COMPLETING FRAGMENTARY RIVER NETWORKS VIA INDUCED TERRAIN

Tsz-Yam Lau and W. Randolph Franklin

Department of Computer Science
Rensselaer Polytechnic Institute
110 8th Street, Troy NY, USA
laut@cs.rpi.edu, wrf@ecse.rpi.edu

ABSTRACT:

Fragmentary river segments have to be joined appropriately before becoming useful to addressing transportation problems like route planning and pollutant tracking. We suggest the induced terrain approach to complete the job with high accuracy and efficiency. By first approximating a terrain compatible with the partial river segment and height information, and then deriving a river network from that induced terrain, we effectively enforce the restrictions imposed by the height observations. Natural neighbor interpolation with stream burning is capable of generating induced terrains that predict river locations accurately without adjustment for any parameters. Considerable time is saved from executing the global river derivation again and again to figure out the optimal parameter values, especially when we are confronted with increasingly massive terrain dataset. In the subsequent river derivation, we propose biasing in favor of the known river locations. Their water amounts are set to the critical value just sufficient for the locations to be regarded as river locations. In the final thinning process, these locations are simply set not to be trimmed. All known river locations are thus fully recovered at the end. We expect the same techniques can be applied to the recovery of some other 2D and 3D networks, like road networks and dendrite networks.

1 INTRODUCTION

Knowledge of the complete river network is essential to a number of geographical and environmental applications. For example, we need to know how the river segments are connected to design for the shortest route for a ship to travel from one place to another. We have to identify the exact segment connectivity and river locations before figuring out the areas that are likely to be affected by pollutants or flooding originating at some place in the terrain. However, that information is usually not available right away with conventional surveying techniques. With stereophoto maps (Kelly et al., 1977), satellite imaginary (Konceny and Lohmann, 1987) or even advanced technologies like Interferometric Synthetic Aperture Radar (IFSAR) (Pike, 2000), the presence of clouds and canopies often occlude parts of the river network (Asante and Maidment, 1999). Very often, the reliable data we can obtain are just a bunch of disconnected river segments, which may be accompanied by some height samples.

To apply these data to connectivity applications, one needs to estimate a complete river network based on these partial observations. By a complete river network we usually expect it is a fully-connected tree. Branches can be multiple-cell thick but very often we aim to figure out one-cell thick branches representing the middle lines of the rivers. Every river location is expected to have a single way for the water to get out to some sea shore or terrain edge (Asante and Maidment, 1999).

If we have the fragmentary river observations only (no height data are available), the only yet intuitive way to fill in the connection gap is to extend the segments (Asante and Maidment, 1999). Criteria like shortest connection distance or segment curvature preservation may be assumed to guide the process. However, the search space is still huge and numerous solutions are possible. In contrast, if some other terrain property, which is usually the set of elevations, is also known, we may rule out certain possibilities. For example, it is nearly impossible for a river segment on one side of the hill to meet another river segment on the other side.

Here we propose the induced terrain approach to this fragmentary river network completion problem. The approach involves first approximating a terrain surface which is compatible with the partial height and river location observations, followed by deriving a river network from the induced terrain. By that, we expect the derived river network to enforce most restrictions imposed by the partial observations.

This paper is organized as follows: In Section 2, we will go through a few concepts related to our discussion, including terrain reconstruction, hydrological correction scheme and river network derivation. In Section 3 where we discuss the first step of the approach, we present our survey on a number of possible strategies to reconstruct legal terrains, and argue why we propose using natural neighbor interpolation and stream burning (NN-SB) from both accuracy and efficiency perspectives in this particular application.

The induced terrain in the previous step, if passed to the conventional river derivation algorithm, will result in some already given partial river locations not being identified. To eliminate this, we propose albiasing towards those locations throughout the subsequent river derivation process. Details will be given in Section 4, before we suggest future directions in Section 5.

2 RELATED CONCEPTS

2.1 Terrain Reconstruction

The first step of the induced terrain approach is to complete the missing values of the elevation grid, because the river derivation step works with a complete elevation grid only.

