MAXIMUM LIKELIHOOD ESTIMATION OF CLOUD HEIGHT FROM MULTI-ANGLE SATELLITE IMAGERY.

By E. Anderes,* B. Yu,† V. Jovanovic, C. Moroney, M. Garay, A. Braverman, E. Clothiaux

University of California at Berkeley, Jet Propulsion Laboratory, Pennsylvania State University.

We develop a new estimation technique for recovering depth-of-field from multiple stereo images. Depth-of-field is estimated by determining the shift in image location resulting from different camera viewpoints. When this shift is not divisible by pixel width, the multiple stereo images can be combined to form a super-resolution image. By modeling this super-resolution image as a realization of a random field, one can view the recovery of depth as a likelihood estimation problem. We apply these modeling techniques to the recovery of cloud height from multiple viewing angles provided by the MISR instrument on the Terra Satellite. Our efforts are focused on a two layer cloud ensemble where both layers are relatively planer, the bottom layer is optically thick and textured, and the top layer is optically thin. Our results demonstrate that with relative ease, we get comparable estimates to the M2 stereo matcher which is the same algorithm used in the current MISR standard product (details can be found in [10]). Moreover, our techniques provide the possibility of modeling all of the MISR data in a unified way for cloud height estimation. Future research is underway to extend this framework for fast, quality global estimates of cloud height.

1. Introduction. The motivation for this paper comes from the problem of recovering cloud height from the Multi-Angle Imaging SpectroRadiometer (MISR) instrument, launched in December 1999 on the NASA EOS Terra Satellite. Clouds play a major role in determining the Earth's energy budget. As a result, monitoring and characterizing the distribution of clouds becomes important in global studies of climate. The MISR instrument produces images (275 m resolution) in the red band over a swath width of 360 km for nine camera angles: $70^o, 60^o, 45.6^o, 26.1^o$ forward angles, a nadir view at 0°, and aft angles $26.1^o, 45.6^o, 60^o, 70^o$ (referred to as Df, Cf, Bf, Af, An, Aa, Ba, Ca, and Da respectively). By taking advantage of the image displacement that results from multi-angle image geometry, one can recover cloud top height and cloud motion (where cloud motion is determined from wind). Unfortunately, transparency, multiple layers, occlusion and height discontinuities present challenges for cloud height estimation. In this paper we apply

new statistical techniques for estimating cloud height and attempt to recover cloud height for a two layer ensemble: an optically thin top layer over a textured bottom layer.

The image displacement that results from ground registration is, almost always, not divisible by the pixel width. By taking advantage of this offset one can construct a super-resolution image from the different viewing angles. It is this super resolution image that we model as a discrete sample from a realization of a continuous Gaussian random field. Under this paradigm, estimating height and wind can be viewed as a statistical likelihood estimation problem. There are several advantages of this approach. First, by changing the model of the latent continuous image, one can change the matching characteristics of the algorithm and, potentially, optimize the matching for different cloud ensembles. Second, the super-resolution framework extends naturally when there are more than two camera angles. Finally, the modeling of the super-resolution image gives a unified way of estimating sub-grid-scale displacement.

There is a considerable amount of existing literature on both the recovery of three dimensional structure from multiple stereo images and constructing super-resolution images from multiple stereo images. The literature on both problems is vast and spans over at least 20 years. We refer readers to two reviews [2], [5] on the recovery of depth-of-field and two reviews [7], [13] on the super-resolution problem. To the authors' knowledge, the two problems have yet to be considered concurrently for recovery of depth-of-field which is the focus of the current paper.

The rest of the paper is organized as follows. In Section 2 we describe our new technique for estimating depth-of-field from multiple images taken from different viewpoints. In Section 3 we show how to use this super-resolution framework for estimating cloud height from the MISR data. Finally, Section 4 presents our test results for cloud height estimation.

2. The Super-Resolution Likelihood .   In this section we describe our technique for recovering depth-of-field from multiple stereo images. We start with our notation for non-uniformly sampled images and finish with a presentation of our new estimation methodology that uses super-resolution techniques and random field models to define a likelihood for estimating depth-of-field. This exposition is done as general as possible to avoid letting the details of the cloud height problem obstruct the general estimation procedure (the details of the cloud height estimation are then presented in Section 3).

