An Application of B-spline Reliefs to Render DICOM Data

Les A. Piegl[1] and William L. Mondy[2]

[1]University of South Florida, lespiegl@usf.edu
[2]University of South Florida, wmondy@usf.edu

Corresponding author: Les A. Piegl, lespiegl@usf.edu

Abstract. This paper aims to initiate research in rendering DICOM data originating from various image acquisition devices such as MRI, CT, and X-ray. The paper argues that while the DICOM data is rich in content, current image-based methods lose a lot of information that could be critical in the medical practice. This paper introduces B-spline reliefs; B-spline surfaces fitted to terrain-type data obtained from the DICOM database to initiate additional research. While the presented examples clearly illustrate that visualizing the DICOM data leaves lots of details out (along with introducing visual artifacts), the reliefs make use of all the data presented in the DICOM file. In contrast, the visual inspection utilizes as little as 1% of the data that the image acquisition device can return.

Keywords: medical imaging, image analysis, DICOM data, B-splines, surfaces.
DOI: https://doi.org/10.14733/cadaps.2022.838-853

1 INTRODUCTION

Image-based medical diagnosis follows a simple process. First, the data is acquired by a body scanner that measures the linear attenuation of rays passing through the body. The measure that is widely used is the Hounsfield unit (HU) of radiodensity [9]. Typical scanners, such as a CT scanner, can return about 4,000 levels of HU. Once the scanning is done, the raw numbers and other patient-related information are placed in a DICOM (Digital Images and Communications in Medicine) file. The image intensities in the DICOM file are displayed on a computer monitor or on a film for the physician to study. So, the pipeline is simply this: scanner – DICOM – display – eyes. This seems like a reasonable process and is being used routinely all over the world. Unfortunately, they may not show enough information to render a proper diagnosis because, as will be clear later, the rendered images reveal only a tiny percentage of the information in the DICOM file.

We will show examples below that the images, while they look perfectly functional, miss critical information necessary for proper diagnosis. For example, we know from experience that internal bleeding is hard to detect from either ultrasound or even CT images. Repeated imaging of patients complaining of internal bleeding revealed no apparent case of bleeding and were treated with pain killers. Internal bleeding and other critical conditions can be fatal if not treated timely and require a higher accuracy level than what the eyes can see.

There have been plenty of papers regarding DICOM data visualization in medicine. This paper is not intended to survey the field; however, a few papers are relevant for this research (an excellent survey on available software packages is found in [5]). The papers can be grouped into three main categories: (1) visualization toolkits [3], [4], [6], (2) 3D reconstruction [1], [2], [7], [11], and (3) multidimensional views with possible overlays [10], [12]. The literature shows that all methods use images either for visualization or 3D reconstruction (voxel and surface-based). The rich content from the DICOM file is lost or compressed to the capabilities of imaging devices.

2 THE EYES

The problem with our eyes is having them. This may sound unkind, but these wonderful organs that allow us to experience the beauty of the world have severe limitations. Some of the limitations are welcome, and others are troublesome. One of the welcome limitations is their inability to distinguish points at a distance. It is welcome because it gave us color graphics; we place three dots, red, green, and blue, next to each other, and the eyes, not being able to distinguish them, fuse them into a color sensation.

The second welcome limitation is that it cannot distinguish fast-moving images. This gave us motion pictures, where images that are moving faster than about 30 times a second are fused together into moving objects.

On the unwelcome side, we have problems with color. Each of us perceives different color sensations (in technical jargon, each person has a different color gamut), and many of us are just color blind.

Then there are the optical illusions; one of the most well-known ones is the Mach band effect, named after physicist Ernst Mach. It is based on the lateral inhibition of the eye; when bright and dark areas meet, the bright appears to be brighter and the dark darker. This happens quite often in medical images where bone structures meet other tissue; the boundaries can appear brighter (and thicker).

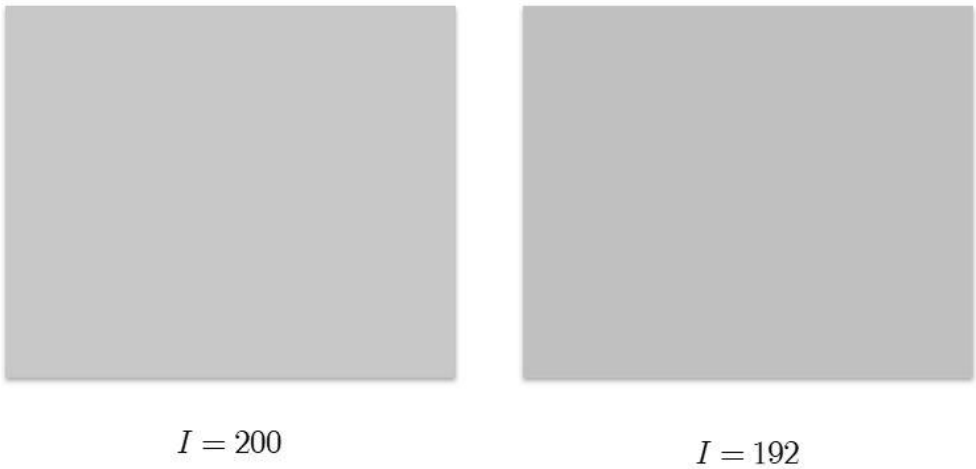


Figure 1: The eye's inability to distinguish various shades of gray.

The third limitation, and most relevant to this paper, is that the eye can only see a certain number of shades in a given color spectrum. Figure 1 shows that for the eyes to distinguish two gray boxes clearly, we may have to increase the intensity by as much as eight levels. This means that out of 256 shades of gray, we can see only about 32! This also means that our eyes are able to see just about 1% of the over 4,000 levels of intensities that the CT scanner returns.

While our eyes are remarkable instruments, it is reasonable to assume that most people, including radiologists, can only see about 30-40, maybe 50 shades of gray. This also means that the information about the tissues is severely compressed when inspecting medical images. In turn this also means that the perceived images do not necessarily reflect the reality inside the body.

