Published in final edited form as:

Proc SPIE Int Soc Opt Eng. 2016 ; 9778: . doi:10.1117/12.2219777.

Virtual rough samples to test 3D nanometer-scale scanning electron microscopy stereo photogrammetry

J.S. Villarrubia, V.N. Tondare, and A.E. Vladár

Engineering Physics Division, Physical Measurements Laboratory, National Institute of Standards and Technology,† Gaithersburg, MD, USA 20899

Abstract

The combination of scanning electron microscopy for high spatial resolution, images from multiple angles to provide 3D information, and commercially available stereo photogrammetry software for 3D reconstruction offers promise for nanometer-scale dimensional metrology in 3D. A method is described to test 3D photogrammetry software by the use of virtual samples—mathematical samples from which simulated images are made for use as inputs to the software under test. The virtual sample is constructed by wrapping a rough skin with any desired power spectral density around a smooth near-trapezoidal line with rounded top corners. Reconstruction is performed with images simulated from different angular viewpoints. The software's reconstructed 3D model is then compared to the known geometry of the virtual sample. Three commercial photogrammetry software packages were tested. Two of them produced results for line height and width that were within close to 1 nm of the correct values. All of the packages exhibited some difficulty in reconstructing details of the surface roughness.

Keywords

critical dimension (CD); dimensional metrology; model-based metrology; scanning electron microscopy (SEM); simulation; stereo photogrammetry; surface roughness; virtual sample

1. INTRODUCTION

With the introduction of non planar memory and logic devices beginning at the 22 nm node, semiconductor electronic devices began to have significant functional dependence on vertical dimensions of their structures. For FinFET or Tri-Gate transistor architectures, for example, the size of the conducting gate channel depends on the height of the fin. Structure height, wall angles, and sidewall roughness join width as critical process variables.¹

In a scanning electron microscope (SEM) image, the lateral dimensions are spatial. The vertical dimension is an intensity related to the backscattered or secondary electron yield. The yield variation carries spatial information that can be retrieved with the help of a model that relates yield to shape.^{2,3} Alternatively, tilting the sample permits a new image in which the vertical axis of the former image has a component in the new lateral direction. The change in lateral separation of features (known as disparity) is a function of tilt angles and

the height difference of the features. Stereo photogrammetry (by which we mean estimation of 3D coordinates from sets of two or more images) may be used to reconstruct the 3D sample. Piazzesi described the mathematics of such reconstruction for SEM in 1973.⁴ Others have looked extensively into various measuring and instrument errors, their avoidance, and their effect on the quality of reconstruction.^{5–8}

There are beginning to be a large number of options for software to determine 3D sample shape from multiple views. At the time of this writing, 58 are listed in the Wikipedia article on “Comparison of photogrammetry software.” When the purpose of 3D reconstruction is not merely an artistically pleasing rendering, but rather quantitative accurate measurement, how is one to judge the adequacy of software? In this paper we adopt the approach of using a virtual sample to assess software performance. By a virtual sample, we mean a mathematical description of an object that does not exist in reality. We make images of the object from different angles using the JMONSEL³ simulator, much as one would make measurements of a real object in an actual SEM. These images then become the inputs to the stereo photogrammetry software. A disadvantage of this approach is that it does not test contributions of the instrument to measurement errors. For this reason, it is not a complete test; such errors must be assessed in other ways. The advantage of the approach is that the true sample shape is known with mathematical accuracy, a level that measurement errors make impossible to replicate in real samples at the nanometer scale relevant for features of interest in semiconductor electronics applications. This makes it possible for software designers to determine whether an algorithm strategy—more noise filtering or less, the method of discovery of homologous points in images of the input set, approximation A vs. approximation B—improves not only the appearance but the accuracy of the result.

The virtual sample was made rough, which permitted photogrammetry software to locate a high density of homologous points in image sets. We purchased three commercial software packages that advertise their use with SEM. We designate these A, B, and C, intentionally leaving them otherwise unspecified. We used these to reconstruct the sample shape. The center profile of the reconstructed shape was compared to the corresponding true profile from the virtual sample in order to determine reconstruction errors. The purpose of this sampling of available software is to learn something about the current state of the art for such reconstruction, as judged by measurement errors in the height, width, and roughness of our virtual specimen. Different software packages exhibit different levels of performance and perform better for some measurements than others. Our results demonstrate by example that useful things can be learned by this technique. We anticipate that such data sets will also be useful to software developers who wish to improve the reconstruction accuracy of their algorithms.

