9-2008

Kernel Density Estimation of Traffic Accidents in a Network Space

Zhixiao Xie
Florida Atlantic University, xie@fau.edu

Jun Yan
Western Kentucky University, jun.yan@wku.edu

Recommended Repository Citation

Xie, Zhixiao and Yan, Jun. (2008). Kernel Density Estimation of Traffic Accidents in a Network Space. *Computers, Environment, and Urban Systems*, 35 (5), 396-406.
Available at: http://digitalcommons.wku.edu/geog_fac_pub/3

Kernel Density Estimation of Traffic Accidents in a Network Space

Zhixiao Xie [a], Jun Yan [b]
[a] Department of Geosciences, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431, USA
[b] Department of Geography and Geology, Western Kentucky University, Bowling Green, KY 42101, USA

Abstract: A standard planar Kernel Density Estimation (KDE) aims to produce a smooth density surface of spatial point events over a 2-D geographic space. However the planar KDE may not be suited for characterizing certain point events, such as traffic accidents, which usually occur inside a 1-D linear space, the roadway network. This paper presents a novel network KDE approach to estimating the density of such spatial point events. One key feature of the new approach is that the network space is represented with basic linear units of equal network length, termed *lixel* (linear pixel), and related network topology. The use of *lixel* not only facilitates the systematic selection of a set of regularly spaced locations along a network for density estimation, but also makes the practical application of the network KDE feasible by significantly improving the computation efficiency. The approach is implemented in the ESRI ArcGIS environment and tested with the year 2005 traffic accident data and a road network in the Bowling Green, Kentucky area. The test results indicate that the new network KDE is more appropriate than standard planar KDE for density estimation of traffic accidents, since the latter covers space beyond the event context (network space) and is likely to overestimate the density values. The study also investigates the impacts on density calculation from two kernel functions, *lixel* lengths, and search bandwidths. It is found that the kernel function is least important in structuring the density pattern over network space, whereas the *lixel* length critically impacts the local variation details of the spatial density pattern. The search bandwidth imposes the highest influence by controlling the smoothness of the spatial pattern, showing local effects at a narrow bandwidth and revealing "*hot spots*" at larger or global scales with a wider bandwidth. More significantly, the idea of representing a linear network by a network system of equal-length *lixel*s may potentially lead the way to developing a suite of other network related spatial analysis and modeling methods.

Keywords: Network Space, Lixel, Kernel Density Estimation, Traffic Accidents, Hot Spots

1. Introduction

To reduce traffic accidents and improve road safety, it is crucial to understand how, where and when traffic accidents occurred. An improved understanding of spatial patterns of traffic accidents can make accident reduction efforts more effective. For instance, by knowing where and when traffic accidents usually occur, law enforcement can conduct more efficient patrols and highway departments can disseminate more effectively to drivers the critical information about roadway conditions. In reality, the occurrences of traffic accidents are seldom random in space and time. In most cases, traffic accidents form clusters (known as "*hot spots*") in geographic space. This is because the occurrence of traffic accidents along a certain roadway segment is largely determined by its traffic volume, which is well-known to exhibit distinct spatial and temporal patterns (Black, 1991). There are some other important factors that may impact the distribution of traffic accidents, including natural and environmental characteristics such as physical environment (steep slope, sharp turn), weather (rain, snow, wind, and fog), configuration of highway networks such as locations of access and egress points, deficient design and maintenance of highways, etc. All of these factors more or less are associated with distinct spatial patterns as well.

Spatial analysis of point events, known as point pattern analysis (PPA), has been widely examined by spatial scientists and a variety of methods have been developed for detecting "*hot spots*" of point events. The PPA methods can be classified into two broad categories (Bailey & Gatrell, 1995; O'Sullivan & Unwin, 2002): (1) Methods examining the first-order effects of a spatial process, (2) Methods examining the second-order effects of a spatial process. The first group focuses on the underlying properties of point events and measures the variation in the mean value of the process. It includes methods such as quadrat count analysis, kernel density estimation and etc. The second group mainly examines the spatial interaction (dependency) structure of point events for spatial patterns, and includes methods such as nearest neighbor statistics, G function, F function, K function and etc. Out of these two categories of methods, Kernel Density Estimation (KDE) is one of the most popular methods for analyzing the first order properties of a point event distribution (Silverman, 1986; Bailey & Gatrell, 1995) partially because it is easy to understand and implement. Some KDE tools are already made available in some leading commercial GIS software, *e.g.* the Spatial Analyst Extension of ESRI's ArcGIS, as well as some popular spatial statistical analysis software, such as CrimeStat (Levine, 2004). The planar KDE has been used widely for traffic accidents "*hot spots*" analysis and detection. The recent examples include study of urban cyclists traffic hazard intensity (Delmelle & Thill, *in press*), pedestrian crash zones detection (Pulugurtha, Krishnakumar, & Nambisan, 2007), wildlife–vehicle accident analysis (Krisp & Durot, 2007), highway accident "*hot spot*" analysis (Erdogan et al., 2008) and etc. The purpose of KDE is to produce a smooth density surface of point events over space by computing event intensity as density estimation. In planar KDE, the space is characterized as a 2-D homogeneous Euclidian space and density is usually estimated at a large number of locations that are regularly spaced (a grid). However, in analyzing the spatial pattern of traffic accidents, which usually occur on roadways and inside a network, the assumption of homogeneity of 2-D space does not hold and the relevant KDE methods are not readily applicable. Many other types of human-induced point events also exhibit similar property in that their distributions are constrained to only the network portion (network space) of the 2-D

Euclidean space, such as residential houses, commercial sites, street lights, moving vehicles, and etc. In short, the uniformity of 2-D space is basically too strong an assumption for the analysis of point events occurring in 1-D infinite space (Miller, 1999a). Special considerations are thus needed for measuring such point events occurring in network spaces[1].

In the early 1990s, spatial scientists started to realize the limitations of spatial methods originated in the 2-D Euclidean space when applying them directly to network-constrained phenomena. A large number of studies have been conducted since then in a variety of application domains, attempting to extend the conventional 2-D spatial methods to network spaces, including network autocorrelation (Black, 1992; Black & Thomas, 1998), network Huff model of market area analysis (Miller, 1994; Okabe & Kitamura, 1996; Okabe & Okunuki, 2001), network distance-decay (Kent et al., 2005), space-time accessibility measures (Kwan, 1998; Miller, 1999b), space-time clustering (Black, 1991) and etc. In particular, increased attentions are recently paid to the applications of standard spatial statistical methods in analyzing spatial point events in a network space (Okabe et al., 1995; Okunuki & Okabe, 1998; Okabe & Yamada, 2001; Yamada & Thill, 2004; Lu & Chen, 2007; Yamada & Thill, 2007). For instance, the K-function (another popular PPA method), has been extended to network spaces. A network version of K-function and its computational implementation are described in Okabe and Yamada (2001). To examine the advantages of network K-function, Yamada & Thill (2004) compared three versions of K-functions in their ability of analyzing traffic

[1] In this paper, a network space is defined as a simplified and abstracted representation of the real world road network, which occupies a portion of the actual 2-D geographic space. It is represented as 1-D lines and line-intersections. The roadway width, traffic direction, and multi-lane properties are not considered for simplicity of concept demonstration.

accident patterns, namely planar K-function, network-constrained K-function and network K-function. Their findings indicate that standard planar K-function tends to over-detect clusters as it searches for the clustered patterns by comparing with the random patterns over the entire 2-D space instead of the network space, which itself often exhibits clustering tendency (*e.g.* the streets in the central city are often denser than those at the outskirts). In another study, Lu & Chen (2007) reach a similar conclusion that planar K-function is likely to produce false alarms of clusters when being applied to detect hot-spots of vehicle thefts in the San Antonio, Texas area. Due to the increasing popularity of the network K-function, an ArcGIS-based software tool, known as Spatial Analysis on a NETwork (SANET)[2], was recently developed by a group of researchers at the University of Tokyo, Japan (Okabe et al., 2006). SANET offers network version of both global and local K-functions as well as some additional utility tools for data processing.

In contrast to the new developments of the network K-function, few studies have attempted to extend the KDE methods to a network space. Recently, Borruso (2005) analyzed patterns of point events distributed on a network with a modified KDE, termed as Network Density Estimation (NDE) in his paper, which considers the kernel as a density function based on network distances. Borruso (2005) pointed out the possibility of extending the standard 2-D KDE to network spaces for identifying potential "linear" clusters along roadways, however in his study, the kernel is still area based (using network service area) and the outcome (point density) is still mapped onto a 2-D Euclidian space. In essence, it is still a KDE in a planar pace instead of a network space.

[2] SANET software tools can be obtained free of charge upon written request.

See Yamada & Thill (2007) for a typology of situations involving planar and network spaces in spatial analysis. In addition, the spatial pattern of such kind of point events is better measured with density values per linear unit over a network instead of per area unit over a 2-D space. Indeed, in real-world applications, the density of traffic accidents is often reported as the number of accidents over a defined linear unit (*e.g.* per mile) rather than per area unit (*e.g.* per square mile).