287	268	282	179	25	44	10	30	18	14	10	11	14	12
456	275	261	231	108	15	38	6	28	12	20	3	20	14
663	382	292	287	205	52	45	30	19	19	11	17	13	20
789	519	375	351	291	137	11	47	1	31	8	20	14	5
848	595	410	367	342	222	55	35	17	25	20	14	23	6
920	659	418	347	358	291	128	15	33	7	24	6	15	3
1024	813	518	368	364	336	196	33	38	10	25	11	12	7
1088	1011	702	440	361	355	263	96	21	22	18	20	9	18
1007	1092	859	533	360	345	316	181	26	33	2	25	7	16
784	1017	963	683	425	337	332	252	93	22	26	18	26	2
602	906	1040	884	584	378	322	285	165	25	40	12	28	15
587	824	1033	1012	754	468	327	292	223	84	31	39	10	26
618	707	895	991	854	582	379	311	265	146	7	52	17	21
544	538	708	906	918	721	485	352	291	197	61	30	31	6
441	424	592	853	983	868	613	405	307	240	129	14	31	11
434	415	544	803	1001	963	719	465	335	278	186	54	27	18
471	427	480	691	931	991	815	555	384	310	227	95	12	30
460	403	380	527	804	995	933	685	454	339	273	162	32	34
434	397	320	384	656	955	1028	832	558	393	325	235	91	23
423	432	343	321	512	833	1029	948	698	483	373	281	141	6
375	433	389	327	414	682	951	1007	830	587	421	316	196	58
284	368	392	348	372	574	861	1012	915	677	475	362	261	124
247	304	368	356	345	480	755	970	950	736	515	393	314	189
315	301	360	368	324	375	604	865	933	764	534	405	352	257
412	338	371	400	349	317	447	700	861	791	589	437	376	303
440	364	383	440	418	332	342	532	771	849	718	518	384	303
400	358	387	469	475	365	289	419	710	916	866	636	425	318

Figure 2: Raw DICOM data.

Figure 2 shows a small excerpt from a DICOM file. The numbers do not change smoothly; in fact, the entire database is very noisy, reflecting, quite correctly, the complexity of human tissue. It may also suffer from a sampling problem, i.e., the tissues are sampled sparsely to avoid a high radiation amount. Now, compressing this data down to 256 shades of gray would do away with all the noise. Looking at the images with our eyes would compress this even further. Unfortunately, this would also eliminate a lot of helpful information. So, the real question is: can we render this data file without losing the information it carries and yet make it worthwhile? This paper proposes a method that can shed some light on some of the issues raised above.

3 B-SPLINE RELIEFS

This section will introduce B-spline reliefs, a particular form of B-spline surfaces [8]. B-spline surfaces, and their rational counterparts, called NURBS, are mathematical functions that can describe just about any free-form entity. For the sake of better understanding, a few B-spline formulas are reviewed below.

A B-spline surface of degrees (p,q) is defined as follows [8]:

$$S(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u)N_{j,q}(v)P_{i,j}$$

where $P_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})$ are the control points, and $N_{i,p}(u), N_{j,q}(v)$ are the normalized B-spline basis functions defined over the knot vectors

$$U = \left\{ \underbrace{u_0 = \cdots u_p}_{p+1}, u_{p+1}, \cdots, u_n, \underbrace{u_{r-p} = \cdots = u_r}_{p+1} \right\} \quad r = n + p + 1$$

$$V = \left\{ \underbrace{v_0 = \cdots v_q}_{q+1}, v_{q+1}, \cdots, v_m, \underbrace{v_{s-q} = \cdots = v_s}_{q+1} \right\} \quad s = m + q + 1$$

B-spline surfaces can be constructed in many ways, and the two most common ones are interpolating or approximating a set of points. In this paper, we used approximation to within tolerance to capture as much detail from the DICOM file as there is to capture. The point set was defined as follows:

$$Q_{i.j} = (x_{i,j}, y_{i,j}, h_{i,j}),\ i = 0, \cdots, k;\ j = 0, \cdots, l$$

where $(x_{i,j}, y_{i,j})$ represent a grid of points on the plane, and $h_{i,j} \approx I_{i,j}$. For each grid point, there is an elevation proportional to the intensity found in the DICOM file. To solve the approximation problem, the following details need to be worked out:

- Data smoothing, i.e., eliminating the noise in the DICOM file.
- Segmenting, or decomposing, the rows/columns of data to determine the degrees of freedom (number of control points and knots) needed to achieve the required accuracy.
- Fitting rows (or columns) of data with B-spline curves, followed by fitting the control points of the fitted curves to obtain the surface. This is a lofting-type algorithm that creates a surface from a set of curves.
- Checking for accuracy to make sure that the fitted surface faithfully represents the DICOM data. If the accuracy requirement is not met, the degrees of freedom need to be increased, and the surface needs to be recomputed.

Let us start with data smoothing.

3.1 Data Smoothing

To lessen the noise in the DICOM file, smoothing can be applied. The corner-cutting method below can be used in either the x- or the y-direction or alternated between them. The algorithm in the y-direction is as follows:

for $k = 1$ to *lev*
  for $i = 0$ to n
    for $j = 0$ to m

$$Q_{i,j}^{k} \leftarrow \frac{1}{4} Q_{i,j-1}^{k-1} + \frac{1}{2} Q_{i,j}^{k-1} + \frac{1}{4} Q_{i,j+1}^{k-1}$$

    end
  end
end

Fitting surfaces with or without smoothing can produce significant variations in accuracy as well as in appearance, especially if the value of *lev* is too high. So, while smoothing does a great job at eliminating the noise, it also eliminates (smooths out) details from the original data.

3.2 Data Decomposition

The DICOM data file represents a matrix of intensities that we turn into terrain-type data. The fitting operates on rows (or columns) of this data by fitting curves to the data points. While B-

spline (least-squares) approximation is a relatively simple task, approximating to within tolerance is still a challenging problem. To obtain the minimum number of control points while maintaining accuracy and proper parametrization requires the following:

- the number of control points and knots; and
- the positions of the knots as derived from an appropriate parametrization.

So, for each row (or column) of data points, we need to segment or decompose these points into a set of points so that each set represents a simple shape, easily fit by a B-spline segment. That is, in high curvature areas, the algorithm should put smaller segments, whereas low curvature areas can be approximated with longer running curve pieces.

More precisely stated, given a row of data points $Q_{a,j} = (x_{a,j}, y_{a,j}, h_{a,j})$, $j = 0, \cdots, l$, we look for the parameters $v_0, \cdots, v_l$ where the points are assumed, and the knot vector V. This process is repeated for every row of data points for $a = 0, \cdots, k$. Once the curves are fitted for each row, a surface skin is pulled over these curves via a process called lofting [8]. We need to determine the parametrization, the number of control points for each row, and the appropriate knot vector. Even though the DICOM data is given over a uniform grid, because of the elevations that are proportional to the intensity, the 3D data is not uniform. Therefore, uniform parametrization is not appropriate here, and hence we use the chordal length method instead.

Determining the degrees of freedom (number of control points) and the knot vector is more challenging. Here we utilize the fact that each B-spline curve is a collection of Bezier segments, stitched together with some level of continuity. So, our idea in determining a reasonable segmentation of the data points is to fit a set of Bezier curves to the data so that they approximate the points to within the required tolerance. These curves will not be joined smoothly as they are fitted individually. However, we are not seeking a smooth piecewise Bezier approximation; we are after the degrees of freedom and the locations of the junctions that are inherent in the Bezier decomposition. A simple method that finds the longest-running Bezier segment from a start index to an end index is described as follows:

- Fit the curve to a data set and check the error.
- If we are within tolerance, check if we can use more points.
- Otherwise, fit to a smaller set, e.g., half the number of points, and check the error.
- We are done if the error condition is satisfied, and no more points can be fitted without exceeding the error bound.