In Sec. 2 we describe our method for constructing a roughness-wrapped virtual sample with a desired power spectral density, representation of the sample in a form suitable for SEM simulation, and the simulation in JMONSEL from a series of different angular viewpoints. In Sec. 3 we describe the 3D reconstruction results with our software packages and compare them to the true sample shape. In Sec. 4 we discuss the significance of the results of this comparison.

2. ROUGHNESS-WRAPPED VIRTUAL SAMPLE

Generation of images to be used as input to the 3D reconstruction software was performed in several steps: (1) A rough surface was generated with the desired power spectral density (PSD). Roughness in this surface was all in the z direction (normal to the substrate). (2) This surface was wrapped around a smooth line. (3) The wrapping sometimes results in surface self-intersections (“collisions”) at inside corners. These collisions were resolved. (4) The collision-resolved, wrapped surface was represented as an intersection of three height maps. This form is recognized by JMONSEL, our image simulation software. (5) Images of the height-map-represented virtual sample were simulated at a sequence of tilt angles.

2.1 Generation of rough surface with desired power spectral density

Our algorithm to generate a rough surface proceeds in these steps:

1. We create a 2D array with the desired dimensions and populate it with unit amplitude “white noise” (white, that is, up to the array’s Nyquist frequency): normally distributed random numbers with mean 0 and variance 1.
2. We take the fast Fourier transform (FFT) of that array. The result is a complex-valued array, $\mathbf{Z}$ such that $\mathbf{Z}_{jk} = \mathbf{A}_{jk} \exp(i\Phi_{jk})$, where $\mathbf{A}$ and Φ are real-valued amplitude and phase arrays. By definition, the two-sided power spectral density of $\mathbf{Z}$ is the array with elements $(\mathbf{A}_{jk})^2$. Even though “white” implies equal amplitudes at all frequencies, the amplitudes (the elements $\mathbf{A}_{jk}$) are only equal *on average*. The equality is statistical, not rigorous in each realization of random noise.
3. Suppose our desired power spectral density is stored in array $\mathbf{P}$. Here we have a choice. If we want our generated surface to have exactly $\text{PSD} = \mathbf{P}$, we replace $\mathbf{A}$ with $\sqrt{\mathbf{P}}$ to form $\mathbf{Z}'_{jk} = \sqrt{\mathbf{P}_{jk}} \exp(i\Phi_{jk})$. If our application is better served by retaining the statistical variation of PSD realizations, we form instead the matrix $\mathbf{Z}'$ with elements $\mathbf{Z}'_{jk} = \sqrt{\mathbf{P}_{jk}} \mathbf{Z}_{jk}$. These options either replace or scale $\mathbf{Z}$ ’s original on-average flat, white-noise, PSD. Because $\mathbf{P}$ is a PSD and hence real valued, either option leaves unaltered the random phases generated in step 2.
4. We form the inverse FFT of $\mathbf{Z}'$. This is our desired rough surface.

Methods similar to this were recently reviewed by Mack⁹. If the statistical PSD option is chosen at step 3, the method is similar (possibly equivalent) to that of Thoros¹⁰.

The desired PSD required for step 3 is specified as a 2D array of squared amplitudes. As such it is quite general, and need not be expressible in a simple analytical form. It could, for example, be chosen equal to the measured PSD of an actual sample. However, for the present application we chose to construct the array from Palasantzas’s isotropic analytical form¹¹ for the PSD:

$$P(k) = \frac{4\pi\alpha\sigma^2\xi^2}{(1+k^2\xi^2)^{1+\alpha}} \quad (1)$$

The leading constant factor in the PSD depends on one's choice of Fourier transform convention and whether the PSD is 1-sided or 2-sided. The version in Eq. 1 follows Zhao et al.¹² This PSD expression has parameters σ^2 for roughness variance, ξ for lateral correlation length, and α for roughness exponent. To keep them realistic we adopted values $\xi = 15$ nm and $\alpha = 1$ close to those obtained by fitting this expression to the PSD estimated from an image of a rough semiconductor industry sample available to us. We used $\sigma = 1$ nm. Equation 1 was used to generate values to populate the PSD array, P , for step 3 of the above procedure. Then step 4 produced our rough surface, a representative section of which is shown in Fig. 1.