This paper presents a novel network KDE approach to estimating the density of traffic accidents strictly over a network space. As secondary objectives, the study also investigates the impacts on density calculation from two different kernel functions, *lixel* lengths, and search bandwidths. Although developed initially for traffic accidents, this new approach could be used to examine the spatial pattern of any point events, as far as their distribution is limited within network spaces. The remainder of the paper is organized as follows. The basic concepts of network KDE are discussed in Section 2. The computational algorithm is detailed in Section 3 along with discussions of some implementation issues. In Section 4, we present a case study with a real road network and traffic accident data. Discussions are made and some conclusions are drawn in Section 5.

2. Kernel Density Estimation in a Network Space

2.1. Planar Kernel Density Estimation

Network KDE is an extension of the standard 2-D KDE and a brief summary of key aspects of the standard 2-D KDE is necessary. The general form of a kernel density estimator in a 2-D space, termed as planar KDE in the rest of this paper, is given by:

$$\lambda(s)=\sum_{i=1}^{n}\frac{1}{\pi r^2}k(\frac{d_{is}}{r}) \qquad (1)$$

where $\lambda(s)$ is the density at location *s*, *r* is the search radius (bandwidth) of the KDE (only points within *r* are used to estimate $\lambda(s)$), *k* is the weight of a point *i* at distance d_{is} to location *s*. *k* is usually modeled as a function (called kernel function) of the ratio between d_{is} and *r*. As a result, rather than choosing a uniform function that gives equal weight to all points within the bandwidth *r*, the KDE uses a model function through which "distance decay effect" can be taken into account – basically the longer the distance between a point and location *s*, the less that point is weighted for calculating the overall density. In the end, all the points within the bandwidth *r* of location *s*, weighted more or less depending on its distance to *s*, are summed for calculating the density at *s*.

A number of forms of model functions, known as kernel functions, can be used to measure the "*distance decay effect*" in the spatial weights *k*, such as Gaussian, Quartic, Conic, negative exponential, and epanichnekov (Levine, 2004; Gibin et al., 2007). Three forms of kernel functions are most commonly used (Schabenberger & Gotway, 2005, p. 111) and are discussed below, including:

(1) Gaussian function:

$$k(\frac{d_{is}}{r})=\frac{1}{\sqrt{2\pi}}\exp(-\frac{{d_{is}}^2}{2r^2}) \;, \qquad \text{when } 0 < d_{is} <= r$$

$$k(\frac{d_{is}}{r})=0 \;, \text{ when } d_{is} > r$$

(2) Quartic function (which approximates Gaussian function):

$$k(\frac{d_{is}}{r}) = K(1-\frac{d_{is}^{\;2}}{r^2}) \;, \qquad \text{when } 0 < \mathrm{d_{is}} <= r$$

$$k(\frac{d_{is}}{r}) = 0 \;, \;\; \text{when } \mathrm{d_{is}} > r$$

Where *K* is a scaling factor and its purpose is to ensure the total volume under Quartic curve is 1. The common values used for K include $\frac{3}{\pi}$ and $\frac{3}{4}$, i.e.

$$k(\frac{d_{is}}{r}) = \frac{3}{\pi}(1-\frac{d_{is}^{\;2}}{r^2}) \;\; \text{or} \;\; k(\frac{d_{is}}{r}) = \frac{3}{4}(1-\frac{d_{is}^{\;2}}{r^2})$$

(3) Minimum variance function

$$k(\frac{d_{is}}{r}) = \frac{3}{8}(3-5\frac{d_{is}^{\;2}}{r^2}) \;, \qquad \text{when } 0 < \mathrm{d_{is}} <= r$$

$$k(\frac{d_{is}}{r}) = 0 \;, \;\; \text{when } \mathrm{d_{is}} > r$$

A wealth of literature has examined the effects of the two key parameters of planar KDE, *i.e.* kernel function *k* and search bandwidth *r*, on the resultant density pattern. There exists a consensus that the choice of the kernel function *k* is less important than the choice of search bandwidth *r* (Silverman, 1986; Bailey & Gatrell, 1995; O'Sullivan & Unwin, 2002; Schabenberger & Gotway, 2005; O' Sullivan & Wang, 2007). It is also agreed that the value of search bandwidth *r* usually determines the smoothness of the estimated density -- the larger the *r* the smoother is the estimation.

2.2. *Network Kernel Density Estimation*

With the linear nature of network spaces in mind, this paper proposes to use the following form of kernel density estimator for the density estimation of network-constrained point events, such as traffic accidents, in a network space:

$$\lambda(s) = \sum_{i=1}^{n} \frac{1}{r} k(\frac{d_{is}}{r})$$

Instead of calculating the density over an area unit, the equation estimates the density over a linear unit. Any of the three forms of kernel functions discussed previously may be used. As shown in previous sub-sections, many earlier studies in planar KDE have concluded that the choice of kernel function is not as important as the choice of bandwidth *r*. We conjecture that the relative significance of the two parameters on density estimation might not change much in the new network KDE, since there is no particular reason for them to perform differently. To confirm our conjecture, we do implement two kernel functions, Gaussian and Quartic functions, and compare their impacts on the resultant density pattern in our case study. To verify the role of search bandwidth in network KDE, we also examine how the density pattern will be impacted when different search bandwidth are chosen.

It is necessary to emphasize that the network KDE differs from the planar KDE in several aspects: (1) the network space is used as the point event context, (2) both search bandwidth and kernel function are based on network distance (calculated as the shortest-path distance in a network) instead of straight-line Euclidean distance, and (3) density is measured per linear unit. These differences are illustrated in a graphic form in Figure 1. It can be noticed that network KDE is a 1-D measurement while planar KDE is a 2-D one. As a result, the actual density values estimated by them would be very different for the same point events dataset. This illustrative example also suggests that a Planar KDE could possibly over-detect clustered pattern , with four traffic accidents falling within the search bandwidth and hence included in density estimation for the focal point (x) in the case of Planar KDE, while only two in the case of Network KDE.

3. Algorithm and Its Implementation

3.1. Computational Algorithm

The basic algorithm for network KDE is presented as follows. The basic terms are defined and highlighted when they appear the first time, and some are illustrated in Figure 2. The key implementation issues are described further in the next sub-section.

(1) Create a segment-based linear reference system out of the original road network, with each ***segment*** being a line segment between two neighboring road intersections. A dangling line segment between a road intersection and a neighboring road end point is also a *segment*. If there are multiple links between two intersections, each link is treated as a separate *segment*.

(2) Divide each *segment* into basic linear units of a defined network length *l*. A basic linear unit is equivalent to a cell in a 2-D raster grid. For description simplicity, we call it ***lixel*** (a contraction of the words *linear* and *pixel*) in the paper. The use of *lixel* not only facilitates the systematic selection of a set of regularly spaced locations along a network for density estimation, but also makes the practical application of the network KDE feasible by significantly improving the computation efficiency. The residual of the division (if there is one), i.e. the last

lixel with length shorter than *l*, is a partial *lixel*, the processing of which will be detailed later. The intersection point of two *lixel*s is called ***lxnode***.

(3) Create a network of *lixel*s by establishing the network topology between *lixel*s, as well as between *lixel*s and *lxnode*s.

(4) Create the center points of all the *lixel*s. They are termed ***lxcenter***s.

(5) Select a point process (traffic accidents in this study), which has to be a type of point events occurring within the network space.

(6) For each point event, find its nearest *lixel*. The total number of events nearest to a *lixel* is counted and assigned to the *lixel* as a property. Those *lixel*s with one or more accidents assigned to them are used as ***source lixels***. Aggregating events to *lixels* is one important step for improving the computational efficiency of the algorithm. The possible impacts and future research plans are further described in the conclusions and discussions section.

(7) Define a search bandwidth *r*, measured with the shortest-path network distance.

(8) Calculate the shortest-path network distance from the *lxcenter* of each *source lixel* to *lxcenter*s of all its neighboring *lixel*s within the search bandwidth *r*. It should be noted not only the first order nearest neighbor *lixel*s, but all the neighbor *lixel*s with network distance not farther than *r* are taken into consideration.

(9) At the *lxcenter* of each *source lixel* and all its neighboring *lixel*s, calculate a density value based on a selected kernel function, the network distance, and the number of events on the *source lixel*.

(10) At the *lxcenter* of each *lixel* within the search bandwidth of any *source lixel*s, sum the density values from different *source lixel*s and assign the total density to the *lixel*; for all other *lixel*s, the density value is zero by default.