That is, this method takes the data points and the parameters and uses the longest Bezier arcs to approximate a subset of these points. Once done, an index set $b_1, \cdots, b_d$ is returned so that the points $Q_{a,b} = (x_{a,b}, y_{a,b}, h_{a,b})$, $b = b_1, \cdots, b_d$ are the junction points of the piecewise Bezier approximation. Figure 3 illustrates how the algorithm works. The junction points are marked by boxes.

3.3 B-spline Fitting

For each row, we will fit a B-spline curve, given the decomposition of the points $b_1, \cdots, b_d$. The first order of business is to determine the number of control points, i.e., the minimum degrees of freedom to achieve the desired accuracy. Since there are d number of Bezier segments, the minimum degree of freedom is $d \cdot q$, where q is the degree of the curve.

Next, we compute the knot vector using the $q-1$ average of parameters that fall within the intervals $\left[v_{b_e}, v_{b_{e+1}}\right]$. The $q-1$ average splits the data set into $q-1$ subsets and computes the averages of those subsets. Once these averages have been computed, we can use the well-known De Boor formula to compute the knots [8].

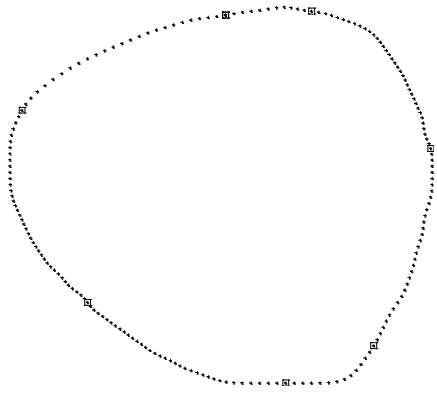


Figure 3: Bezier decomposition of a data set.

Given the number of control points and the knot vector, the only thing left is to compute the missing control points. This can be done by solving the following system of linear equations.

$$N^T N \; P = R$$

$$N = \begin{bmatrix} N_{0,q}(v_0) & N_{1,q}(v_0) & \cdots & N_{m,q}(v_0) \\ N_{0,q}(v_1) & N_{1,q}(v_1) & \cdots & N_{m,q}(v_1) \\ \vdots & \vdots & \cdots & \vdots \\ N_{0,q}(v_l) & N_{1,q}(v_l) & \cdots & N_{m,q}(v_l) \end{bmatrix} \quad R = \begin{bmatrix} N_{0,q}(v_0)Q_{0,j} + N_{0,q}(v_1)Q_{1,j} + \cdots + N_{0,q}(v_l)Q_{k,j} \\ N_{1,q}(v_0)Q_{0,j} + N_{1,q}(v_1)Q_{1,j} + \cdots + N_{1,q}(v_l)Q_{k,j} \\ \vdots \\ N_{m,q}(v_0)Q_{0,j} + N_{m,q}(v_1)Q_{1,j} + \cdots + N_{m,q}(v_l)Q_{k,j} \end{bmatrix}$$

This simple fitting technique can be extended by adding various constraints: (1) end derivatives can be specified, (2) floating end conditions, i.e., the end data points for each row do not coincide with the control points, and (3) end constraints, i.e., the endpoints are interpolated. For DICOM data fitting, the end constraints have been applied to precisely delineate the boundary points.

After each row has been fitted, the method is repeated on the control points obtained from the fitting algorithms. That is, a surface is lofted through the set of curves either by approximating the control points or by interpolating them. The approximation can also be turned into interpolation by setting the tolerance to zero or a very small number. Therefore, the method has a large variety of options to control the amount of data generated and the precision of the rendering.

3.4 Error Control

Controlling the error of curve or surface approximations is a simple task; at least theoretically: all data points are projected to the curve/surface, and the distance between the point and its projection is computed. Point projection is expensive and error-prone. It requires the computation of higher derivatives and iterative methods that sometimes do not converge or converge to the wrong solution. Hence, we did not use this method. Instead, we computed the parametric error

$$E = \left| S(u_i, v_j) - Q_{i,j} \right|$$

where $Q_{i,j}$ are assumed at the parameter pair u_i, v_j . This parametric error is always larger than the actual perpendicular error, however, by not much. In approximation, the error tolerance is usually an order of magnitude, e.g., 0.1, or 0.01, etc. So, if the parametric error is, say, 1.5 times larger than the perpendicular error, this discrepancy is quite acceptable.

If the parametric error test is passed, the approximation has been completed. Otherwise, in some areas, the approximation failed the error condition. We have two options: (1) introduce more degrees of freedom locally, or (2) reapproximate globally with a tighter tolerance. We elected to pursue the second method because we wanted the piecewise Bezier approximation to capture the shape information inherent in the data. Practical experience shows that 0.25 to 0.5 of the original tolerance always produced the desired result. In fact, we can start the decomposition with about 0.25 of the initial tolerance so that no iteration is ever needed.

The final step in the approximation process is cleanup. Because the exact number of control points is not known ahead of time, either we have to iterate a few times or use a smaller tolerance to guarantee accuracy. The end result is that we have more control points than are absolutely needed. This is not a problem in the world of B-splines. Excess control points mean extra knots, which can be eliminated by knot removal [8]. So, passing the surface through a knot removal system that removes knots while checking the error between the original surface and the knot removed surface will eliminate all excess knots, and hence control point.

Fitting B-spline surfaces to the DICOM data has a few distinct advantages. First, they are not subject to visualization or optical artifacts. They are 3-D surfaces that can be looked at in any position with various lighting conditions. Second, there is no loss of data. The DICOM data is precisely fit, and the resulting surface does pass near/through every data point. Third, if the noise in the DICOM file is to be eliminated, smoothing can be applied to any level. In fact, a high level of smoothing can "melt" the surface down to the plane (see the example below). Last but not least, and this is most important, the surfaces are scalable without any loss. Image scaling is done via pixel replication, which produces blurred pictures. That is, zooming into a medical image won't reveal any more details; in fact, it will make the image worst. Zooming into a B-spline surface gives all the details there are in the DICOM file.

4 APPLICATION EXAMPLES

This section provides several examples to illustrate the potential applications of B-spline reliefs as an adjunct methodology to simple imaging.

4.1 Brain MRI

Let us start a series of examples with an analysis of a brain MRI image. The image is shown in Figure 4 top. It has a 512 x 512 resolution with a maximum intensity of 1,631. The corresponding B-spline relief is shown in Figure 4 bottom. This is what is in the DICOM file; no smoothing, no compression—a world of difference between the two, especially around the borders. The relief is shown in a somewhat larger format for the reader to appreciate the intricate details.

The visual artifact around the border in the top image is immediately apparent. The image indicates that the border is smooth and thick. The relief shows something else: the border is sharp and very jaggy. The relief also shows that the border consists of several layers, each layer representing the bone structure of the skull. The model also illustrates the clear separation of the brain from the bone. To appreciate the intricate details the DICOM file contains, Figure 5 shows a zoomed-in part of the brain.