Nearly all terrain reconstruction algorithms follow this first law of geography: everything is related to everything else, but near things are more related than distant things (Tobler, 1970). Essentially they set the elevation of an unknown position (i,j), $z_{i,j}$, to be

a weighted average of known elevations h_l where $l = 1, 2, \ldots, k$:

$$z_{i,j} = \sum_{l=0}^{k} w_{(i,j),l} h_l \qquad (1)$$

Proximity polygon (or *Voronoi polygon* or *nearest point*) sets $z_{i,j}$ to its nearest known neighbor, which means $w_{(i,j),l} = 1$ for the nearest known position but $w_{(i,j),l} = 0$ for all the others (Thiessen, 1911). Because the one neighbor to use for elevation estimation has been clearly defined, it requires no parameter and is therefore simple. However, the surface so generated is blocky because the value used as the interpolated height changes abruptly when crossing the Voronoi boundaries of the known elevations.

Incorporating multiple known heights in the calculation allows their influences to transit smoothly across the terrain, and is a way to improve the surface realism. *IDW (Inverse distance weighting)* sets $w_{(i,j),l}$ to the inverse power of distance between (i, j) and the known position l, usually square (Shepard, 1968). *Kriging* is a geostatistical approach, in which all control point data are involved in finding optimal values of the general weighting function $w(s)$ for a known point distant s from the unknown position. The main assumption here is that the covariance between two elevations depends solely on the distance between the positions (Krige, 1951). But we often have problems optimizing the number of points to use.

One method that takes multiple known heights yet requires no parameters is the natural neighbor interpolation (Barnett, 1981). The set of neighboring known heights together with their respective weights are often said to have been well-governed by Voronoi diagrams. Another way involves fitting splines in between the known heights. In this case, first-order and even second-order continuity are explicitly enforced, thereby ensuring the slope of the surface is smooth.

In all the approaches described above, we take the measured elevation values as is for the reconstructed surface, also known as *interpolation*. However, in most cases such interpolation of the known points is not necessary because of measurement imprecision. *Approximation*, which allows relaxation from the measured values, allows much more desirable overall reconstruction results and much smoother surfaces.

Trend surface analysis is a representative technique used for surface approximation. It involves specifying a general form of a mathematical function at the beginning. This is the trend which is expected to represent a large-scale systematic change that extends from one map edge to the other. Then we fit the function with the sample data aiming least squares, a process also known as regression. A review of the technique can be found from Wren (Wren, 1973). However, to model complicated surfaces, it has to be used with some other techniques.

ODETLAP (Overdetermined Laplacian Partial Differential Equation) (Gousie and Franklin, 2005, Xie et al., 2007), or *Inverse interpolation* (Claerbout and Fomel, 2010) sets up an overdetermined system to solve for the elevations of the whole terrain grid. The system includes an exact equation for each known-height position. That equation sets the height value of the respective position to its known value. The system also contains an averaging equation for every position. The equation attempts to regularize the respective height to the average of its immediate four neighbors. Through adjusting the weight R of the set of exact equations over the set of averaging equations, we obtain terrain surfaces of desirable accuracy-smoothness tradeoff. It can

work with contour lines (continuous or intermittently broken), infer mountain tops inside a ring of contours, and enforce continuity of slope across contours. All these are favorable features of natural-looking terrains.

2.2 Hydrological Correction Schemes

All the above are general terrain reconstruction techniques working primarily with partial height grids. To incorporate the known river locations in the reconstructed terrain, a few *hydrological correction schemes* have been developed. These schemes are originally intended for full digital elevation models (DEMs) accompanied with the respective full river networks, and they aim to improve the DEMs' ability to replicate hydrological patterns, especially in flat landscapes in which noise in the height data can harm the river derivation result drastically.

Most correction schemes are available as programs that take the output from the above general terrain interpolation algorithm and use the given river location data to 'correct' the terrain. Their common tactic is to sink the elevations of the identified river locations (as well as their neighborhoods in some advanced implementations). The simplest of them all is to trench only those river locations by a certain *trench amount*, a process commonly known as stream burning (Hutchinson, 1989). We expect the lowered positions are more likely to see water stop there, thus increasing their chance of becoming river locations. Some advanced algorithms suggest sinking the neighborhoods as well, but they require more parameter inputs. For instance, in `agree.aml` (Hellweger, R., 1997), an obvious parameter to be decided is the *sink width w*. (Meanwhile, its trench amount is split into *sharpdist* for river locations and *smoothdist* for their neighborhoods.)