3.2. Key Implementation Issues

As shown in the basic algorithm, there are some key implementation issues to be carefully considered in density estimation in a network space, including dividing *segment*s into *lixel*s, selecting kernel functions, defining search bandwidth, and computing *lixel*-based density. In a 2-D space, a lattice of grid cells can be placed to exhaustively cover the space for a systematic selection of a set of locations for density estimation. But in a network space, this may not be suitable and special treatments are needed since the real-world networks are usually represented as linear features and are often irregularly configured. Therefore, we have proposed a simple but effective segmentation solution in this study: (1) first break the network into a series of independent *segment*s with each *segment* being a line segment between two neighboring road intersections; (2) then divide each *segment* into equal-length *lixel*s. Both steps are done in a linear reference system (LRS). The center of a *lixel* is used as the density estimation location and the estimated density at the center is used to represent the entire *lixel*. In essence, like a cell in a 2-D raster representation, a *lixel* is treated as a homogeneous unit and the internal variation is not considered. After segmentation, the topological relationship is established between *lixel*s, and between *lixel*s and *lxnode*s. The resultant network system of *lixel*s is the basis for assigning accident points, measuring network distance, and ultimately calculating the accident densities for *lixel*s.

One issue to notice in the segmentation process is that the length of a *segment* (e.g. 51 meters) may not be exactly an integer number times of a defined *lixel* length (e.g. 10 meters). Hence, a residual *lixel* with length, here 1 m, shorter than the defined *lixel* length often results for this kind of *segment*. These residual *lixel*s do not actually present any issues in implementation. The density values at the *lxcenter*s of these residual *lixel*s is calculated in the same way as for regular *lixel*s, based on the same kernel functions and the actual network distance from their *lxcenter*s to the centers of the *source lixel*s, although the network distance will not be integer number times of the defined *lixel* length. A minor caveat is that the resolutions (length) are different between the regular and residual *lixel*s, however, the overall density pattern should not be affected because residual *lixel*s generally only amounts to a very small proportion of the entire inventory of *lixel*s. Further, as far as the *lixel* length is sufficiently short (e.g. 10 m), including or excluding a single residual *lixel* in a hot spot may have trivial effects on the "*hot spots*" detection even at local scales.

Another related issue is how to determine whether a *lixel* lies within the search bandwidth from a *source lixel*. There are at least two different options when a *lixel* could be labeled within a search bandwidth: (1) if the network distance from the *lxcenter* of a *source lixel* to the farthest end point of the *lixel* is shorter than or equal to the search bandwidth; or (2) if the network distance from the *lxcenter* of a *source lixel* to the *lxcenter* of the *lixel* is shorter than or equal to the search bandwidth. The two options may perform a little differently for the marginal neighbor *lixel*s, but the impacts should be trivial. In this paper, we adopted the second option.

As described in Steps (8)-(10) in the algorithm, the density computation is an iterative process starting from *a source lixel* to all its neighbor *lixel*s within the search bandwidth. The number of accidents associated with a *source lixel* is used as a multiplier of the density values for all the *lixel*s within the search bandwidth. After computing the density for one *source lixel* and all its neighbors, the algorithm repeats the same process for another *source lixel* till all the *source lixel*s are processed. For a *lixel* within the search bandwidth of multiple *source lixel*s, its density is the cumulative value of the density derived from all the sources. Since the number of *source lixel*s is generally much smaller than the total number of *lixel*s in a network space, this density computation process is obviously much more efficient than computing density for all *lixel*s, most of which may never fall within the search bandwidth of a *source lixel*.

4. Case Study – The Analysis of Traffic Accident Patterns in Bowling Green, Kentucky

The proposed algorithm is implemented in the ESRI ArcGIS environment, using Microsoft Visual C# 2005. The test dataset includes a real transportation network system in the Bowling Green, Kentucky area, and the traffic accident data for year 2005 (Figure 3). The traffic accident data is provided by the Kentucky State Police Department. The point location of each accident is recorded as a pair of longitude and latitude via the carry-on GPS unit within a reporting police car. A total of 3226 traffic accidents are analyzed in the case study. A series of tests are conducted to demonstrate the applicability of the algorithm, to compare the difference between the new algorithm and a standard planar KDE, and to examine the impacts of different parameters on density calculation, including kernel functions, *lixel* length, and search bandwidth. Two kernel functions, Gaussian and Quartic, are compared at a fixed search bandwidth of 100 m for local details and a search bandwidth of 1000 m for the overall pattern, both with a *lixel* length of 10 m. Four versions of segmentation scenarios, with the *lixel* length at 5 m, 10 m, 50 m, and 100 m respectively, are tested at a fixed search bandwidth of 100 m and with a Gaussian kernel function. We also examine the impacts of search bandwidth at local and larger spatial extents. Total six search bandwidths are used, including 20 m, 100 m, 250 m, 500 m, 1000 m, and 2000 m, all with the same *lixel* length (10 m) and a Gaussian kernel function.

4.1. Comparison of Planar KDE and Network KDE

An intuitive approach to present the spatial pattern of accidents is to map the accident locations or the simple accident count per *lixel* after each accident is assigned to its closest *lixel*. Figures 4-a and 4-b illustrate the spatial pattern of traffic accidents for a small part of the study area with these two intuitive approaches respectively. Figures 4-c and 4-d show the density values calculated for the same area using a standard planar KDE (10-m raster cell) and the new network KDE (10-m *lixel*) respectively, both with a Gaussian kernel and a 100-m search bandwidth. Figures 5-a and 5-b present the overall spatial pattern of density values computed with the network KDE (10-m *lixel*) and the planar KDE (10-m raster cell) respectively, both with a Gaussian kernel but a wider search bandwidth of 1000 m. A wider bandwidth is used to better reveal the differences of the two KDE approaches and the role of search bandwidth will be discussed further in section 4.4. By a simple visual comparison, both KDE based density values present a much informative pattern of accident likelihood. However the density maps of the planar KDE and network KDE are rather different, although some common hot (high value) regions can be identified in both maps. The density surface by the planar KDE exhaustively fills out the entire study area (although some areas with value of zero) while that by the network KDE only covers the space occupied by the street network. In addition, the maps indicate that the planar KDE is likely to overestimate the density values in comparison to the network KDE. In Figures 4-c and 4-d, the density value for the planar KDE can reach as high as over 2000 (per km^2) (Figure 4-c), in comparison, the highest value for the network KDE is only about 140 (per km) (Figure 4-d).

4.2. The Impacts of Kernel Functions

Figures 6-a and 6-b illustrate the spatial patterns of density values computed in a small part of the study area with the proposed network KDE algorithm based on kernel functions of Gaussian and Quartic respectively, both at a 10-m *lixel* length and with a100-m search bandwidth. It is obvious that the two kernel functions result in very similar local density variations. Figures 7-a and 7-b show the overall density pattern for the study area using a Gaussian and a Quartic function respectively, at the same *lixel* length (10 m) and with the same search bandwidth (100 m). It appears that the choice of kernel functions also make little difference in the overall density pattern. Only that the density values estimated with Quartic kernel are higher than those with Gaussian kernel. These observations corroborate our conjecture that kernel functions are least important in structuring the density pattern in network KDE, similar to the case in planar KDE.

4.3. The Impacts of Lixel Length

Lixel length, like raster cell size in a planar KDE, significantly affects the local variation details of the network density value pattern. Figure 8(a)-(d) show the network KDE estimated density values, with the *lixel* length at 5 m, 10 m, 50 m, and 100 m respectively. Although the kernel function is the same (Gaussian), and with the same search bandwidth (100 m), the density values along roads lose local variation details as *lixel* length increases. The larger *lixel* lengths effectively hide the detailed structures shown at finer resolutions. For example, for the same network *segment* from A to B in Figures 8(a)-(d), we can notice Figure 8-a shows much more local details of variation in the density values than Figure 8-d. As a trivial point, the density values remain in almost the same range between 0 to about 200 per kilometer even when different *lixel* lengths are used.

4.4. The Impacts of Search ***Bandwidth***

Search bandwidth plays the most significant role in structuring the network density pattern. As shown in a local part of the study area (Figure 9(a)-(d)), the density pattern gets smoother with increasing search bandwidth (20 m, 100 m, 250 m, and 500 m respectively), even when the kernel function is the same (Gaussian) and at the same *lixel* length (10 m),. The density values are almost invariant with a 500-m search bandwidth (Figure 9-d), whereas the density variation pattern is quite bumpy with a 20-m bandwidth (Figure 9-a).

The Figure 9 only shows the response of density value to the search bandwidth at a small part of the study area with relatively narrower bandwidths. Figures 10(a)-(f) present the impacts of the search bandwidth (20 m, 100 m, 250m, 500 m, 1000 m, and 2000 m respectively) on the overall density pattern for the whole study area. It appears that the narrow bandwidths (20 m, 100 m, and 250 m) may produce patterns suitable for presenting local effects or "*hot spots*" at smaller scales. As the search bandwidth increases from 20 m to 2000 m, the local "*hot spots*" are gradually combined with their neighbors, and larger clusters appear. The maps at wider search bandwidths (500 m, 1000 m, 2000 m) seemingly give better sense of locations of the "*hot spots*" at larger spatial scales.