2.1. *Image Notation.*   Even though it is easy to visualize the construction of a super-resolution image, the mathematical notation to express this construction is somewhat clumsy. The basic object for our notation is an image, denoted $(\boldsymbol{x}, \boldsymbol{y})$. The pixel locations are encoded in the ordered list of spatial coordinates $\boldsymbol{x}$ and the corresponding gray values, or radiances, are encoded in the vector $\boldsymbol{y}$. In the regression setting, one might consider $\boldsymbol{x}$ to be a matrix with two columns where each row is a pixel location. However, we prefer to use the 'list' characterization so that, for example, a function $f \colon \mathbb{R}^2 \to \mathbb{R}$ evaluated at a list of locations $\boldsymbol{x}$, denoted $f(\boldsymbol{x})$, will represent component-wise evaluation of f on each element of the list (rather than row-wise evaluation using the regression notation). In this way, shifting all the spatial locations in $\boldsymbol{x}$ by the same $\delta \in \mathbb{R}^2$,

can be written $\boldsymbol{x} + \delta$.

The pixel locations $\boldsymbol{x}$ allow us to define non-uniformly sampled images. Therefore, when working with one image, $\boldsymbol{x}$ serves to specify the overall structure of the image. When working with multiple images, the individual pixel locations may be useful for comparing locations across multiple images. For example, if one has n images on a square grid of equal size, it may make sense to set the lower left pixel at the origin, all with the same pixel spacing. Multiple images will be written with superscripts $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}), \ldots, (\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)})$, reserving subscripts to denote the elements of a list. In particular, $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$ denotes the jth pixel and the corresponding j^{th} gray value or radiance. For example, a 2×2 gray level image could be written $\boldsymbol{x} = ((0,0),(0,1),(1,0),(1,1))$, $\boldsymbol{y} = (0.4, 0.2, 0.1, 0.7)^T$ so that the second pixel has coordinates $\boldsymbol{x}_2 = (0,1)$ with gray value $\boldsymbol{y}_2 = 0.2$.

Now the super-resolution image is defined by direct concatenation of the pixel locations and the gray values. In particular, let $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ and $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)})$ be two images which we want to overlay to construct a super-resolution image. The intuition is that these two images are of the same physical object but projected on different pixel grids (see Figure 2). The super-resolution image is defined as $\left(\begin{pmatrix} \boldsymbol{x}^{(1)} \\ \boldsymbol{x}^{(2)} \end{pmatrix}, \begin{pmatrix} \boldsymbol{y}^{(1)} \\ \boldsymbol{y}^{(2)} \end{pmatrix} \right)$. It may be the situation that the first image $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ needs to be shifted by some vector $\delta \in \mathbb{R}^2$ before it is overlaid with the second image $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)})$. The process of shifting the first image by δ then overlaying the two to construct a super-resolution image is written $\left(\begin{pmatrix} \boldsymbol{x}^{(1)} + \delta \\ \boldsymbol{x}^{(2)} \end{pmatrix}, \begin{pmatrix} \boldsymbol{y}^{(1)} \\ \boldsymbol{y}^{(2)} \end{pmatrix} \right)$.

2.2. *Super-resolution, Random Fields and Depth-of-field.* We start with n images, each representing different pictures of the same object taken from n different camera viewpoints (see the first illustration of Figure 1). Since the cameras are from different viewpoints, the object will appear in different image locations in each camera (see the second illustration of Figure 1). We define *parallax* as the difference in image location of the object in each camera. Notice that once the geometry of the camera configuration is fixed, the parallax is completely characterized by the distance, d, of the object to one of the cameras. Typical estimation of d (i.e. depth-of-field) amounts to estimating the parallax of the object between each pair of cameras then using these parallaxes to solve for d. Our method, on the other hand, estimates d directly, treating it as an unknown statistical parameter. The advantage is that d completely characterizes the parallax observed between each pair of cameras, thereby reducing the problem to estimating a single parameter and modeling the observations jointly rather than pairwise.