The B-spline relief method allows us to model what happens when the DICOM file is rendered on a display with 256 levels of grayscale or visualized by the eyes that can distinguish about 40-50 different levels of shade. Figure 6 shows the result. On the left is what the display can render, and on the right is what the eyes can see. Comparing Figures 4 and 6 makes it clear that the naked eye is a very inefficient diagnostic tool.

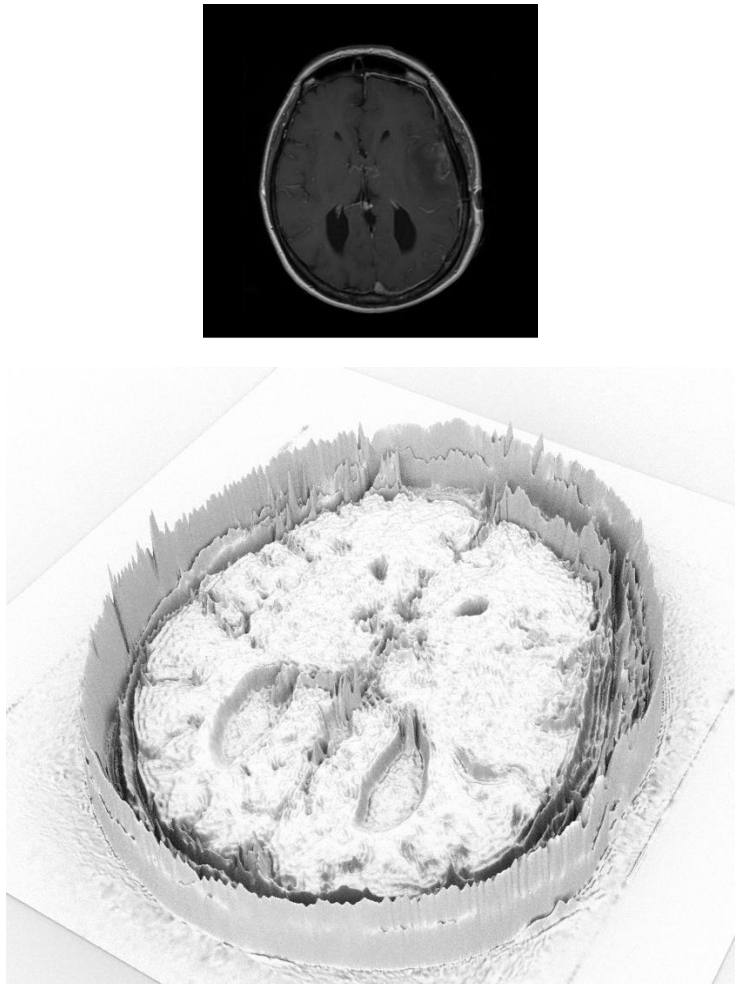


Figure 4: Brain MRI (top) and corresponding B-spline relief (bottom).

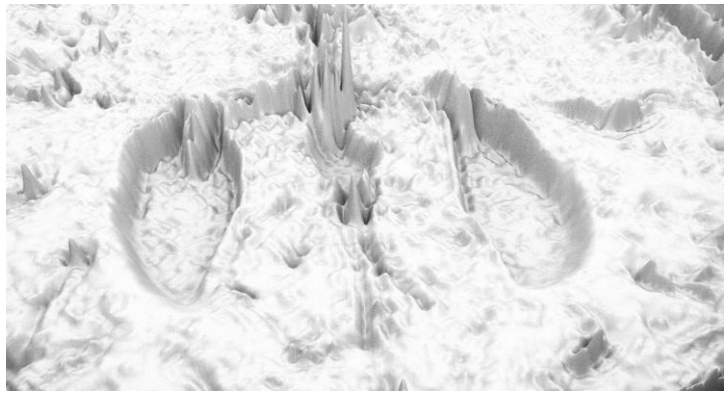


Figure 5: Intricate details of the brain slice after zooming into the B-spline relief.

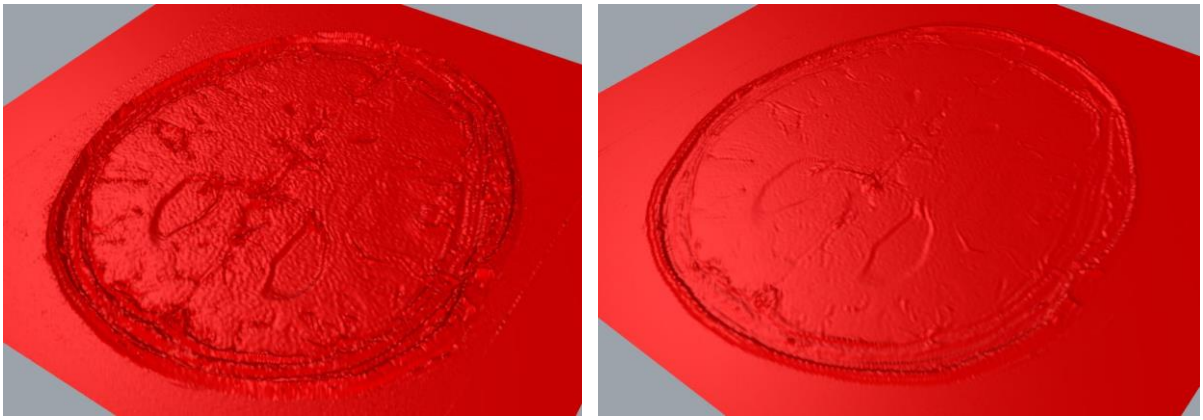


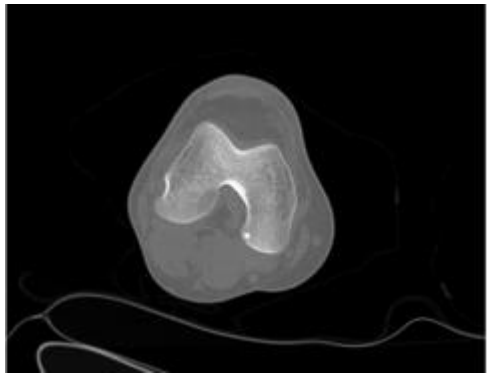
Figure 6: Display level (left) and eye level (right) details.

Figure 6 right depicts what one can see in Figure 4 top. It is still questionable what is the preferred way of looking at DICOM files: the image in Figure 4 top or the relief in Figure 6 right. To our eyes, the relief is considerably more compelling than the image, even with such a significant compression.

4.2 Knee CT

Next, let us investigate a cross-section of the knee. Figure 7 shows the knee CT image on the top and the corresponding B-spline relief on the bottom. The CT image has a 512 x 512 resolution, and the maximum intensity is 2,412. Similar to the brain MRI, the relief reveals an incredible amount of detail. One can also observe the Mach band effect where the bone meets the muscle; the bone boundary appears lighter and the muscle darker. On the other hand, the relief shows that the outer and the inner boundaries (the elevations) are sharp. The image's intensities bleed over to the adjacent parts, whereas the intensities in the relief result in higher elevations, eliminating the optical illusions.