4.5. Density Visualization

In a 2-D space represented by regularly spaced grid cells or points, a density surface can be easily displayed in a raster GIS. The visualization of density values across a network needs to adopt different strategies to reflect the linear nature of the network space. Usually three visual variables are associated with drawing line features, including color, width, and height. The visualization can be implemented with one of the three variables or in the combination of two and even all three of them. We already show density patterns with the width variable in Figures 4, 6, 7, 8, 9, and 10. Figure 5-a uses color (gray tone) to show the overall density pattern estimated by the network KDE. Figure 11 presents the spatial pattern of density using 3-D symbols (gray tone and height) in ESRI ArcScene. The visualization methods presented in these figures appear to be effective in presenting the network constrained accident density pattern. As a quick and general observation, it appears that major corridors in the study area tend to have relatively higher estimated density values of traffic accidents. And several major intersections can be visually identified as "*hot spots*" of traffic accidents.

5. Conclusions and Discussions

Recognizing the limitations of applying standard 2-D planar KDE methods in a network space, this paper develops a network KDE approach to characterizing the spatial patterns of traffic accidents on roadways. The basic operation unit of the new KDE algorithm is a *lixel* of a defined network length along linear road segments. All the accidents are assigned to their nearest *source lixel*s. The density value at the center point of a *lixel* is computed as the sum of kernel-function derived densities from all the *source lixel*s within a specified search bandwidth measured by the shortest-path network distance. As a case study, the new KDE algorithm is successfully applied in a real world road network system and traffic accidents dataset. The operational success of the application demonstrates the applicability of the algorithm. Visual comparison of the resultant density patterns of a standard planar KDE and the network KDE shows that the planar KDE covers space beyond the network context and over-estimates the density values. The case study also examines the impacts on network KDE density calculation from different kernel functions, *lixel* lengths, and search bandwidths. Although we only implement two kernel functions and cannot make sound conclusions, it does appear that the kernel functions have less important role in structuring the density patterns over network space. The *lixel* length, like the cell resolution in raster representation, impacts the local variation details of the spatial pattern of density. It is also found that the search bandwidth imposes the highest influence by controlling the smoothness of the spatial pattern. A narrower search bandwidth could reveal local effects, whereas a wider bandwidth makes "*hot spots*" much more obvious at large spatial scales.

It should be pointed out that the proposed algorithm only calculates the density value at the center point of each *lixel* and uses that value to represent the whole *lixel*. Although we can certainly compute density values for as many points along a linear segment as possible, we argue that it is appropriate to use the center point density value to approximate the density value over the *lixel* of a reasonable length, mainly because there are infinite points on a *lixel* of any length and a certain degree of simplification and abstraction is always needed for a practical application. Similar simplification and abstraction are not uncommon. For example, in the 2-D space raster representation, the center value of a grid cell has often been used to approximate the value of the whole cell. In addition, an equal-length segmentation approach is used in the current algorithm. In the future study, it may be interesting to examine other segmentation scenarios with a set of varied *lixel* lengths, e.g. shorter length for central urban, longer for rural areas, and so on.

It is very tempting to recommend an optimal *lixel* length. However, we avoid doing so since a universally applicable *lixel* length may not exist, like the cell resolution in a 2-D raster representation. In remote sensing, imagery of a particular resolution is chosen to suit specific application context, and it should be so in the network KDE. The length of *lixel*s must be carefully chosen based on application context. The inherent data quality of the road network and accident locations is also an important factor to take into consideration. In traffic accident, a 10-m *lixel* length should be sufficiently short if not optimal given the length of a vehicle, the direct impact area of an accident, and the location accuracy in measurement process. This *lixel* length may not suit other point events. We also refrain from recommending an optimal search bandwidth for similar reasons and believe search bandwidth selection should also be application dependent. It may be wise to use a set of search bandwidths to reveal "*hot spots*" from very local effects to global scale. In essence, this would lead to a multi-scale exploration, which is actually needed for many systems with hierarchical structure.

In the implementation, we aggregate all the accident points associated with a focal *source lixel* into an accident count of the focal *lixel*, and their locations on the focal *lixel* are not further distinguished. It is from the center point of focal *source lixel* that the network distance to each neighboring *lixel* center is calculated. The aggregation has the advantage of computational efficiency and more importantly it is closer to the reality in the sense that a real accident seldom occurs precisely at a dimensionless point location but occupying a certain length along a roadway. The location accuracy is also affected by measurement process and contributes uncertainties to the event position. Regardless, the impacts of aggregation on the measured distance and subsequently the density value should be minor. For comparison purpose, we will implement the network KDE with un-aggregated events in the future study.

In the current algorithm, road segments as well as *lixels* are also not further distinguished and are treated equally. In reality, road segments are different in their traffic flow capacity (e.g. the number of lanes and speed limit), directionality (one way or two ways, left turn or right turn) and etc. The same accidents will have different implications depending on the type of road segments they are on, and may have different impacts on the traffics ahead of or behind the accident vehicles. Such information are not accessible for the study area, however in areas where they are available, it may be beneficial to incorporate them into the density estimation. The network KDE also has one of the same fundamental drawbacks as the planar KDE for "*hot spots*" detection. Because no statistical significance is employed in the process, it is *ad hoc* and there is no indication of a density threshold above which "*hot spots*" can be confidently declared. Experiments with different density thresholds may be needed in real applications. Nevertheless, we believe the network KDE method presented in this paper could be useful and are readily applicable in real world setting for traffic accident decision making by different agencies. More significantly, although designed for KDE estimation, the idea of segmenting a linear road network into a network system of equal-length *lixel*s may potentially lead the way to developing a suite of other network related spatial analysis and modeling methods and have a larger impact than what is shown in this initial application.

Acknowledgement

The authors greatly appreciate the helpful comments of three anonymous reviewers and the detailed comments by the Editor, Dr. Jean-Claude Thill.

References

Bailey, T. C., & Gatrell, A. C. (1995). *Interactive Spatial Data Analysis*. Essex: Longman.

Black, W. R. (1991). Highway Accidents: A Spatial and Temporal Analysis. T*ransportation Research Record, 1318*, 75–82.

Black, W. R. (1992). Network Autocorrelation in Transport Network and Flow Systems. *Geographical Analysis, 24,* 207–222.

Borruso, G. (2005). Network Density Estimation: Analysis of Point Patterns over a Network. *Lecture Notes in Computer Science: Computational Science and Its Applications,* 3482, 126–132. Erdogan, S., Yilmaz, I., Baybura, T., & Gullu, M. (2008). Geographical information systems aided traffic accident analysis system case study: city of Afyonkarahisar. *Accident Analysis and Prevention, 40*(1), 174–181.

Delmelle, E. C., & Thill, J.-C. (2008). Urban Bicyclists – A Spatial Analysis of Adult and Youth Traffic Hazard Intensity. *Transportation Research Record*, in press.

Gibin, M., Longley, P., & Atkinson, P. (2007). Kernel Density Estimation and Percent Volume Contours in General Practice Catchment Area Analysis in Urban Areas. In the Proceedings of the *GIScience Research UK Conference (GISRUK) 2007*. Maynooth - Ireland.

Kent, J., Leitner, M., & Curtis, A. (2006). Evaluating the Usefulness of Functional Distance Measures When Calibrating Journey-to-Crime Distance Decay Algorithms. *Computers, Environment and Urban Systems, 30*(2), 181–200.

Krisp, J.M., & Durot, S. (2007). Segmentation of lines based on point densities —An optimisation of wildlife warning sign placement in southern Finland. *Accident Analysis and Prevention, 39*(1), 38–46.

Kwan, M-P., (1998). Space–Time and Integral Measures of Individual Accessibility: A Comparative Analysis Using A Point-Based framework. *Geographical Analysis, 30*(3), 191–216.

Levine, N. (2004). *CrimeStat III: A Spatial Statistics Program for the Analysis of Crime Incident Locations*. Ned Levine & Associates, Houston, TX, and the National Institute of Justice, Washington, DC.

Lu, Y., & Chen, X. (2007). False Alarm of Planar K-Function when Analyzing Urban Crime Distributed along Streets. *Social Science Research, 36*(2), 611–632.

Miller, H. J. (1994). Market Area Delineation within Networks Using Geographic Information Systems. *Geographical Systems, 1*(2), 157–173.

Miller, H. J. (1999a). Potential Contribution of Spatial Analysis to Geographic Information Systems for Transportation (GIS-T). *Geographical Analysis, 31*(4*),* 373–399.

Miller, H.J. (1999b). Measuring Space–Time Accessibility Benefits within Transportation Networks. *Geographical Analysis, 31*(2), 187–212.

Okabe, A., Yomono, H., & Kitamura, M. (1995). Statistical Analysis of the Distribution of Points on a Network. *Geographical Analysis, 27(*2), 152–75.

Okabe, A., & Kitamura, M. (1996). A Computational Method for Market Area Analysis on a Network. *Geographical Analysis, 28*(4), 330–349.