2.2 Wrapping the sample

Our underlying sample shape is the smooth line shown in Fig. 2. It is an 80 nm tall line, 50 nm wide at half-height, wider on the bottom than the top, with sidewalls 3° from the vertical and 10 nm top corner radii. This represents the smooth average shape around which the rough "skin" described in the previous section must be wrapped. The skin consists of vertical (z) displacements (initially with $\sigma = 2$ nm) at equal intervals in x and in y . To wrap our shape we must discretize our smooth surface into a series of (x, y, z) coordinates at equal distances (arc length) along the surface, i.e., in the direction parallel to the local surface, not necessarily the x direction. Each of these points must then be displaced by an amount dictated by the corresponding point in the skin, but in a direction along the local surface normal, as indicated by the arrows in Fig. 2.

2.3 Collision resolution

Sometimes (e.g., near the bottom corners in the figure) these displacements cause one part of the surface to cross another part. If $p_i = (x_i, z_i)$ and p_{i+1} are adjacent points (same y value) along a roughened profile, we check whether the line segment that joins them intersects each of the other line segments, from p_j and p_{j+1} for all $j > i$. If any such intersections are found, the collision of the two affected segments can be resolved by reducing the displacements of their 4 endpoints to some fraction f ($0 < f < 1$) of their original values. We use the largest value of f that resolves the collision. This may simply propagate the collision to a neighboring line segment. The procedure of checking and resolving collisions is therefore iterated until all collisions are resolved. At this point all displacements were further reduced by an additional factor of 2, yielding a surface roughness just under 1/2 the original value, for $\sigma = 1$ nm. A point cloud rendering of the $\sigma = 1$ nm resulting virtual sample is shown in Fig. 3.

2.4 Representation as an intersection of height maps

Electron trajectory simulations are necessary in order to produce simulated images of the virtual sample, but the surface point cloud representation described in the previous section is insufficient for that purpose. The simulator needs to assign scattering properties to volumes,

and those volumes must be bounded by surfaces. Intersection of an electron's path with a bounding surface signals the transition to a new volume with possibly different scattering properties. The points in the point cloud lie on the boundary, but it remains to specify the surfaces that connect the points and the volumes that these surfaces bound. For this reason the point cloud representation was converted to a representation in terms of height maps.

A basic height map in the JMONSEL SEM simulator³ that we used is a 2D array of z values, equally spaced in x and y . Consider one "cell" of this array, defined as the 4 points at positions (i, j) , $(i+1, j)$, $(i, j+1)$, and $(i+1, j+1)$. These 4 points define two triangular surfaces, one with vertices (x_i, y_j, z_{ij}) , $(x_{i+1}, y_j, z_{i+1,j})$, and $(x_i, y_{j+1}, z_{i,j+1})$, the other with vertices $(x_{i+1}, y_{j+1}, z_{i+1,j+1})$, $(x_{i+1}, y_j, z_{i+1,j})$, and $(x_i, y_{j+1}, z_{i,j+1})$. Beyond the borders of the explicitly specified portion of the height map, i.e., to the left, right, above, or below in the x - y plane, the heights are considered to remain constant at the last specified value. The union of the surfaces from all the cells forms a surface that partitions space into two volumes: a lower one defined as the inside and an upper one defined to be outside. A cut at constant y through our virtual sample (with roughness exaggerated for clearer illustration) is indicated by the thin line in Fig. 4a. A cut at the same y through a corresponding height map is indicated by the thick line. The height map is constructed by dividing the desired interval into regularly spaced x values, $\{x_i\}$. For each x_i we find all intervals in the point cloud (thin line) that contain this x_i , interpolate the corresponding z for each such interval, and associate the maximum of these z values with that x_i . At most places, the virtual sample surface shown in Fig. 4a is single-valued and the height map very closely approximates it. (Errors are only those from interpolation, and may be made as small as we wish by making the interval between x_i small enough.) At the sidewalls, however, the roughness makes our virtual sample reentrant. At such locations, the higher parts of the sample shadow the lower parts, as is evident in the figure. The height map contains or bounds the sample, but in some places more tightly than others.