Figure 8 illustrates the process of denoising (using the algorithm in Section 3), i.e., eliminating sudden changes in intensities. The top left figure shows one level of denoising, the top right has two levels, whereas the bottom left is denoised 15 times, and the bottom right 50 times. It is pretty evident that as the denoising continues, details gradually disappear, and the relief kind of "melts" away. This is the same phenomenon when the 1 to 2,412 intensities are compressed down to 1-256 on the display and finally down to 1-40 in our eyes. Information simply disappears!



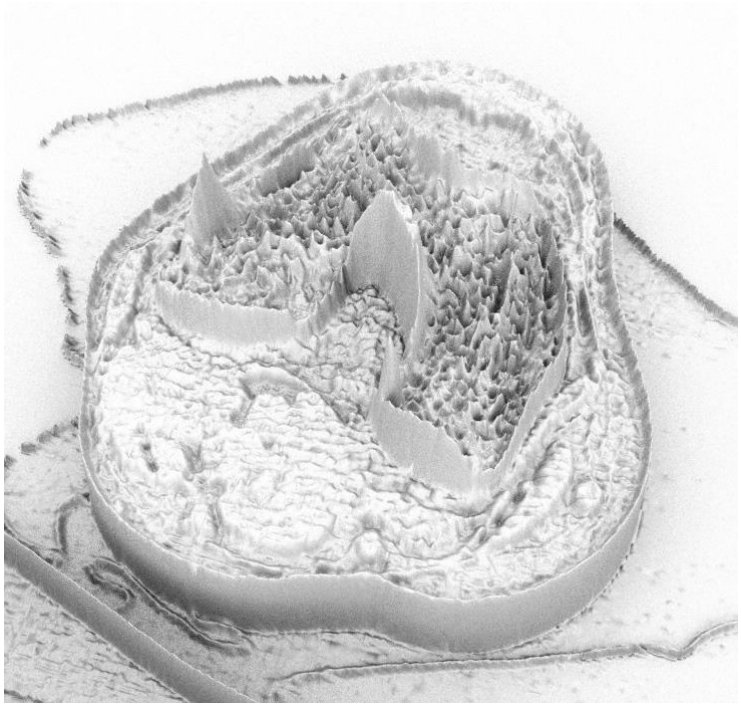
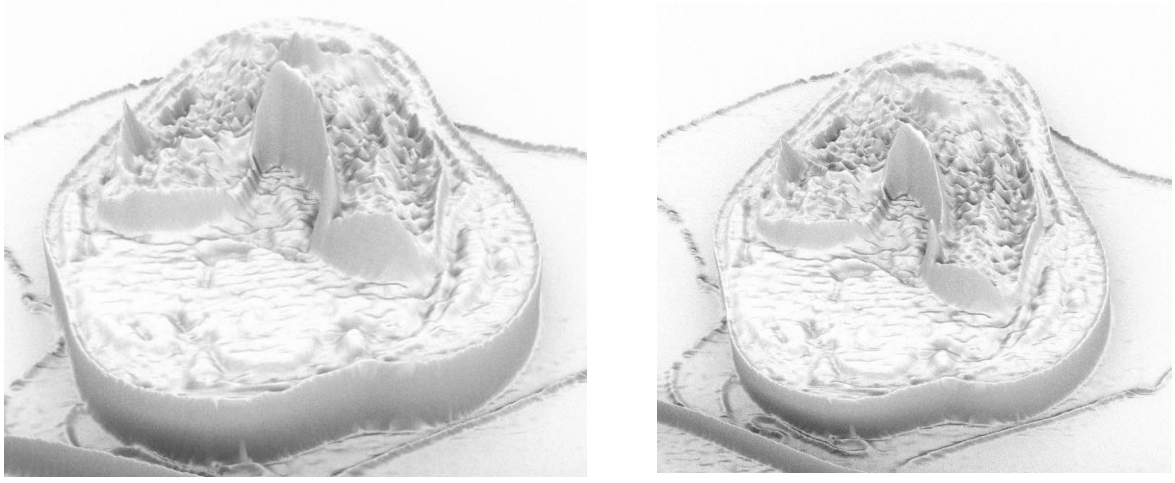


Figure 7: Knee CT (top) and B-spline relief (bottom).

From this simple example, it should be clear that denoising is not the way to eliminate significant variations in scanned intensities. Even one-to-two levels of denoising reduce details that could be useful in medical diagnosis (compare Figure 7 bottom to Figure 8 top).

If image compression causes loss of information and visual artifacts, and if denoising creates a "melting" effect, then so far, the original DICOM data is still the clear winner when analyzing tissues using linear attenuation of radiation. The following example reveals an even more profound difference between the original data and its image-based interpretation.



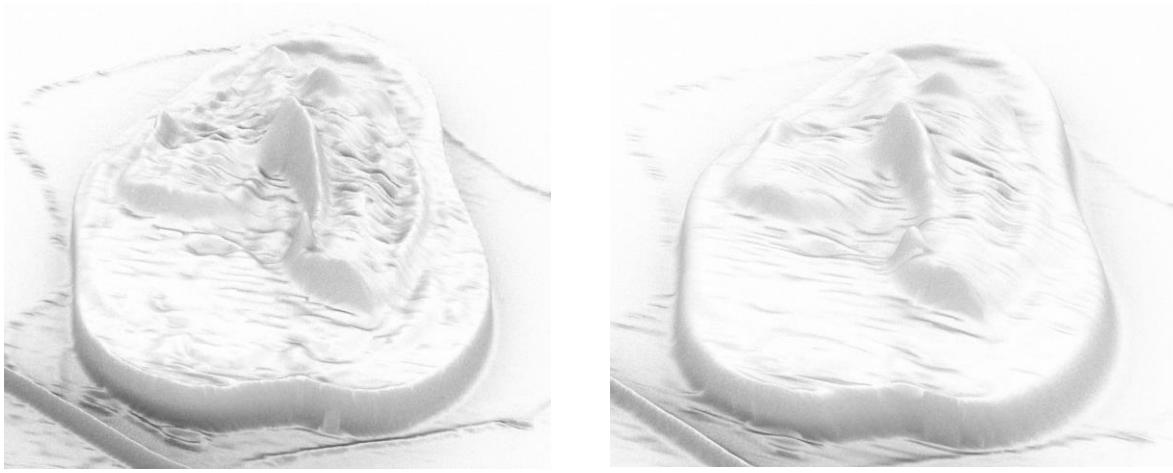


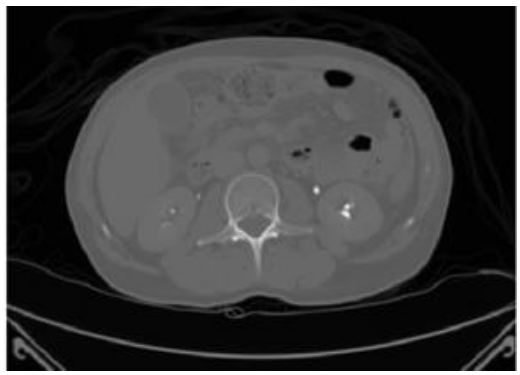
Figure 8: The effect of denoising the DICOM data: top left, level 1; top right, level 2; bottom left, level 15; bottom right, level 50.

