512 \times 512$  pixels (Figure 10) were analyzed to evaluate the calculation accuracy of the absolute Boltzmann entropy in Table 5.



**Figure 10.** Two DEMs obtained from the Geospatial Data Cloud site (<http://www.gscloud.cn>). (a) One DEM with low heterogeneity; (b) Another DEM with low heterogeneity.

**Table 5.** A comparison of absolute Boltzmann entropy computation by the resampling-based method and the aggregation-based method.

DEMs	Resampling-Based Method			Aggregation-Based Method		
	Accurate Values	Estimated Values	Relative Error (%)	Accurate Values	Estimated Values	Relative Error (%)
(a)	$5.6 \times 10^7$	$4.5 \times 10^7$	19.6	$1.4 \times 10^5$	$1.5 \times 10^5$	7.1
(b)	$7.0 \times 10^7$	$5.2 \times 10^7$	25.7	$1.6 \times 10^5$	$1.7 \times 10^5$	6.3

$$\text{Note: Relative error} = \frac{|\text{Accurate values} - \text{Estimated values}|}{\text{Accurate values}} \times 100\%$$

Table 5 reveals that the relative errors generated by the aggregation-based method were lower than those by the resampling-based method. It is possible that the resampling-based method is poorly suited for DEMs with low heterogeneity because it does not conform to its thermodynamic analog.

## 7. Conclusions and Future Work

In this study, we have proposed and tested a method for efficiently estimating the absolute Boltzmann entropy of numerical raster data. The proposed method borrows the idea of head and tail breaks to classify all points into a few classes. This classification scheme not only reduces the complexity of the curve but it also reflects the scaling or hierarchy pattern of a dataset with a heavy-tailed distribution. Meanwhile, the classification scheme naturally determines the optimal number of classes and class intervals. The average of the data values in a given class is able to serve as the representative

value of that class because the differences among them are comparatively small. This proposed method substitutes the representative value into a linear function, and the expression of that function can be obtained through an inverse solution. The remaining representative values can be estimated from this function. After getting these estimates, the method computes the absolute Boltzmann entropy of the numerical raster data.

Our experimental results showed that the proposed method achieved higher computational efficiency than the previous method when applied to the same data as Gao et al. [18]. The new method saved about 99% of the computation time when applied to eight  $600 \times 600$  pixel DEMs. However, two DEMs with low heterogeneity (DEM b2 and DEM c2) encountered high error rates, and the relative errors of the remaining six DEMs were not sufficiently stable. To solve this problem, we tested a range of numbers of points to fit the linear function. These results showed that four-point fitting was the most suitable solution. Computation time was reduced by about 98% when dealing with eight  $600 \times 600$  pixel DEMs. Relative errors of the six heterogeneous DEMs were relatively stable. In addition, to improve calculation accuracy, we used an aggregation-based method [17] to replace the resampling-based method to improve the number of points involved in linear fitting. Results demonstrated that this method has the ability to improve the calculation accuracy for DEMs with low heterogeneity.

The proposed method is not perfect, and more extensive research is required. This paper assumed that the relative Boltzmann entropies of all points decrease at each step, and it ignored the situation in which the relative Boltzmann entropy of a given individual point is larger than the previous point. Moreover, this paper sets the percent of all heads and tails at 40% and 60%, respectively. It is possible that the number of points in the head was less than that in the tail but we rounded them off. Hence, the process of classification can only be terminated when the number of points in the head equals 1. An alternative termination condition would be when the number of points in the head is greater than or equal to that in the tail [20], which would make it possible to terminate the process of classification earlier.

**Author Contributions:** Conceptualization, H.Z. and Z.W.; methodology, H.Z. and Z.W.; validation, H.Z. and Z.W.; formal analysis, Z.W.; investigation, Z.W.; resources, H.Z.; data curation, Z.W.; writing—original draft preparation, H.Z. and Z.W.; writing—review and editing, H.Z.; supervision, H.Z.; project administration, H.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was jointly funded by National Natural Science Foundation of China (NSFC) (Grant No. 41471383 and 41971330), and the Program of Science and Technology of Sichuan Province, China (Grant No. 20YYJC2919).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Rao, K.N. Unit-5 GIS Data Models and Spatial Data Structure. *IGNOU*. 2017. Available online: <http://egyankosh.ac.in/bitstream/123456789/39613/3/MGY-003E-B2.pdf> (accessed on 31 January 2020).
2. Blasch, E.P.; Bryant, M. Information assessment of SAR data for ATR. In Proceedings of Proceedings of the IEEE 1998 National Aerospace and Electronics Conference, Dayton, OH, USA, 17 July 1998.
3. Roberts, J.W.; Aardt, J.A.v.; Ahmed, F.B. Assessment of image fusion procedures using entropy, image quality, and multispectral classification. *J. Appl. Remote Sens.* **2008**, *2*, 1–28.
4. Ramesh, K.P.; Gupta, S.; Blasch, E.P. Image fusion experiment for information content. In Proceedings of the 2007 10th International Conference on Information Fusion, Quebec, QC, Canada, 9–12 July 2007.
5. Bajwa, S.; Bajcsy, P.; Groves, P.; Tian, L. Hyperspectral image data mining for band selection in agricultural applications. *Trans. ASAE* **2004**, *47*, 895. [[CrossRef](#)]
6. Sotoca, J.M.; Pla, F.; Klaren, A.C. Unsupervised band selection for multispectral images using information theory. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26 August 2004.
7. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [[CrossRef](#)]