The sample representation is improved by taking advantage of transformations and set operations that can be performed on height maps. The virtual sample was rotated 87° to the right as shown in Fig. 4b. A second height map, indicated again by the thicker line, was generated in this orientation, and then rotated 87° to the left, returning it to the original orientation. A third height map (not shown) was also formed, this time with the line rotated 87° to the left to expose the other sidewall, followed once again by the opposite rotation. Since each of these represents an outer bound on the sample, their intersection also represents an outer bound, in this case a noticeably tighter one (Fig. 4c). This intersection of three height maps was used to represent the sample in JMONSEL.

2.5 Simulation of SEM images

We used JMONSEL to import the height map representation of the virtual sample and produce images of the same location at sample tilts from -85° to 85° at 5° intervals. For each image, 10 000 electrons at 500 eV were raster scanned across 241×101 pixels, each of which had size $0.5 \text{ nm} \times 0.5 \text{ nm}$. Electron landing positions were normally distributed around the target position with standard deviation 0.5 nm. Electrons with energy between 0

eV and 50 eV that escaped to a hemispherical collector above the sample were counted and converted to a proportional intensity level to produce images.

Inside the sample, assumed to be uncharged silicon, electrons were propagated in trajectory steps each of which was terminated by a scattering, boundary crossing, or trajectory termination event. The scattering events included elastic electron-nuclear scattering, inelastic secondary electron generation, and phonon scattering. Elastic scattering was modeled using the Mott cross-sections in NIST SRD 64.¹³ Secondary electron generation was modeling using scattering tables computed using dielectric function theory without the single-pole approximation.¹⁴ The phonon scattering model is based on that of Llacer and Garwin.¹⁵ At the Si/vacuum interface, electrons reflected or refracted at the boundary according to a quantum mechanical barrier transmission model in which the potential energy was described by $U(x) = U/[1 + \exp(-2x/w)]$ with x the distance from the boundary, $w = 1$ nm, and $U = 3.75$ eV. Secondary electrons were simulated in the same way as primary electrons. Simulation of an electron stopped when it either escaped the sample and was detected or when its energy dropped so low that escape was impossible. More details of JMONSEL's models were previously published.³ Some of the resulting images are shown in Fig. 5.

3. 3D STEREO RECONSTRUCTION RESULTS

We purchased three commercial software packages, which we designate A, B, and C, with which to perform stereo reconstruction from our images. All packages provided an option to generate a text file containing (x, z) pairs on the surface of the reconstructed sample along a desired slice at constant y . We used this to generate the center profile from each reconstructed data set. A typical such result is shown in Fig. 6. This one was reconstructed from input images at -15° , -10° , and -5° . If the software accepted only two images, the middle image was omitted. In this case, reconstruction has a chance for accuracy for the visible parts of the sample on the top, left wall, and part of the substrate, but no chance to accurately reconstruct the right sidewall, which is hidden at these tilt angles. Thus, all the reconstructed profiles exhibit a characteristic and expected departure from the true profile on the right side. On the top and left side, packages A and B generally followed the trend though not the detailed roughness of the true profile. Package C generally follows the top and the left part of the substrate, but does a poorer job on the sidewall than A or B.

The immediate goal was a survey of quantitative errors in the height, width, and roughness of the line. A single image set centered on the usual top-view position, e.g., at 0° and $\pm 5^\circ$, would be a poor choice for our purpose because each sidewall would be visible in only one of the 3 images. In order to have enough information to determine a width, it is necessary to reconstruct both the left and right sides of the line. This necessitated two sets of stereo images, one set all at negative tilts to make the left side visible in all its images, and another set all at positive tilts to make the right side visible. The negative set consisted of the three images at -25° , -20° , and -15° in Fig. 5. The positive set were the three at 15° , 20° , and 25° . None of the packages would accept more than one set at a time, so we performed separate reconstructions and stitched the results ourselves.