The remaining schemes have terrain reconstruction and hydrology-adaptation bundled together as complete *hydrology-aware terrain construction algorithms*. The most famous among them all is ANUDEM (Hutchinson, 1989), in which iterative finite difference interpolation is interleaved with the drainage enforcement algorithm. The enforcement algorithm creates not only valleys at the river locations but also chains of decreasing elevations along the flow pathways to guide the water flow. Even without the stream network, the routine can still infer drainage lines via flow directions or even just partial elevations (through an analysis of grid points and saddle points). This practice potentially gives better results. However, this also means we need to recompute the whole elevation grid when data are updated, because now there is a tight topological relationship between consecutive locations along the river.

2.3 River network derivation

To derive the river network of a complete elevation grid, we use some *river derivation algorithm* like `r.watershed` (Ehlschlaeger, 2008) in GRASS GIS, or TERRAFLOW (Arge et al., 2003). The first key step of nearly these algorithms is to reduce the false local depressions due to noise in the elevation data. Otherwise, in the later drainage direction determination, we will come up with a false vast number of cells whose water has nowhere to go. An exception is `r.watershed` which uses least-cost search algorithm. Because the drainage of a cell is not determined until its downstream location is determined, false pits will not cause drainage flow pointers to go askew. In other words, the single water flow direction can be to some other deeper neighbor which may not be the lowest (Ehlschlaeger, 1989). But if we rely on some other single-flow direction approaches like the D-8 algorithm (or its 4-neighbor version D-4) in which water is directed to the lowest deeper immediate neighbor (O'Callaghan and Mark,

1984), or multiple-flow direction schemes in which water is allowed to flow to multiple neighbors as long as they are deeper, as in large-dataset-optimized TERRAFLOW, we cannot skip the process. After finding the drainage directions, we evaluate the number of cells draining into each position. And then based on the initial water amounts allocated to the cells, we produce an accumulation map. Finally, the river network compatible with the terrain can be extracted by excluding those positions with accumulation less than a certain threshold (Universita Degli Studi Di Trento, 2008). The one-cell thick, no-loop constraint, if needed, can be enforced by a thinning algorithm (Lam et al., 1992) afterwards.

3 PARAMETER-LESS TERRAIN RECONSTRUCTION AND HYDROLOGICAL CORRECTION

As seen in the previous section, a plethora of terrain reconstruction and hydrological correction schemes is around for use already. We need to find out which one works with the partial river network that we have here. Accuracy is, as always, an important concern when deciding which scheme to use. In this case, it is evaluated in terms of how well the subsequently derived river network links the river segments and classifies the locations as river/non-river.

However, we do need to take care of efficiency as well. While a local update on height/river location just means a height change in the neighborhood (a consequence of the first law of geography), it is not the case for the subsequent river derivation. We have to feed the whole completed elevation grid every time the grid changes for the new river configuration. It is justified as a river originating at a location can reach virtually any location in the grid no matter how far away it is. The implication to the terrain reconstruction and hydrological correction processes is that their operations should depend on as few parameters as possible (preferably none at all), or we are sure some universal parameter setting is highly likely to yield satisfactory results. Otherwise, we need to go through the global river derivation operation again and again to find out the optimal parameter values, which time and resources may not allow.

3.1 Natural neighbor with stream burning

Among all the general terrain interpolation algorithms, natural neighbor interpolation is the only known algorithm to-date that offers reasonably delicate results without any supply of parameter. So it turns out to be the general terrain reconstruction algorithm that we recommend. For the hydrological correction strategy, we adopt stream burning. First, it is reported to offer the best reproduction of the river locations (Callow et al., 2007), which we look for. Second, even though the method takes *trench amount* as the only parameter, one can treat it as virtually parameter-less. Arbitrary setting of the trenching amount is sufficient to see its effect. Further fine-tuning is unnecessary as the result is similar for different *trench amount* settings as long as it is sufficient (Callow et al., 2007). This is an advantage over the other more sophisticated approach that requires adjusting for optimal parameters, like *sink width* w for agree.aml.