Okunuki, K., & Okabe, A. (1998). A Computational Method for Optimizing the Location of a Store on a Continuum of a Network When Users' Choice Behavior Follows the Hu_ Model. *Discussion Paper*, Center for Spatial Information Science at the University of Tokyo, No.19.

Okabe, A., & Okunuki, K. (2001). A Computational Method for Estimating the Demand of Retail Stores on a Street Network Using GIS. *Transactions in GIS, 5*(3), 209–220.

Okabe, A., & Yamada, I. (2001). The K-Function Method on a Network and its Computational Implementation. *Geographical Analysis,* 33, 271–290.

Okabe, A., Okunuki, K., & Shiode, S. (2006). SANET: A Toolbox for Spatial Analysis on a Network. *Geographical Analysis,* 38, 57–66.

O'Sullivan, D., & Unwin, D. J. (2002). *Geographic Information Analysis*. John Wiley, Hoboken, New Jersey.

O' Sullivan, D., & Wong, D. W. S. (2007). A Surface-Based Approach to Measuring Spatial Segregation. *Geographic Analysis, 39*(2), 147–168.

Pulugurtha, *S.S.,* Krishnakumar, V. K., & Nambisan, S. S. (2007). New methods to identify and rank high pedestrian crash zones: An illustration. *Accident Analysis and Prevention, 39*(4,) 800–811.Schabenberger, O., & Gotway, C. A. (2005). Statistical Methods for Spatial Data Analysis. Chapman & Hall/CRC, Boca Raton, Florida.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman Hall, London.

Yamada, I., & Thill, J.-C. (2004). Comparison of Planar and Network K-Functions in Traffic Accident Analysis. *Journal of Transport Geography,* 12, 149–158.

Yamada, I., & Thill, J.-C. (2007). Local Indicators of Network-Constrained Clusters in

Spatial Point Patterns. *Geographical Analysis, 39*(3), 268–292.

List of Figures

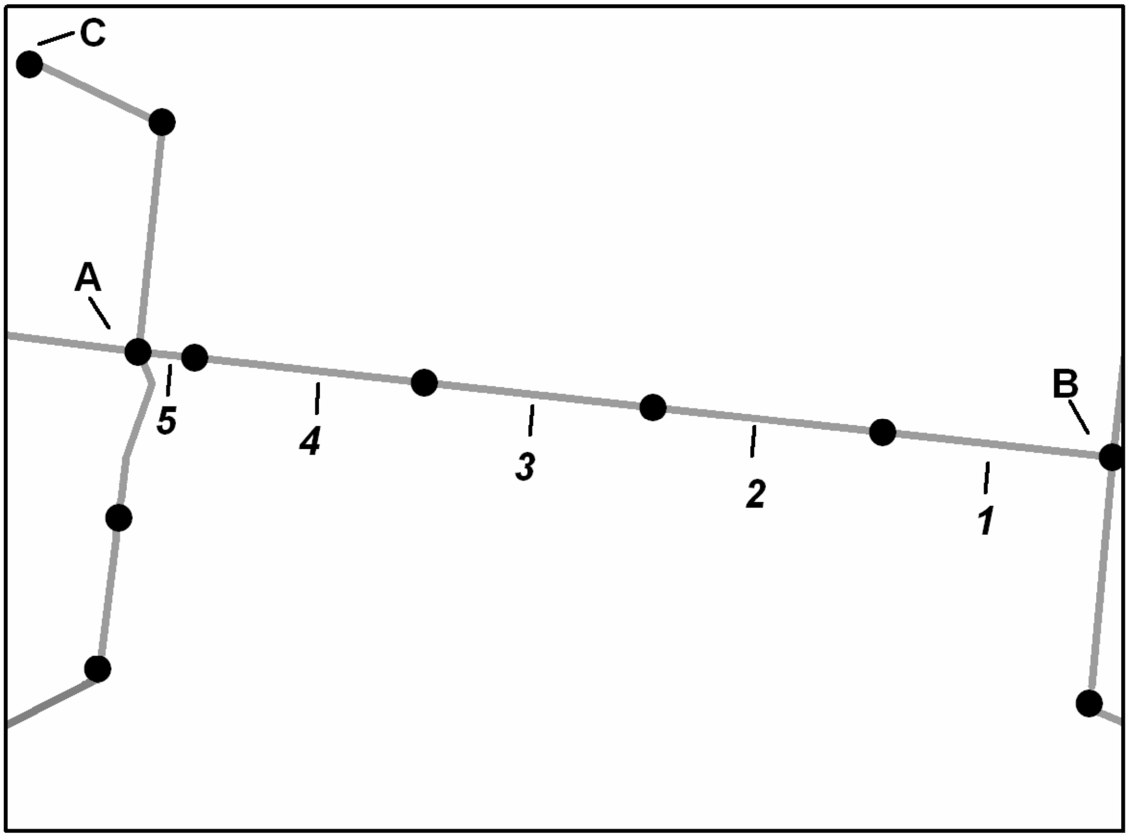


Figure 1. Illustration of the basic differences between the Planar KDE and Network KDE for the same point event dataset. To estimate the density value at a focal point x, the planar KDE treats the whole 2-D space as the context and finds 4 accident points (solid dots) within a search bandwidth *r*, whereas the Network KDE only finds 2 accidents within the same bandwidth in the network space based on network

distance.

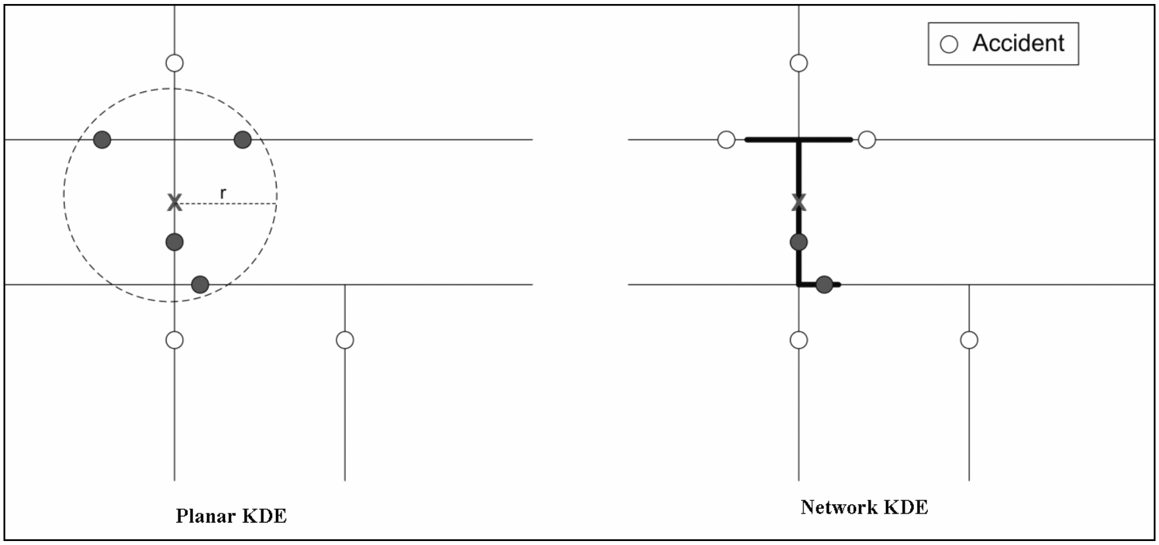


Figure 2. Illustration of the basic terms used in the proposed network KDE algorithm. The line segment between the two road intersections A and B is called a *segment*, so is the dangling line segment AC between a road intersection A and a road end point C. Each *segment* is divided into *lixel*s of a defined network length *l*. Here, five *lixel*s are created for *segment* AB, with *lixel*s 1, 2, 3 and 4 being regular *lixel*s with length *l*, and *lixel* 5 being the residual *lixel* with length less than *l*. The dark dots are the lxnodes, the intersection points of *lixel*s. Note the original road intersections (A, B) and end points (C) are always *lxnode*s.