In more detail, consider a physical object captured in the image patch $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ and let $d \in (0, \infty)$ denote the distance of this object to the first camera. As one varies d, there will exist different image patches $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)})$, from the subsequent camera angles, all of which capture the object appearing in $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$. This implies the existence of shifts $\delta_2, \ldots, \delta_n$
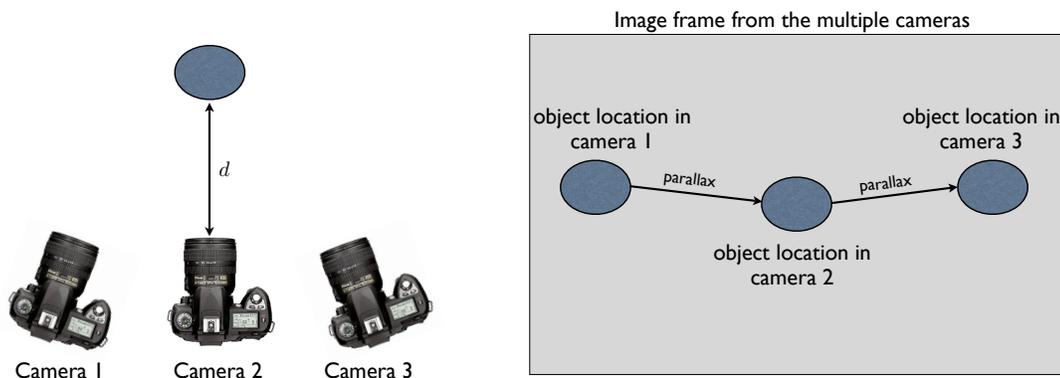
Fig 1. *An illustration of the distance parameter d and parallax. Once the geometry of the camera array is fixed the parameter d completely determines the parallax.*

so that the new super-resolution image, $(\boldsymbol{x}, \boldsymbol{y})$, given by

$$(1) \qquad \boldsymbol{x} := \begin{pmatrix} \boldsymbol{x}^{(1)} \\ \boldsymbol{x}^{(2)} + \delta_2 \\ \vdots \\ \boldsymbol{x}^{(n)} + \delta_n \end{pmatrix} \qquad \boldsymbol{y} := \begin{pmatrix} \boldsymbol{y}^{(1)} \\ \vdots \\ \boldsymbol{y}^{(n)} \end{pmatrix},$$

is a picture of the same object captured by $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ (see Figure 2). Note that the shifts $\delta_2, \ldots, \delta_n$ and the image patches $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)})$ depend solely on the distance parameter d. We call the process of combining image patches to construct a super-resolution image, interlacing.

To compute a likelihood for estimating the parameter d, we start by supposing there exists a latent continuous function $Y \colon \mathbb{R}^2 \to \mathbb{R}$ that models the continuous image in the local patch $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$. In other words, $Y(\boldsymbol{x}^{(1)}) = \boldsymbol{y}^{(1)}$. Note that $Y(\boldsymbol{x})$ denotes the list obtained by evaluating Y at each pixel location in the list $\boldsymbol{x}$. For the true distance parameter d, the super-resolution image $(\boldsymbol{x}, \boldsymbol{y})$ will also satisfy $Y(\boldsymbol{x}) = \boldsymbol{y}$. The wrong distance parameter will interlace images of different regions which will result in a 'noisy' super-resolution image for which it will be difficult to find a continuous interpolator that satisfies $Y(\boldsymbol{x}) = \boldsymbol{y}$. By putting a probability measure $\mathbb{P}$ on the latent continuous image Y, we can estimate the distance of the object by minimizing the following negative log likelihood:

$$(2) \qquad -\ell(d) := -\log \mathbb{P}\big(Y(\boldsymbol{x}) = \boldsymbol{y}\big),$$

where the super-resolution image $(\boldsymbol{x}, \boldsymbol{y})$, depending on d, is constructed as in (1).

An equivalent way to specify the log-likelihood (2) is to simply claim that there exists image patches $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)})$ and location shifts $\delta_2, \ldots, \delta_n$, all depending on the parameter