3.2 Evaluation

We compare NN-SB with a few alternatives:

- **SF-SB: Second-order spline fitting with stream burning** (*trench amount = 30*). This is to investigate if some other parameter-less general terrain reconstruction scheme performs better)

- **OS-SB: Optimal-R overdetermined system with stream burning (trench amount = 30).** This is to see how better some other more sophisticated general terrain reconstruction schemes perform. We vary R from among 1, 2, 5, 10 and 20, and pick the setting that leads to the smallest false negative with respective to the given river locations. (False negative refers to the situation in which a location is told to be where river flows across, but fails to be identified as a river location in the reconstructed network.)

- **NN-AGREE: Natural neighbor with** agree.aml **under optimal** w (*sharpdist = 30*, *smoothdist = 30* so that when $w = 1$, it is equivalent to stream burning). This is to check how better can more complicated hydrological correction scheme perform. We vary w from among 1, 2, 5 and 10 and pick the one that gives smallest false negative with respective to the given river locations.

- **ANUDEM**. This is to see whether the existing integrated algorithm that is originally designed for full river networks also works well with partial river segments. We use the implementation named TOPOGRID in ARCGIS 9.3. To compare fairly with our scheme which requires virtually no adjustment of parameters, we do not alter any default parameter settings in TOPOGRID.

Our test dataset includes some six 400 × 400 DEMs shown in Figure 1. A bigger plot of *mtn1* is available in Figure 2 top.
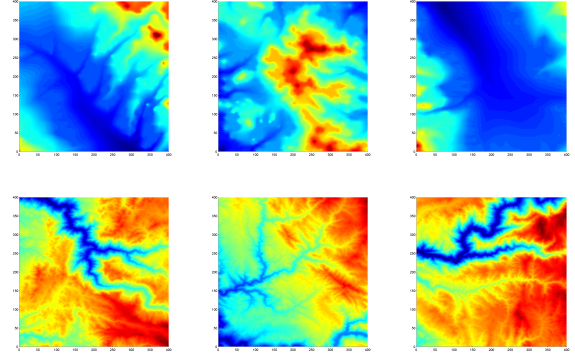

Figure 1: Test 400×400 DEMs: (first row) *hill1*, *hill2*, *hill3*, (second row) *mtn1*, *mtn2*, *mtn3*.

We first run r.watershed with accumulation cutoff threshold = 200, initial water amount at each location = 1 over these six DEMs to obtain the respective theoretical four-connected river network. (Figure 2 bottom shows the full river network of *mtn1*.) Next, we sample for the partial heights and river locations as follows: For river locations, first we divide the whole grid into 20 × 20 subgrids. In each subgrid, we randomly pick a point and mask an area of 12 × 12 around it. The height at every position is available with 10% probability. After that, we pass the partial heights and river locations data to the algorithms and obtain the reconstructed terrain surface.

We are interested in how well the river segments are correctly reconnected. Visual comparison of the reconstructed networks with the original one is the most intuitive approach to evaluate for that. Meanwhile, we also compute the rate that the reconstructed terrains correctly classify a location as a river/non-river site to quantify how well the algorithms recover river locations.