8. Fan, Y.; Yu, G.; He, Z.; Yu, H.; Bai, R.; Yang, L.; Wu, D. Entropies of the Chinese land use/cover change from 1990 to 2010 at a county level. *Entropy* **2017**, *19*, 51. [[CrossRef](#)]
9. Ekström, M. Quantifying spatial patterns of landscapes. *AMBIO J. Hum. Environ.* **2003**, *32*, 573–577. [[CrossRef](#)] [[PubMed](#)]
10. Li, Z.; Huang, P. Quantitative measures for spatial information of maps. *Int. J. Geogr. Inf. Sci.* **2002**, *16*, 699–709. [[CrossRef](#)]
11. Haralick, R.M.; Shanmugam, K.; Dinstein, I.H. Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *6*, 610–621. [[CrossRef](#)]
12. Brink, A. Using spatial information as an aid to maximum entropy image threshold selection. *Pattern Recognit. Lett.* **1996**, *17*, 29–36. [[CrossRef](#)]
13. Rakshit, S.; Mishra, A. Estimation of structural information content in images. In Proceedings of the Asian Conference on Computer Vision, Berlin/Heidelberg, Germany, 13–16 January 2006.
14. Cushman, S.A. Thermodynamics in landscape ecology: The importance of integrating measurement and modeling of landscape entropy. *Landsc. Ecol.* **2015**, *30*, 7–10. [[CrossRef](#)]
15. Cushman, S.A. Calculating the configurational entropy of a landscape mosaic. *Landsc. Ecol.* **2016**, *31*, 481–489. [[CrossRef](#)]
16. Vranken, I.; Baudry, J.; Aubinet, M.; Visser, M.; Bogaert, J. A review on the use of entropy in landscape ecology: Heterogeneity, unpredictability, scale dependence and their links with thermodynamics. *Landsc. Ecol.* **2015**, *30*, 51–65. [[CrossRef](#)]
17. Gao, P.; Li, Z. Aggregation-based method for computing absolute Boltzmann entropy of landscape gradient with full thermodynamic consistency. *Landsc. Ecol.* **2019**, *34*, 1837–1847. [[CrossRef](#)]
18. Gao, P.; Zhang, H.; Li, Z. A hierarchy-based solution to calculate the configurational entropy of landscape gradients. *Landsc. Ecol.* **2017**, *32*, 1133–1146. [[CrossRef](#)]
19. Gao, P.; Zhang, H.; Li, Z. Boltzmann Entropy for the Spatial Information of Raster Data. In Proceedings of the Abstracts of the ICA, Tokyo, Japan, 15–20 July 2019.
20. Jiang, B. Head/tail breaks: A new classification scheme for data with a heavy-tailed distribution. *Prof. Geogr.* **2013**, *65*, 482–494. [[CrossRef](#)]
21. Andreadis, G.; Lampridou, E.; Sherar, P. A Computer-Aided Design (CAD) Embroidery Font Digitizing System. *J. Autom. Control Eng.* **2015**, *3*. [[CrossRef](#)]
22. MacDonald, B.D.; Rowe, A.M. Experimental and numerical analysis of dynamic metal hydride hydrogen storage systems. *J. Power Sources* **2007**, *174*, 282–293. [[CrossRef](#)]
23. Jiang, B.; Jia, T. Zipf's law for all the natural cities in the United States: A geospatial perspective. *Int. J. Geogr. Inf. Sci.* **2011**, *25*, 1269–1281. [[CrossRef](#)]
24. Jiang, B.; Yin, J. Ht-index for quantifying the fractal or scaling structure of geographic features. *Ann. Assoc. Am. Geogr.* **2014**, *104*, 530–540. [[CrossRef](#)]