Image frame from the multiple cameras
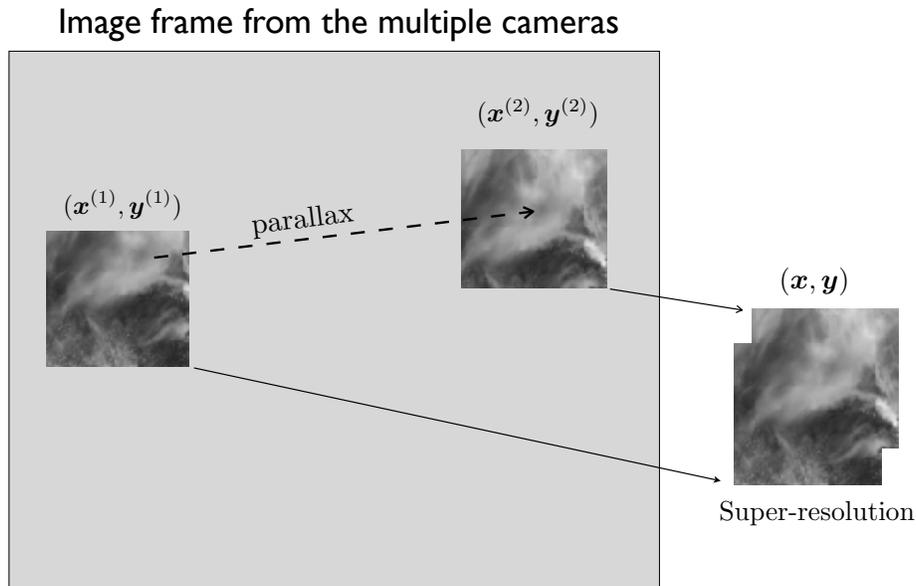
FIG 2. *An illustration of the construction of the super-resolution image* $(\boldsymbol{x}, \boldsymbol{y})$. *The object captured by* $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ *shifts to* $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)})$ *in the next camera angle. If the parallax is known and is not a multiple of the grid spacing, the two image patches can be overlaid to create a super-resolution image.*

d, such that $Y(\boldsymbol{x}^{(1)}) = \boldsymbol{y}^{(1)}$ and

$$(3) \qquad \boldsymbol{y}^{(k)} = Y(\boldsymbol{x}^{(k)} + \delta_k), \qquad k = 2, \ldots, n.$$

Since Y doesn't depend on k in the above equation, the super-resolution image defined by (1) satisfies $\boldsymbol{y} = Y(\boldsymbol{x})$. Therefore, under the model $\mathbb{P}$, the joint distribution of $\boldsymbol{y}$ is the same as that obtained in (2). This alternative way of expressing the super-resolution likelihood will become useful in the following sections for introducing nuisance parameters associated with the MISR cameras.

3. Cloud Height Estimation. This section contains the modeling details for cloud height estimation from MISR satellite data. In our view, one of the advantages of the estimation techniques outlined in the previous section is the ease in which the technical details of a particular observation scenario can be incorporated into the recovery of depth-of-field. We demonstrate this flexibility in our application of this methodology to cloud height estimation. This section starts with a discussion of the relationship between parallax, cloud height and wind. Section 3.1 presents

a summary of all the modeling assumptions used to compute the maximum likelihood estimation of height. A more thorough explanation of the modeling assumptions and the techniques for computing the likelihood can be found in Sections 3.1.1, 3.1.2 and 3.1.3.

For the MISR data, instead of estimating distance to the satellite, we want to estimate cloud height and wind. Since the parallax of a cloud patch is completely determined by height and wind, the techniques from Section 2 are easily applicable. Indeed, the one of the main features of the methodology developed in the last section is that it relates all the observed parallaxes to a single parameter d. In the MISR data, the parameter d, i.e. distance of the object to the satellite, is directly related with cloud height (since the height of a cloud determines the distance to the satellite). However, there is an added complication of cloud movement as a function of wind. Since wind will also effect the image location of a cloud in each camera, we include it as a parameter so that now height h and a wind velocity $\boldsymbol{v} := (v_1, v_2)$ completely determine parallax and the construction of the super-resolution image $(\boldsymbol{x}, \boldsymbol{y})$.

For the MISR images there is an approximate linear relationship that relates wind, cloud height and image parallax. Let $t^{(k)}$ denote the fly-over time delay (in seconds) for the kth camera and $x^{(k)}$ the along-track image location of a local cloud region in camera image k. When h is in meters, $\boldsymbol{v}$ is in meters per second, and the pixel locations specified by $\boldsymbol{x}^{(k)}$ all represent the same 275 meter grid, the linear equation that relates wind, height and along-track parallax are

$$(4) \qquad v_2(t^{(i)} - t^{(j)}) + h(\tan(\theta^{(i)}) - \tan(\theta^{(j)})) = x^{(i)} - x^{(j)}$$

where $\theta^{(k)}$ is the angle of each camera. The across-track parallax is given by $v_1(t^{(i)} - t^{(j)})$. For more details see [6].