The river locations offered to the reconstruction algorithms are shown in Figure 3 top left. The other diagrams in the same figure
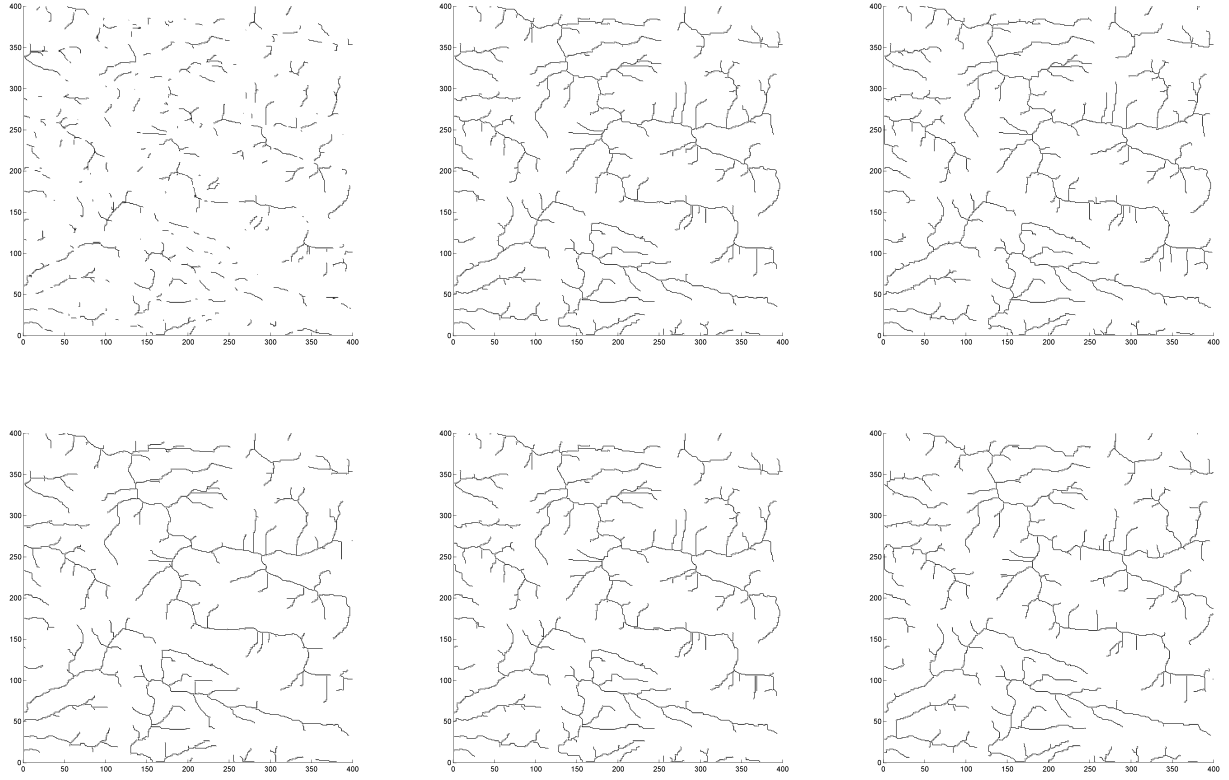
Figure 3: The given partial river network of *mtn1* (top left). Reconstructed river networks with different algorithms, NN-SB (top middle), SF-SB (top right), OS-SB (bottom left), NN-AGREE (bottom middle), ANUDEM (bottom right).
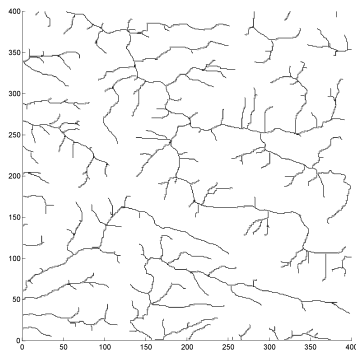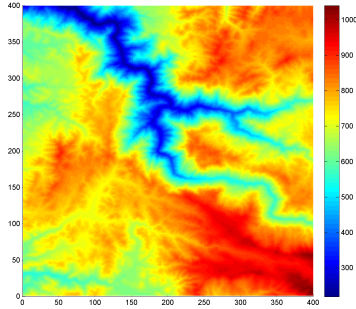

Figure 2: Full *mtn1* data. Elevation (left), river network (right).

shows the reconstructed river network of corresponding to `mtn1` of different algorithms. All of them appear to do a good job in reproducing the connections.