In order to obtain a height, both the top of the line and some part of the substrate must be reconstructed. Unfortunately, package C employed a windowing function that reconstructed a subset of the image smaller than the part visible in all images of the set. Because of this, this condition was not met. For this reason the remainder of the comparisons are restricted to the packages A and B. 3D visualizations of the left sidewall from these two are shown in Fig. 7.

To reconstruct full profiles, we stitched results obtained from opposite tilts by using the offsets that produced the best match for the top of the line, which was visible in both sets. For both A and B, there was some sensitivity of the determined lateral shift to the exact placement of the boundaries of “the top,” e.g., whether the region stopped short of the rounded corners or extended into them. For a reasonable range of choices, the sensitivity (1 standard deviation) of the lateral shift was about 0.3 nm for A and 1 nm for B. The sensitivity of the vertical shift was much smaller, less than 0.1 nm for both A and B. With these offsets determined, the stitched profile was computed as a weighted average of the left and right profiles, with the weights a function of position. The left profile was weighted 100% for positions to the left of the top’s left boundary, 0% for points to the right of the right boundary, transitioning linearly in between. The right profile weights were the reverse. The results are shown in Fig. 8, where they are compared to the true surface profile at that location.

For the purpose of quantifying differences between the reconstructed profiles and the true profile, we defined some relevant parts of each profile. Points in the profile were assigned to the left baseline by first selecting those that were left of the line center and with $z < z_0 + 0.05r$, where z_0 is the mean height of the leftmost 3 points and $r = z_{\max} - z_{\min}$ is the vertical range (the difference between the maximum and minimum z values in the profile). Then, of those points, the rightmost 3 nm were removed. Points were assigned to the left edge if their x coordinate was left of the line’s center and their z coordinate satisfied $z_{\min} + 0.2r < z < z_{\min} + 0.8r$. The right baseline and right edge were defined by the mirror image of these procedures. Points were assigned to the top first by selecting those with $z_{\max} - 0.1h < z < z_{\max}$ and then removing those within the first and last 5 nm. The clipping by several nanometers of the parts of the top and baseline nearest the sidewalls served to remove points associated with the transition between these regions.

With these collections of points defined, the width was defined as $w = x_{\text{right}} - x_{\text{left}}$ with x_{right} and x_{left} the average of the x coordinates of points in the right and left sidewalls. Similarly, the height of the left edge was $z_{\text{top}} - z_{\text{leftBaseline}}$ the difference between the mean z coordinate of those points in the top and those in the left baseline. A height on the right was likewise defined. The standard deviation of z coordinates in the top region was designated σ_{top} . Since the sidewalls are nearly vertical, the sidewall roughness, $\sigma_{\text{sidewalls}}$, was computed as the standard deviation of the x coordinates after subtraction of a linear best fit trend line. The results of these operations are shown in Table 1. The rough skin that we wrapped around our line causes the true values of width and height for individual profiles to differ randomly from the whole-line mean values of 50 nm and 80 nm respectively. The tabulated true values are for our chosen profile, so they reflect this difference. The uncertainties attributed to the

A and B widths reflect the stitching sensitivity to choice of boundaries for the area we matched.

4. DISCUSSION AND CONCLUSIONS

For height measurements, Package A had errors of -1.6 nm on the left and -0.7 nm on the right. Package B had errors of $+2.3$ nm on the left and -3.6 nm on the right. The average of Package B's heights was 0.5 nm closer, but the average hides greater variability, which can be seen by visually comparing the left- and right-side heights in Fig. 8a and Fig. 8b. For width measurements, the errors were -0.9 nm and -0.6 nm for A and B respectively. Once again however, Package B had a higher variability, in this case because of the sensitivity of its result to the somewhat subjective designation of the "top" of the line. (Significant sensitivities are indicated by ± 1 standard deviation values in Table 1.) A reason for the differences in these sensitivities is visible in Fig. 8. Package A's profile has variation from the mean height that appear to more closely approximate those of the true profile (albeit somewhat too smoothly) than the variations in Package B's profile. These variations play a strong role in determining the stitching of left and right profiles, which in turn is directly related to the determined width. Good consistency in these rough features between the left and right profiles leads to a strong correlation at a particular lateral offset. These observations about the rough top of the line carry over to the sides. In Fig. 8a, Package A's profile appears to be a smoothed version of the actual profile. With that observation, it is not then surprising that the quantitative roughness determination in Table 1 shows this package's roughness values to be systematically low. Package B's are also somewhat low, though closer to those of the true profile. However, in Package B's profile in Fig. 8b, some of the roughness seems poorly correlated with the true profile. With its default filtering settings, this software's reconstruction had significant outliers. We specified increased filtering in order to perform the present reconstruction. It may be that some residue of these outliers remains in the form of roughness that is not correlated to actual roughness of the true profile. Package C had an unexpected limitation in the subset of the image set that it reconstructs. This limitation prevented us from including it in the detailed comparison that we have so far summarized. Had our line features occupied a smaller fraction of our simulated images we might have been able to do so. However, the partial (left side only) reconstruction in Fig. 6 suggests its errors would be larger than those of the other two packages.