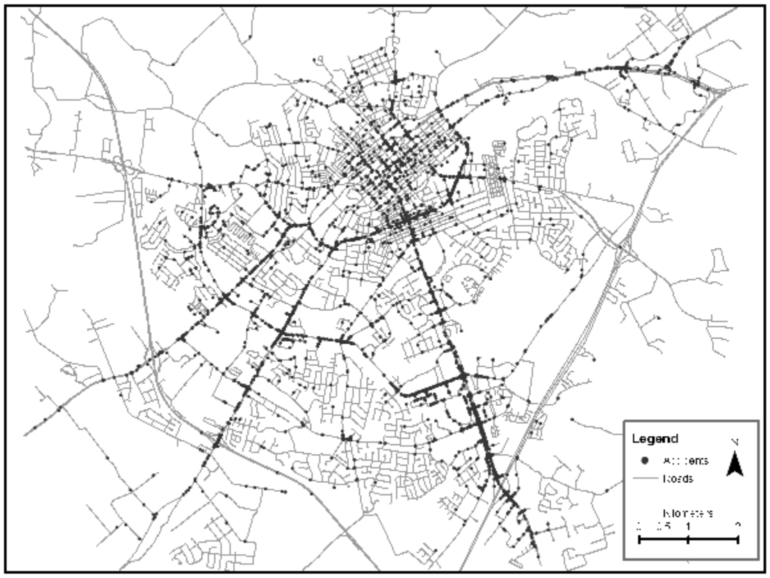
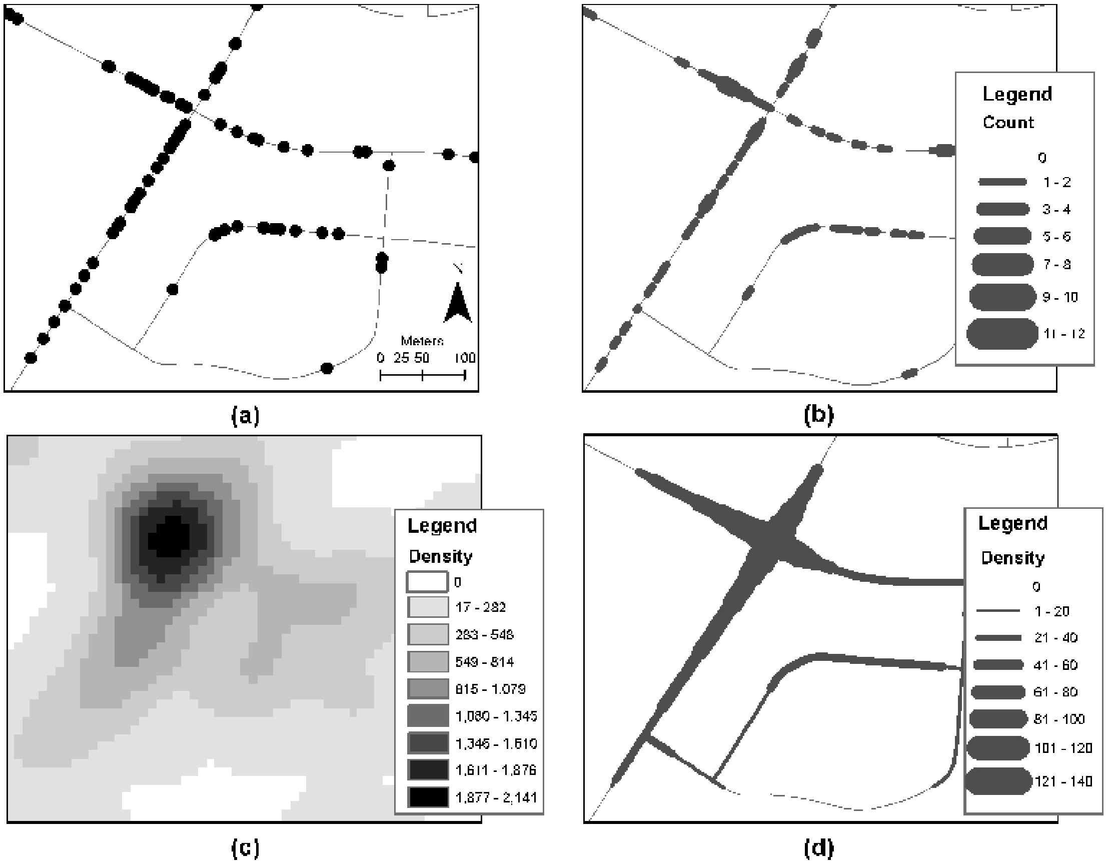


Figure 3. The study area, Bowling Green, Kentucky. Test dataset include a road transportation network system and traffic accident data for year 2005.



(a)

(b)

(c)

(d)

Figure 4. Illustration of different ways of presenting accident spatial patterns in a local part of the study area: (a) accident point locations, (b) number of accidents per 10-m *lixel*, (c) a standard planar KDE (10-m raster cell), (d) the proposed network KDE (10-m *lixel*). For both (c) and (d), a Gaussian kernel and a 100-m search bandwidth are used. Both KDE are more informative in presenting the density pattern, but only the network KDE estimates density in the event context, the network space.

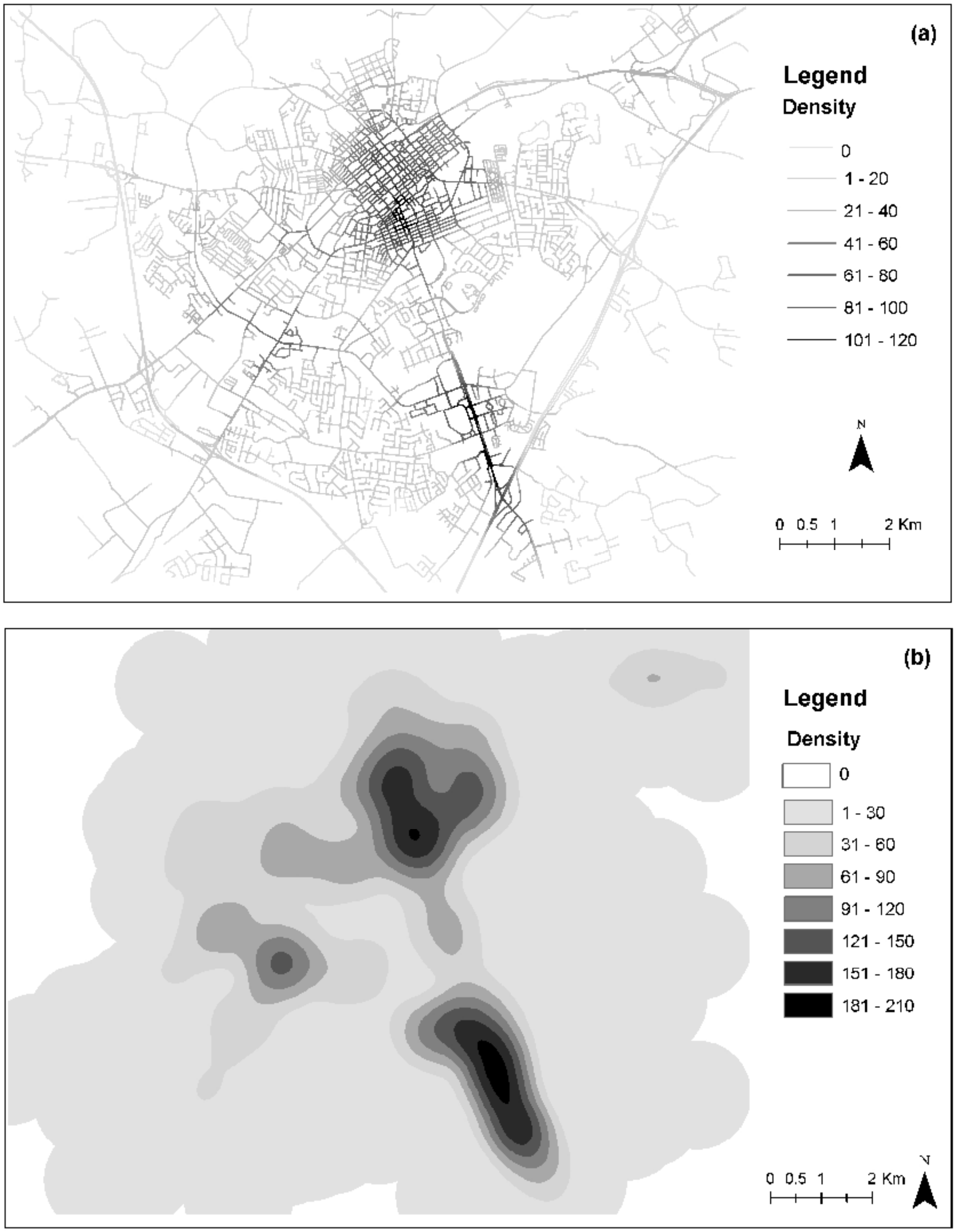


Figure 5. Illustration of different overall patterns produced by the network KDE and planar KDE in the study area: (a) the proposed network KDE (10-m *lixel*), and (b) a

standard planar KDE (10-m raster cell). A Gaussian kernel and a 1000-m search bandwidth are used.