Data	NN-SB	SF-SB	OS-SB	NN-AGREE	ANUDEM
hill1	1.94	2.38	1.76	2.00	4.17
			($R=1$)	($w=2$)	
hill2	2.14	2.17	1.93	2.11	3.76
			($R=10$)	($w=2$)	
hill3	3.19	4.54	3.46	3.28	6.45
			($R=5$)	($w=2$)	
mtn1	2.48	2.88	2.47	2.44	3.82
			($R=20$)	($w=5$)	
mtn2	2.42	2.53	1.90	2.29	3.43
			($R=20$)	($w=5$)	
mtn3	2.50	2.51	2.02	2.35	3.40
			($R=10$)	($w=5$)	

Table 1: River/Non-river classification errors for all locations of different algorithms. The parameters needed for the optimal results are in brackets.

Table 1 lists the correct classification rates. When comparing within the virtually parameter-less schemes, NN-SB consistently presents smaller errors than SF-SB. This means NN makes a better choice on what neighbor to be picked and the respective weights to be used than SF. When compared with algorithms which require parameter inputs, NN-SB occasionally performs a little bit poorer, especially with OS-SB which we have discussed to have a few advantages in building natural-looking terrain surface. However, the expense is we have to running the global river derivation algorithms again and again for the optimal parameter settings. Such reruns cannot be skipped as there exists no one-size-fit-all parameter value that consistently gives superior results over dif-

ferent datasets. For example, with NN-AGREE, setting $w = 1$ (which is equivalent to NN-SB) sometimes offers superior results (See *hill1* and *hill3*). Together with the fact that the gain is usually within 1 percentage point, we recommend the simpler NN-SB scheme. ANUDEM does not perform well. We suspect the reason is that the topological relationship buildup that it features to improve accuracy does not work well with broken river segments.

4 BIASED RIVER DERIVATION

The result above is yet a distance from completing the fragmentary river networks. In particular, we fail to reproduce every identified river location. Our second technique aims to eliminate this situation within the river derivation procedure.

4.1 Biased initial water assignment

The middle diagram of Figure 4 highlights those identified river locations that fail to show up in the reconstructed river network with *mtn1*.

We observe that a major source of such false negatives is the set of the side streams which mark the onset of rivers. They fail to be parts of the river network again because insufficient water gets through them in the reconstructed terrain. As the terrain is already the best that we can reconstruct and there can hardly be any more improvement, we attempt to make river flow in those problematic locations in a different way: we simply assign them an initial water amount that is equal to the critical amount to be claimed as river locations. As a result, they can stay after the accumulation thresholding step even if no other water flows into these locations. Our justification for that biased water allocation is as follows: there can be many different reasons for a cell to get enough water to be a river location. It may collect sufficient water from other places, or it may have a considerable amount of reserve by itself (for example from the sky).

As a result of this biased initial water assignment, the identified river locations are guaranteed not to be excluded after the thresholding procedure because the amounts of water flowed through them cannot be smaller than the initial volumes assigned to them.

Such a change does not incur much additional work to the river derivation algorithm. For example, with `r.watershed`, one is ready to specify the initial water amounts of individual locations as a matrix parameter.

4.2 Biased thinning

To guarantee the final river network fulfills the thickness upper bound (for instance, if the network is vectorized later, it has to be at most one-cell thick throughout.), we often apply thinning (Lam et al., 1992) as a post-processing step: pixels at the outer layers of the object are successively removed, while retaining any pixel whose removal would alter the connectivity or shorten the legs of the skeleton (Jang and Chin, 1990). For a well-implemented conventional thinning algorithm, its thinning result should not be biased: the kept pixels are located at the geographical center of the object, where the identified river locations may not be.

To avoid ruining our previous effort, we impose an additional constraint on the thinning procedure: we simply disallow any removal of the known river locations in any case. Without any prior knowledge, it is natural to assume the geographical center is where the central lines lie. But now we do know where parts of the river are located. It is justifiable to privilege those identified

river locations when deciding whether to keep the cells or not in the thinning process.

Such a small change, again, does not introduce too much overhead. In our prototype, we adapt a mask-based thinning algorithm (Diaz de Leon et al., 2004) for biased thinning. The gain in running time is insignificant.

4.3 Evaluation

The right diagram of Figure 4 shows the river derivation results of *mtn1* using these biasing techniques. One can see that not only the entire given river locations but also their connections to the main streams (an example: the river segment at the bottom left corner) are identified as river locations. Table 2 presents the respective error rates. All the false negatives with respect to given partial river locations fall to zero, leading to a drop in overall classification errors.