4.3 Spine CT

In this section, let us investigate an abdominal slice that cuts through the spine. The focus is on the spine section, Figure 9. The resolution is 512 x 512, and the maximum intensity is 2,568. At first glance, nothing seems out of order; i.e., it appears that everything is where it should be. Now let us start taking a closer look by zooming into the relief. Figure 10 shows the result.

The areas marked by the arrows are not visible in the image of Figure 9 top. When zooming into the image, everything becomes very blurry, making the enlarged image completely useless. It shows small air pockets in the bone, which may or may not be a concern. Speaking of air pockets, Figure 11 shows a close-up view of many of the air pockets that are pretty apparent in the abdominal cavity. Figures 10 and 11 also provide details of the tissue variations on this slice. In contrast, in the image, these tissues are grayed out, i.e., they are just a blob of gray, giving very little helpful information.

The images in Figures 10 and 11 are top views of the relief surfaces. One of the B-spline reliefs' significant advantages is that they can be viewed in many different ways: top, side, front, and perspective views. The perspective views are dynamic, i.e., one can adjust the view in real-time to suit the viewer's preference. Various kinds of zooming are also available, e.g., dynamic or window zooming. On the other hand, the image is a static entity, and it is non-scalable without losing sharpness and hence details.



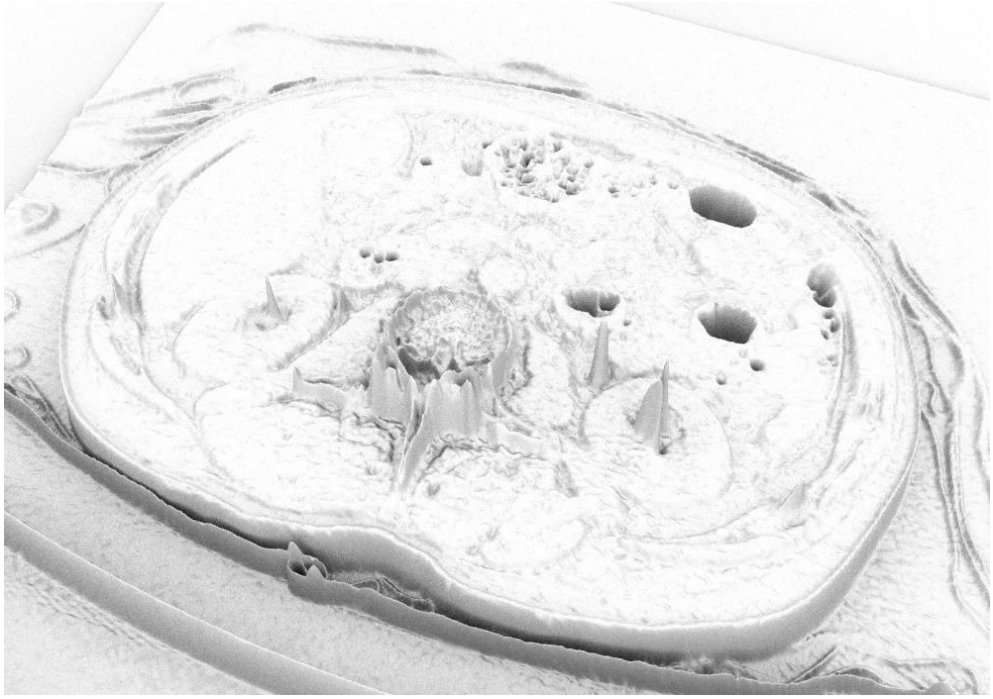


Figure 9: Abdomen/spine CT (top) and B-spline relief (bottom).

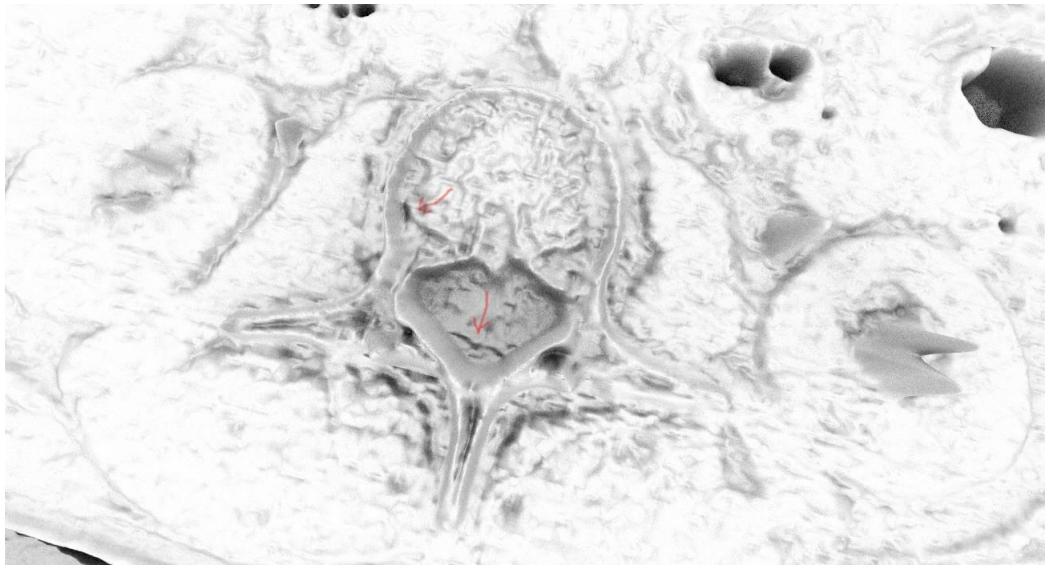


Figure 10: Close-up view of the spine bone.