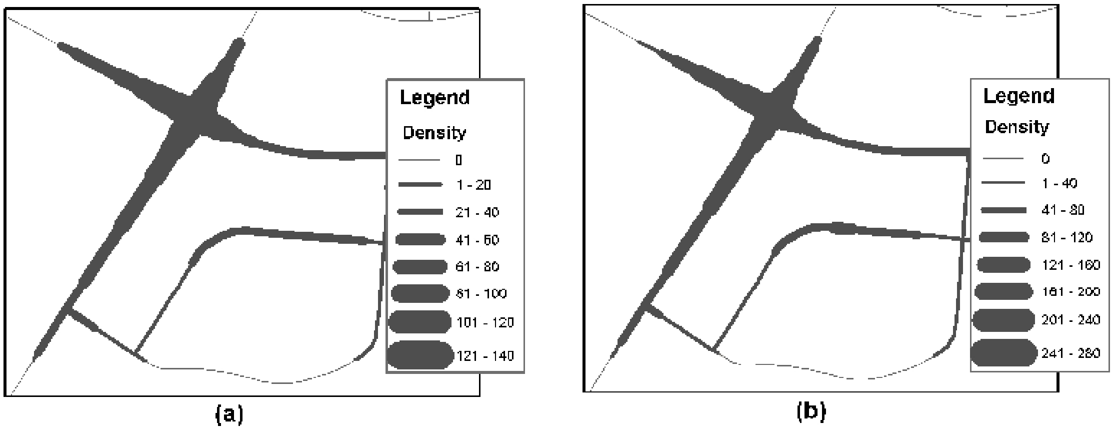


Figure 6. Illustration of the local impacts of kernel functions in the network KDE: (a) a Gaussian kernel based network density (per km), and (b) a Quartic kernel based network density (per km). For both (a) and (b), a 100-m search bandwidth and a 10-m *lixel* length are used. Both kernel functions result in similar local details.

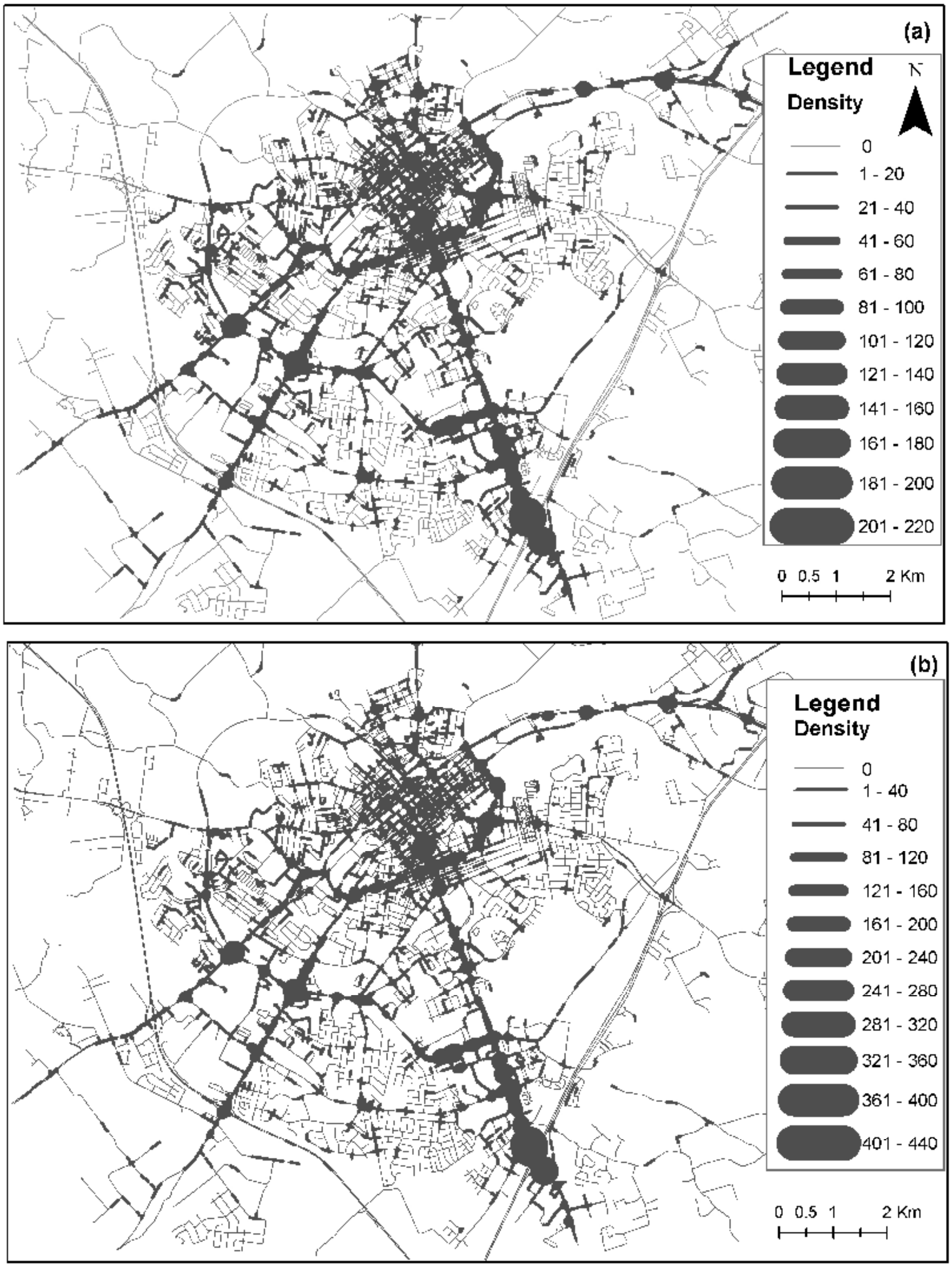


Figure 7. Illustration of the impacts of kernel functions on the overall density pattern in the network KDE: (a) a Gaussian kernel based network density (per km), and (b) a

Quartic kernel based network density (per km). It appears that the kernel functions make less difference in the overall pattern.

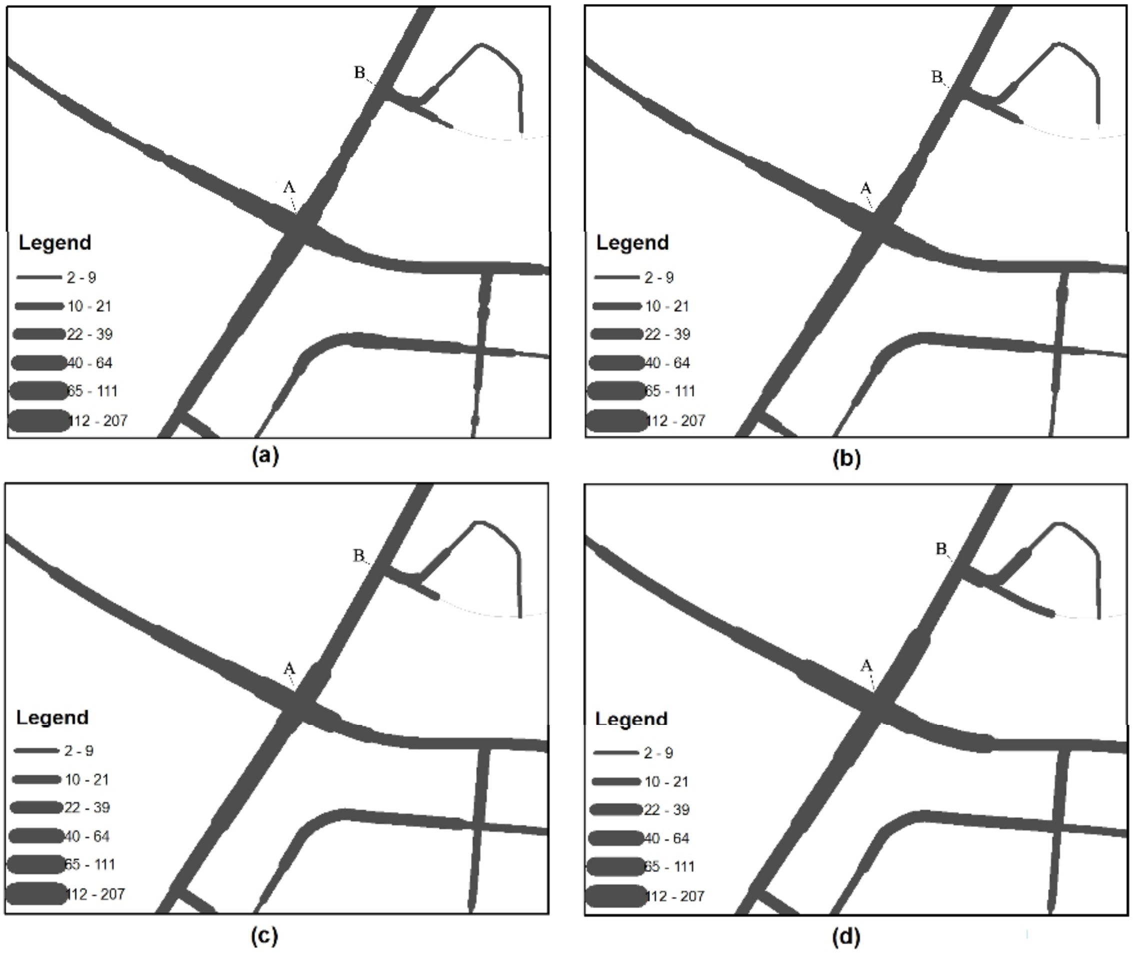


Figure 8. Illustration of the impacts of t *lixel* length on the calculated density values for the network KDE. (a)-(d) are the results for the *lixel* length of 5 m,10 m, 50 m and 100 m respectively, with a Gaussian kernel and a 100-m search bandwidth. The shorter the *lixel* length, the more detailed location variation can be revealed, as shown in the *segment* from A to B.