Data	Before	After
hill1	1.97 (9.51)	1.77 (0.00)
hill2	2.14 (12.12)	1.89 (0.00)
hill3	3.19 (24.37)	2.82 (0.00)
mtn1	2.48 (10.33)	2.21 (0.00)
mtn2	2.42 (11.35)	2.13 (0.00)
mtn3	2.50 (11.46)	2.25 (0.00)

Table 2: River/Non-river classification errors for all locations of different algorithms. The false negatives with respect to given river locations are bracketed.

5 CONCLUSION AND FUTURE WORK

We have presented a simple yet complete framework for completing fragmentary river networks through the induced-terrain approach. The two steps, namely hydrology-aware terrain reconstruction and river derivation, are designed with honoring the given identified river locations as the central principle.

In the terrain reconstruction process, this doctrine is realized by modeling the identified river locations as local minima with the virtually parameter-less stream burning. Together with parameter-less natural neighbor as the accompanied general terrain reconstruction scheme, we achieve competitive results in terms of river location classification, even when compared with other sophisticated alternatives requiring adjustment of parameters. Decoupling terrain reconstruction from the global river derivation saves us considerable time and resources from running the latter just for the optimal parameter values of the former. Such a saving is especially crucial when we are confronted with increasingly massive terrain datasets.

In the subsequent river derivation process, this goal is achieved via allocating critical initial amounts of water to the given known river locations, and preserving them in any case during the final thinning process. As a result, not only the entire given river locations are retained but also their connections to the main streams are recovered.

Having seen the success of this set of techniques with completing hydrology networks, we are anxious to port the same solution framework to complete some other 2D networks like road networks, and extend it to solve 3D network problems like fragmentary dendrite networks.
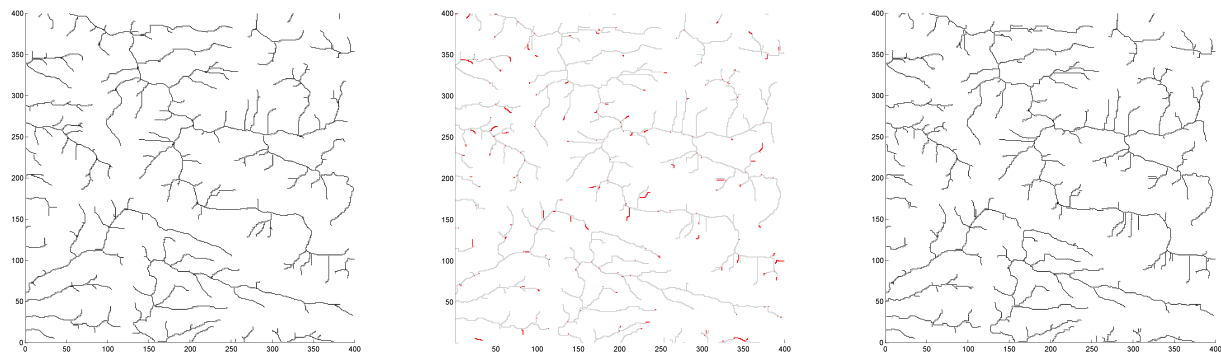
6 ACKNOWLEDGMENTS

Figure 4: Full river network of *mtn1* (left). Identified river locations not recovered when initial water assignment is unbiased is marked in dark red. The river network recovered from the induced terrain is highlighted in light grey (middle). River network derived by our biased river derivation scheme (right).

REFERENCES

Arge, L., Chase, J. S., Halpin, P., Toma, L., Vitter, J. S., Urban, D. and Wickremesinghe, R., 2003. Efficient flow computation on massive grid terrain datasets. Geoinformatica 7(4), pp. 283–313.

Asante, K. and Maidment, D., 1999. Creatng a river network from the arcs in the digital chart of the world. `http://www.ce.utexas.edu/prof/maidment/grad/ asante/dcw/rivernet.htm`, (retrieved Jun 23, 2010).

Barnett, V., 1981. Interpreting Multivariate Data. John Wiley & Sons, New York, chapter 2, pp. 21–36.