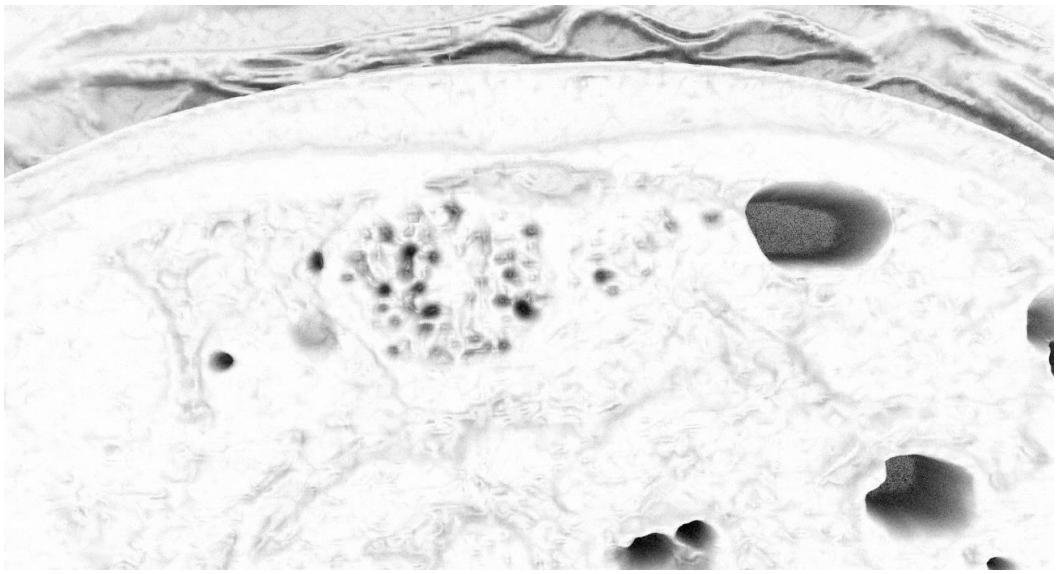


Figure 11: Close-up view of air pockets in the abdominal slice.

4.4 Abdomen CT

Let us now take a slice of the body vertically to expose the chest and the abdominal region, Figure 12. The image on the left has a resolution of 940 x 774 with a maximum intensity of 2,266.

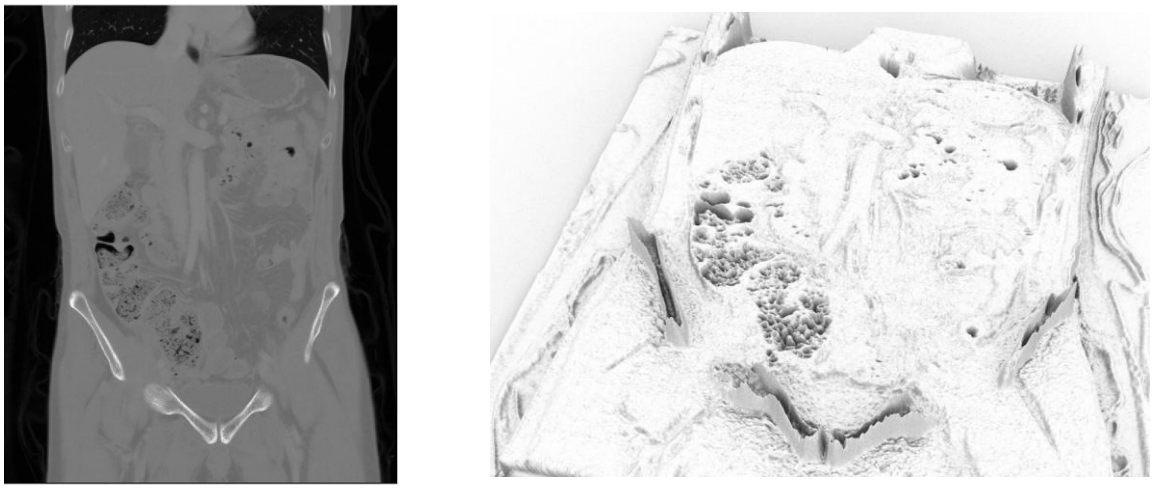


Figure 12: Abdomen CT (left) and B-spline relief (right).

There are many air pockets that need to be explored; however, magnifying the left image makes everything very fuzzy. Zooming into the relief gives us a clearer picture, Figure 13.

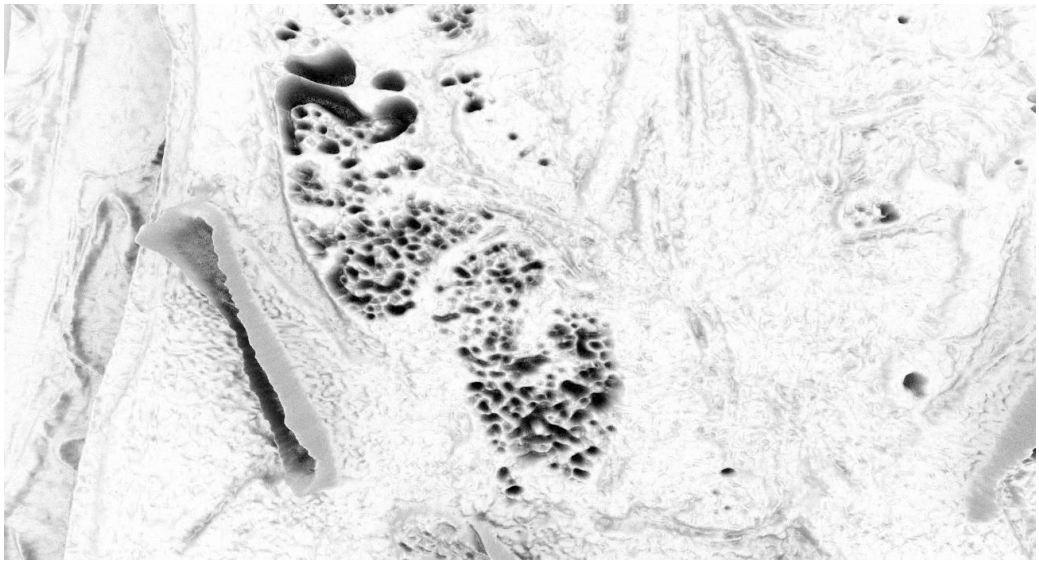


Figure 13: Abdominal air pockets close-up view.

When staples perforated the bowels, these air pockets were noted in the bowels used in a hernia repair. The staples are still not visible, though. They become more apparent when we take an X-ray of the lower abdomen/hip area, which is the next section's subject.

4.5 Hip X-Ray

The hip X-ray is shown in Figure 14. The X-ray comes with a resolution of 2130 x 1750 with a maximum intensity of 4,095. The corresponding B-spline relief does not seem to be all that useful when the full range of the DICOM file is exposed. However, when taken down to the display level, Figure 15, the relief appears to be more helpful.