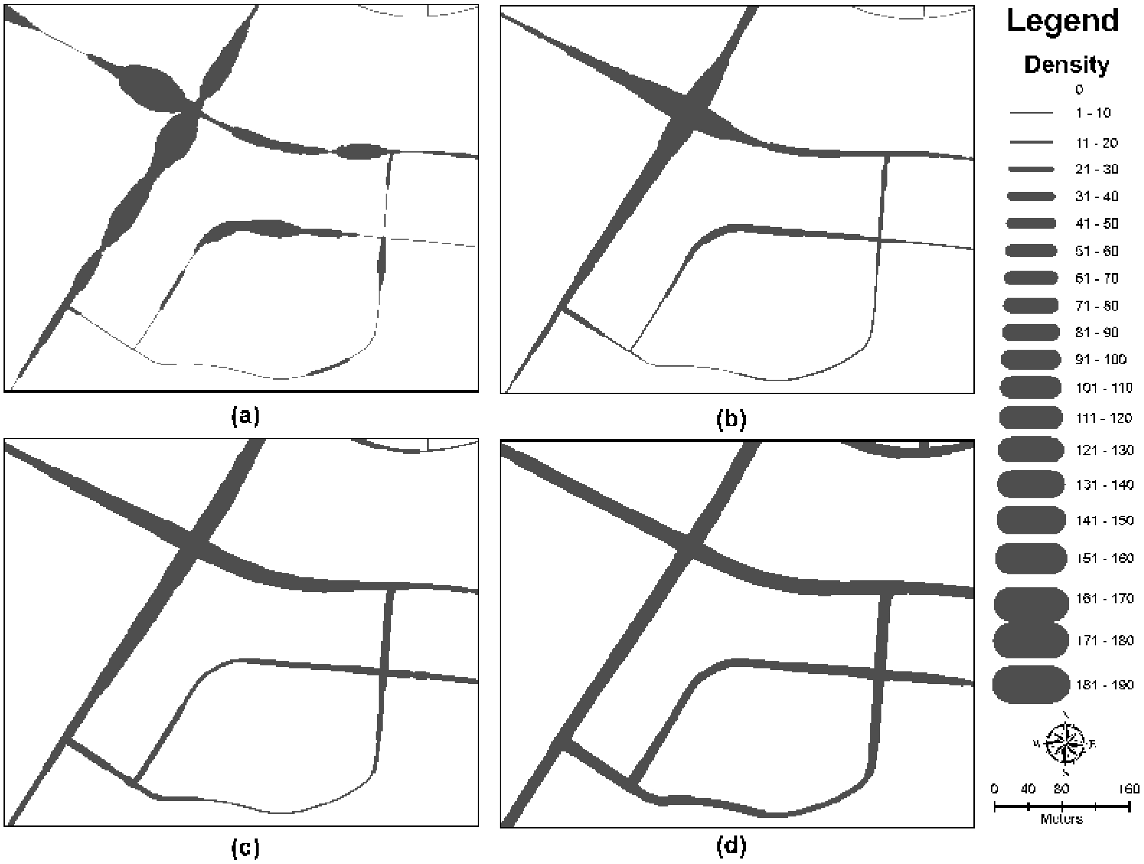


Figure 9. Illustration of the impacts of narrower search bandwidths on the calculated density values (per km) at local scales. (a)-(d) are the results for the search bandwidth of 20 m, 100 m, 250 m, and 500 m respectively, with a Gaussian kernel and a 10-m *lixel* length. The local variations of density gradually get lost with increased search bandwidths.

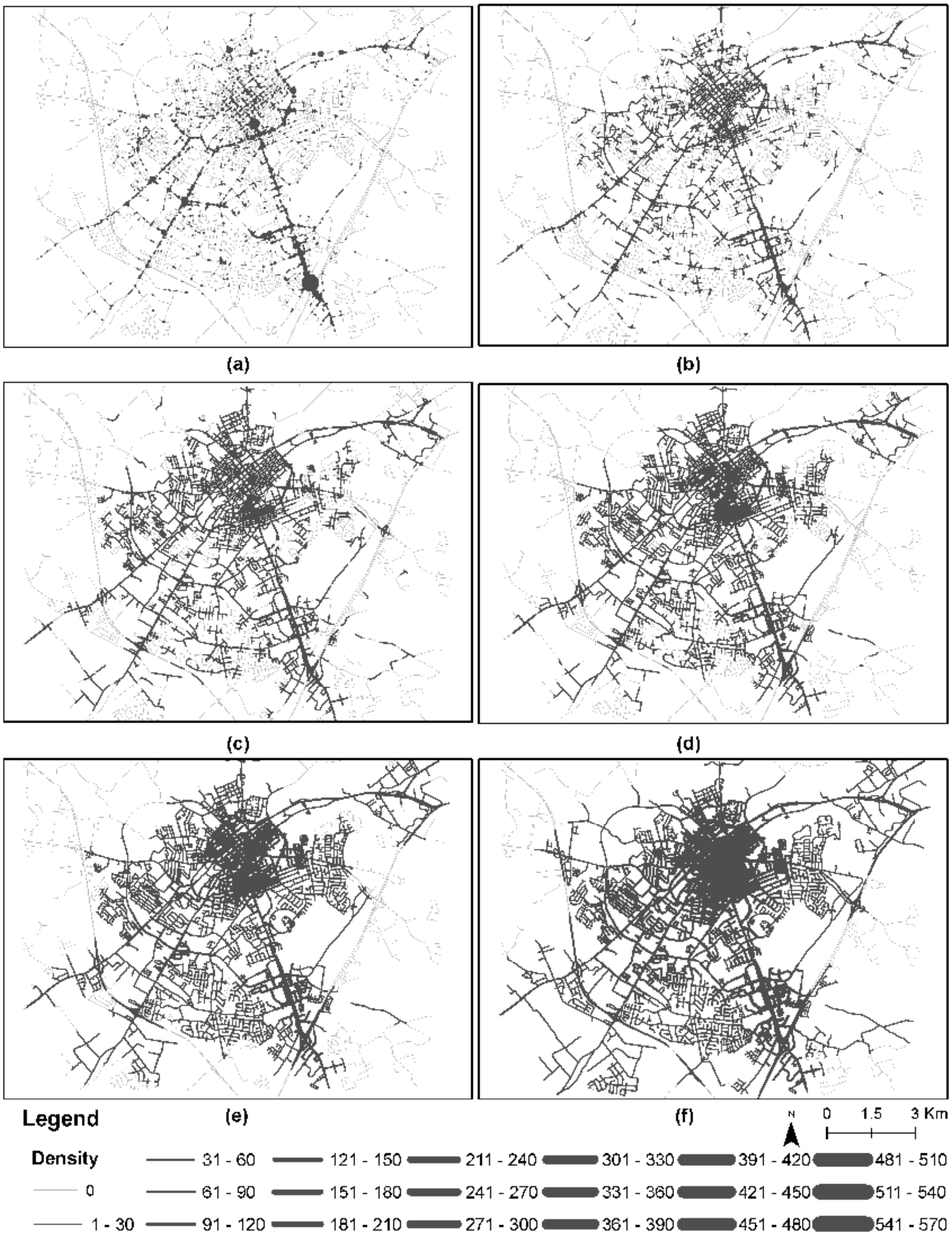


Figure 10. Illustration of the impacts of different search bandwidths on the overall density pattern. (a)-(f) are the results for the search bandwidth of 20 m, 100 m, 250 m, 500 m, 1000 m, and 2000 m respectively, with a Gaussian kernel and a 10-m *lixel* length.

As the search bandwidth increases from 20 m to 2000 m, the local "*hot spots*" are gradually combined with their neighbors and larger clusters appear.

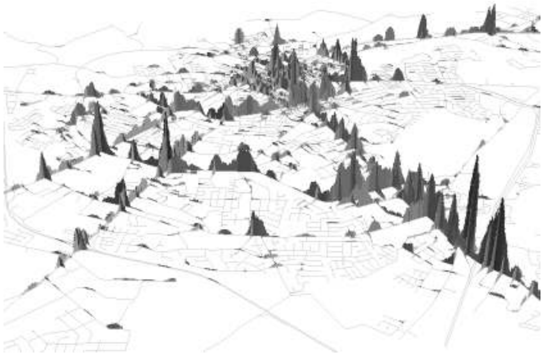


Figure 11. Three-Dimensional visualization of the spatial pattern of density estimated by the proposed network KDE (10-m *lixel* length, Gaussian kernel, 100-m search bandwidth), with ESRI ArcScene.