SEM photogrammetry is subject to errors from many sources, not all of which are tested by the procedure described above. For example, in a real measurement we generally do not know exactly the sample tilt angles for the images that are used. The sample may vibrate, drift, or charge during imaging. The instrument's scale calibration may have errors, and its scan may not be perfectly linear nor the x and y axes exactly orthogonal. The sensitivity of 3D stereo reconstruction to these and other error sources were the subject of earlier work.⁵⁻⁸

However, there are other possible errors that are usefully and uniquely addressed through the use of virtual samples. Since the three software packages we tried above produced different reconstructions from the same input data set, it is evident that they are not using identical algorithms. Once homologous points within the image set are identified and located, the mathematics of reconstruction is well-established. Variability could be caused by faulty

implementation in some of them. However, differences are not necessarily caused by errors. All three of the tested software packages use a preliminary pattern recognition step to identify homologous points within the image set. Typically such algorithms look for offsets in x and y that maximize the correlation between regions in the image set. A good correlation identifies corresponding regions in the images. However, correlation will not be perfect, partly because of noise in the images but also partly because the interaction of electrons with the sample is such that differences in appearance of a feature at different angles are not entirely explained by geometrical projection formulas. Algorithms may employ different strategies, approximations, amounts of filtering for noise reduction, etc., to strike different speed/accuracy trade-offs. Test problems with known correct answers are useful for algorithm testing, development, and validation. Since a virtual sample is a model in a computer, its true dimensions are known with mathematical accuracy, a level not achievable in real samples since these can only be known by virtue of a measurement that will have its own uncertainty, an uncertainty generally significant at the nanometer scale of interest to us.

Considered as a small survey of the available stereo reconstruction software, the fact that there were significant variations in the performance of different software packages suggests that the use of virtual samples has good potential to evaluate and then support and validate improvements in software quality. If there is a market for 3D reconstruction, and if differences in software quality are rendered easily visible by tests such as those described here, it seems likely that competition will drive more software closer to optimization. The fact that the best software had errors in width and height at only the ~ 1 nm level suggests that stereo reconstruction has good potential to support the electronics industry's need for better height information. To realize that potential it will be important to minimize errors from sources for which we did not test, such as errors in the tilt angles. SEMs would also need the ability to accurately mechanically tilt the sample or perform an equivalent electronic tilt of the beam's angle and scan direction.

References

1. Bunday B, Germer TA, Vartanian V, Cordes A, Cepler A, Settens C. Gaps Analysis for CD Metrology Beyond the 22 nm node. *Proc SPIE*. 2013; 8681:86813B.
2. Villarrubia JS, Vladár AE, Postek MT. Scanning electron microscope dimensional metrology using a model-based library. *Surf Interface Anal*. 2005; 37:951–958.
3. Villarrubia JS, Vladár AE, Ming B, Kline RJ, Sunday DF, Chawla JS, List S. Scanning electron microscope measurement of width and shape of 10 nm patterned lines using a JMONSEL-modeled library. *Ultramicroscopy*. 2015; 154:15–28. <http://dx.doi.org/10.1016/j.ultramic.2015.01.004>. [PubMed: 25747180]
4. Piazzesi G. Photogrammetry with the scanning electron microscope. *J Phys E: Sci Instrum*. 1973; 6:392–397.
5. Bariani P, De Chiffre L, Hansen HN, Horsewell A. Investigation on the traceability of three dimensional scanning electron microscope measurements based on the stereo-pair technique. *Prec Eng*. 2005; 29:219–228.
6. Marinello F, Bariani P, Savio E, Horsewell A, De Chiffre L. Critical factors in SEM 3D stereo microscopy. *Meas Sci Technol*. 2008; 19:065705.
7. Carli L, Genta G, Cantatore A, Barbato G, De Chiffre L, Levi R. Uncertainty evaluation for three-dimensional scanning electron microscope reconstructions based on the stereo-pair technique. *Meas Sci Technol*. 2011; 22:035103.