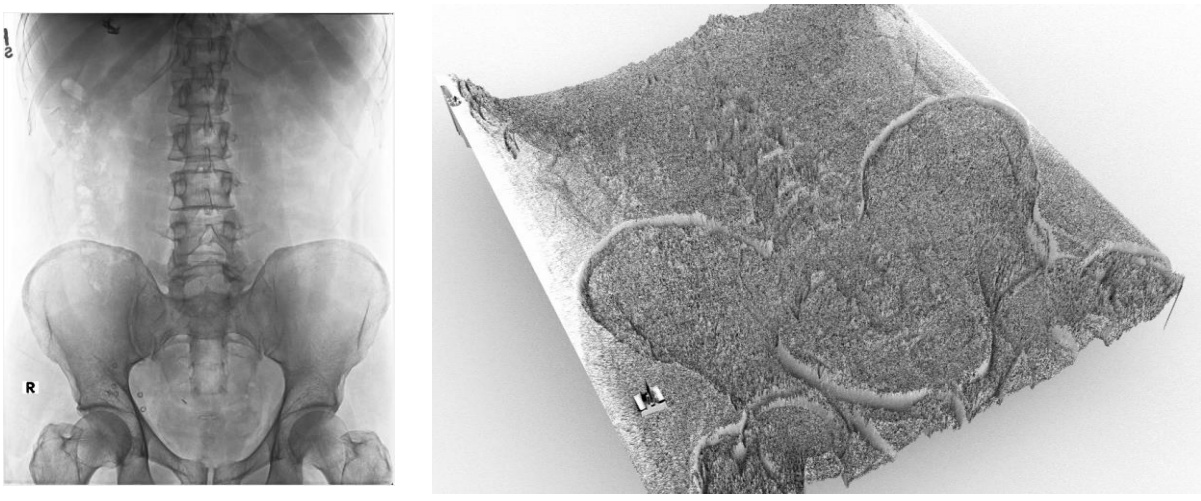


Figure 14: X-ray (left) and DICOM level B-spline relief (right).

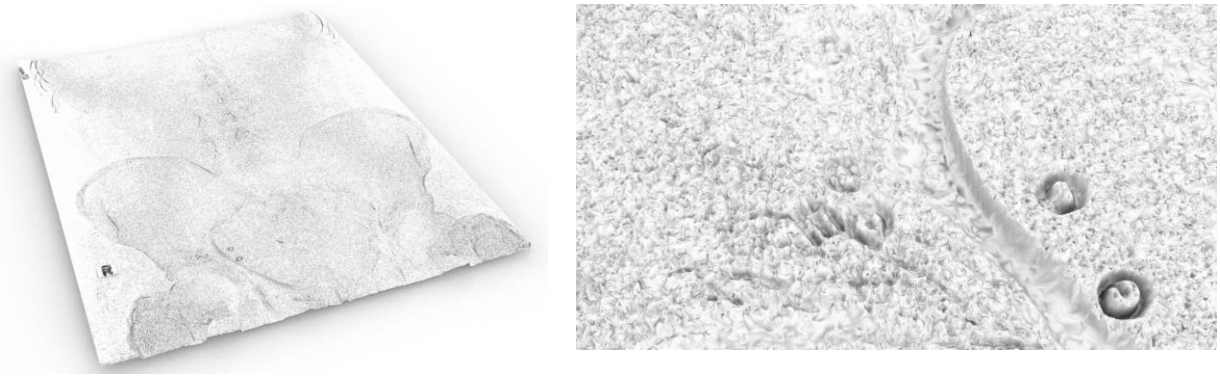


Figure 15: Display level X-ray rendering (left) and close-up view to expose staples (right).

Zooming into the area where the staples are located reveals quite a bit of detail about the shape and the locations of these sharp instruments used to secure meshes onto tissues.

While B-spline reliefs did very well with MRI and CT images, rendering the complete DICOM data of X-rays with B-spline surfaces may not be immediately valuable, Figure 14 right. However, going down to the display level may turn out to be the way to go, especially for close-up views.

5 CONCLUSIONS

In this paper, we introduced B-spline reliefs to illustrate some of the shortcomings of image-based diagnosis using MRI, CT, and X-ray images. The reliefs clearly show that images alone may be inadequate in finding enough details when conditions are not in their advanced stage. The images suffer from visual artifacts, they are non-scalable, and the eyes can only see about 1% of the available information in the DICOM file. On the other hand, B-spline reliefs can show all the data, and the maximum height can be adjusted to show as much detail as needed; the surface is scalable, and features not present in the images reveal themselves at specific heights of the reliefs.

Les A. Piegl, https://orcid.org/0000-0003-0629-8496
William L. Mondy, https://orcid.org/ 0000-0003-4464-4783

REFERENCES

[1] Chiorean, L-D.; Szasz, T.; Vaida, M-F.; Voina, A.: 3D reconstruction and volume computing in medical imaging, Acta Technica Napocensis, 52(3), 2011, 18-24.
[2] Fisher, F.; Selver, M. A.; Gezer, S.; Dicle, O.; Hillen, W.; Systematic parametrization, storage, and representation of volumetric DICOM data, Journal of Med. Biol. Eng., 35, 2015, 709-723. https://doi.org/10.1007/s40846-015-0097-5
[3] Gambino, O.; Rundo, L.; Cannella, V.; Vitabile, S.; Pirrone, R.: A framework for data-driven adaptive GUI generation based on DICOM, Journal of Biomedical Informatics, 88, 2018, 37-52. https://doi.org/10.1016/j.jbi.2018.10.009
[4] Gutman, D. A.; Dunn, W. D.; Cobb, J.; Stoner, R. M.; Kalpathy-Cramer, J.; Erickson, B.: Web based tools for visualizing image data and development of XNATView, a zero footprint image viewer, Frontiers in Neuroinformatics, 8(53), 2014, 1-10. https://doi.org/10.3389/fninf.2014.00053
[5] Haak, D.; Page, C-E.; Desemo, T. M.: A survey of DICOM viewer software to integrate clinical research and medical imaging, Journal of Digital Imaging, 29(2), 2016, 206-215. https://doi.org/10.1007/s10278-015-9833-1

[6] Klotzer, J. D.; Padate, P. B.; Jester, E. T.; Arlagada, V. K. R.: Zero footprint DICOM images viewer, US Patent Pub. No: US 2014/0143298 A1, 2014.
[7] Liu, S.; Liao, W.; Yu, Q.; Cheng, X.; Dai, N.; Zhang, X.: The development of a system for 3D reconstruction from DICOM data and collaborative visualization, Journal of Biomedical Engineering, 24(5), 2007, 1152-1156.
[8] Piegl, L. A.; Tiller, W.: The NURBS Book, Springer-Verlag, New York, NY, 1997. https://doi.org/10.1007/978-3-642-59223-2
[9] Romans, L. E.: Computed Tomography for Technologists: A Comprehensive Text, Wolters Kluwer, Philadelphia, PA, 2019.
[10] Rosset, A.; Spadola, L.; Ratib, O.: OsiriX: an open-source software for navigating in multidimensional DICOM images, Journal of Digital Imaging, 17(3), 2004, 205-216. https://doi.org/10.1007/s10278-004-1014-6
[11] Stutsman, S.; Napoli, J.; Chun, W-S.: 3D volume construction from DICOM data, US Patent Pub. No.: US 2006,0056680 A1, 2006.
[12] Taka, S. J.; Srinivasan, S.: NIRViz: 3D visualization software for multimodality optical imaging using visualization toolkit (VTK) and insight segmentation toolkit (ITK), Journal of Digital Imaging, 24(6), 2011, 1103-1111. https://doi.org/10.1007/s10278-011-9362-5