Callow, J. N., Niel, K. P. V. and Boggs, G. S., 2007. How does modifying a DEM to reflect known hydrology affect subsequent terrain analysis? Journal of Hydrology 332, pp. 30–39.

Claerbout, J. F. and Fomel, S., 2010. Image estimation by example: geophysical soundings image construction. `http://sepwww.stanford.edu/sep/prof/gee2. 2010.pdf`, (retrieved Jun 23, 2010), chapter 3, pp. 85–108.

Diaz de Leon, J. L., Yanez, C. and Giovanni, G., 2004. Thinning algorithm to generate k-connected skeletons. In: A. Sanfeliu, J. F. Martinez Trinidad and J. A. Carrasco Ochoa (eds), Progress in Pattern Recognition, Image Analysis and Applications, Lecture Notes in Computer Science, Vol. 3287, Springer Berlin / Heidelberg, pp. 115–135.

Ehlschlaeger, C., 2008. GRASS GIS: r.watershed. `http: //grass.itc.it/gdp/html_grass63/r.watershed.html`, (retrieved Jun 23, 2010).

Ehlschlaeger, C. R., 1989. Using the A^T search algorithm to develop hydrologic models from digital elevation data. In: Proceeding of the International Geographic Information System (IGIS) Symposium, pp. 275–281.

Gousie, M. B. and Franklin, W. R., 2005. Augmenting grid-based contours to improve thin plate DEM generation. Photogrammetric Engineering & Remote Sensing 71(1), pp. 69–79.

Hellweger, R., 1997. Agree.aml. Center for Research in Water Resource, The University of Texas at Astin, Austin, TX.

Hutchinson, M. F., 1989. A new procedure for gridding elevation and stream line data with automatic removal of spurious pits. Journal of Hydrology 106, pp. 211–232.

Jang, B. and Chin, R., 1990. Analysis of thinning algorithms using mathematical morphology. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, pp. 541–551.

Kelly, R., McConnell, P. and Mildberger, S., 1977. The Gestalt photomapping system. Photogrammetric Engineering and Remote Sensing 43(11), pp. 1407–1417.

Konceny, G. and Lohmann, P., 1987. Evaluation of SPOT imagery on analytical photogrammetric instruments. Photogrammetric Engineering and Remote Sensing 53(9), pp. 1223–1230.

Krige, D. G., 1951. A statistical approach to some mine valuations and allied problems at the Witwatersrand. Master's thesis, University of Witwatersrand.

Lam, L., Lee, S. and Suen, C., 1992. Thinning methodologies - a comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 14, pp. 869–885.

O'Callaghan, J. F. and Mark, D. M., 1984. The extraction of drainage networks from digital elevation data. Computer Vision, Graphics, and Image Processing 28, pp. 323–344.

Pike, J., 2000. Interferometric synthetic aperture radar - elevation (IFSARE). `http://www.fas.org/irp/program/collect/ ifsare.htm`, (retrieved Jun 23, 2010).

Shepard, D., 1968. A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 1968 ACM National Conference, pp. 517–524.

Thiessen, A. H., 1911. Precipitation averages for large areas. Monthly Weather Review 39(7), pp. 1082–1084.

Tobler, W., 1970. A computer movie simulating urban growth in the Detroit region. Economic Geography 46(2), pp. 234–240.

Universita Degli Studi Di Trento, 2008. GRASS tutorial - watershed analysis. `http://www.ing.unitn.it/~grass/ docs/tutorial_62_en/htdocs/esercitazione/dtm/ dtm4.html`, (retrieved Jun 23, 2010).

Wren, A. E., 1973. Precipitation averages for large areas. Canadian Journal of Exploration Geophysics 9(1), pp. 39–44.

Xie, Z., Andrade, M. A., Franklin, W. R., Cutler, B., Inanc, M., Tracy, D. M. and Muckell, J., 2007. Approximating terrain with over-determined Laplacian PDEs. In: 17th Fall Workshop on Computational Geometry, IBM TJ Watson Research Center, Hawthorne NY. poster session, no formal proceedings.