8. Zhu T, Sutton MA, Li N, Orteu J-J, Cornille N, Li X, Reynolds AP. Quantitative Stereovision in a Scanning Electron Microscope. *Exp Mech.* 2011; 51:97–109.
9. Mack CA. Generating random rough edges, surfaces, and volumes. *Appl Opt.* 2013; 52:1472–1480. [PubMed: 23458801]
10. Thorsos EI. The validity of the Kirchhoff approximation for rough surface scattering using a Gaussian roughness spectrum. *J Acoust Soc Am.* 1988; 83:78–92.
11. Palasantzas G. Roughness spectrum and surface width of self-affine fractal surfaces via the K-correlation model. *Phys Rev B.* 1993; 48:14472–14478.
12. Zhao, Y., Wang, GC., Lu, TM. Characterization of Amorphous and Crystalline Rough Surface: Principles and Applications. Academic Press; San Diego, CA: 2001. p. 38Vol 37, Experimental Methods in the Physical Sciences
13. Jablonski, A., Salvat, F., Powell, CJ. NIST Electron Elastic-Scattering Cross-Section Database-Version 3.1. National Institute of Standards and Technology; Gaithersburg, MD: 2002. <http://www.nist.gov/srd/nist64.cfm>
14. Mao SF, Li YG, Zeng RG, Ding ZJ. Electron inelastic scattering and secondary electron emission calculated without the single pole approximation. *J Appl Phys.* 2008; 104:114907.
15. Llacer J, Garwin EL. Electron-Phonon Interaction in Alkali Halides. I. The Transport of Secondary Electrons with Energies between 0.25 and 7.5 eV. *J Appl Phys.* 1969; 40:2766.

FIG. 1.

A subset of the generated rough sample skin with root mean square 1 nm roughness and 15 nm correlation length.

FIG. 2.
A profile of the smooth, near-trapezoidal line around which the rough skin was wrapped.
The arrows indicate local normals to the surface.

FIG. 3.
Point cloud rendering of the virtual sample. Details in the right edge are obscured because there is no hidden surface removal.

FIG. 4.

Schematic of the representation of a sample by a combination of height maps. In all panels the sample is indicated by the thin line and its height map approximation by the thick line. (a) A height map in the sample's ordinary orientation forms an outer bound. (b) Another height map with the sample rotated 87° to the right, giving a better approximation of the left sidewall. (c) The intersection of 3 height maps (the 3rd from a rotation 87° to the left, not shown) after each was returned to the ordinary orientation.

FIG. 5.
Simulated 120 nm $\times$ 50 nm images of the virtual sample at several tilt angles.

FIG. 6.

Comparison of the actual sample surface profile and the reconstructed profiles from software packages A, B, and C.

FIG. 7.
3D renderings of the left side provided by packages (a) A and (b) B.

FIG. 8.

Stitched reconstructions (thick lines) for software packages (a) A and (b) B compared to the true profile (thin line).

TABLE 1

Comparison of actual and reconstructed heights and widths.

	True	Package A	Errors A (result-true)	Package B	Errors B (result-true)
h_{left} (nm)	79.5	77.9	-1.6	81.8	2.3
h_{right} (nm)	79.4	78.7	-0.7	75.8	-3.6
w (nm)	49.6	48.7 ± 0.3	-0.9 ± 0.3	49 ± 1	-0.6 ± 1
σ_{top} (nm)	1.8	1.3	-0.5	1.5	-0.3
$\sigma_{\text{sidewalls}}$ (nm)	0.9	0.4	-0.5	1.0	0.1