3.1. *Detailed Summary of the Model.*   A major obstacle in using the methods in Section 2 for the MISR data is that different images corresponding to different angles often have different overall brightness. Therefore it becomes difficult to directly interlace patches from different images. In what follows we model the brightness change, due to the different angles, as a linear correction for each camera.

Suppose the image patch $(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ captures a small cloud region and let Y be the continuous representation of this image so that $Y(\boldsymbol{x}^{(1)}) = \boldsymbol{y}^{(1)}$. The true height and wind, $(h, \boldsymbol{v})$, allow us to retrieve patches $(\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)}), \ldots, (\boldsymbol{x}^{(n)}, \boldsymbol{y}^{(n)})$ from the subsequent camera angles, each capturing the same cloud region projected on potentially shifted grids. For the remainder of the paper, n denotes the number of cameras and m denotes the number of pixels in each patch $(\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)})$, $k = 1, \ldots, n$. Our model for the gray values $\boldsymbol{y}^{(k)}$ is summarized by

$$(5) \qquad \boldsymbol{y}^{(k)} = \sigma_k Y(\boldsymbol{x}^{(k)} + \delta_k) + A_k \boldsymbol{x}^{(k)} + b_k$$

for cameras $k = 2, \ldots, n$, where

- Y is the latent continuous cloud image, modeled by a Gaussian random field with Matérn covariance function $\mathcal{K}$ with parameters $(\sigma, \rho, \nu) = (1, 4, 4/3)$ (see Section 3.1.1) so that

$$\text{cov}(Y(\boldsymbol{t}), Y(\boldsymbol{s})) = \mathcal{K}(|\boldsymbol{t} - \boldsymbol{s}|),$$

for all $\boldsymbol{t}, \boldsymbol{s} \in \mathbb{R}^2$ where $|\cdot|$ denotes Euclidean distance;

- σ_k is a multiplicative brightness correction for each camera;
- δ_k is a two dimensional shift vector which is completely determined by $(h, \boldsymbol{v})$ and plays the role of shifting the pixel locations $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}$ so that they are interlaced;
- A_k is a 2×1 matrix which represents an affine brightness correction for each patch;
- b_k is an overall constant additive correction.

Notice that Y models the latent cloud image for each camera so we obtain a joint model for the full super-resolution vector $\boldsymbol{y} := (\boldsymbol{y}^{(1)\,T}, \ldots, \boldsymbol{y}^{(n)\,T})^T$. By the Gaussian assumption on Y we have

$$(6) \qquad\qquad \boldsymbol{y}^{(k)} \sim \mathcal{N}(A_k \boldsymbol{x}^{(k)} + b_k, \sigma_k^2 \Sigma_k)$$

where $\Sigma_k := \big(\mathcal{K}(|\boldsymbol{x}_i^{(k)} - \boldsymbol{x}_j^{(k)}|)\big)_{ij}$.

Since we have introduced an additive and multiplicative brightness correction on each patch, it is no longer true that the super-resolution image satisfies $\boldsymbol{y} = Y(\boldsymbol{x})$, where $\boldsymbol{x}$ is defined by (1). What is true, however, is that $\boldsymbol{y} = diag(\sigma_1 I_m, \ldots, \sigma_n I_m) Y(\boldsymbol{x}) + \mu$, where μ is a vector with k^{th} block $A_k \boldsymbol{x}^{(k)} + b_k$ and I_m is the $m \times m$ identity matrix. The main difficulty for estimation of $(h, \boldsymbol{v})$ now becomes devising techniques for dispensing with the nuisance parameters σ_k, A_k, b_k. This is the main contribution of Sections 3.1.2 and 3.1.3. Both sections use REML techniques (see [15]) to filter out the additive brightness corrections $A_k \boldsymbol{x}^{(k)} + b_k$. Dealing with the multiplicative nuisance parameters σ_k is more difficult and requires separate treatments for high and low clouds. For high clouds, a brightness stabilization allows us to marginalize out σ_k. For low clouds, there is no closed form solution for the marginalized likelihood and we resort to an approximate profile likelihood.

3.1.1. *Gaussian Random Field model for* $\mathbb{P}$. In (5) we use a Matérn autocovariance function $\mathcal{K}$ (with parameters $(\sigma, \rho, \nu) = (1, 4, 4/3)$) to model the covariance structure of the latent continuous cloud image Y. This section outlines our motivation for our choice of the Matérn autocovariance function and the specific parameter values. The basic idea is to use the observed fractal behavior of clouds (see [3], [1], [16], [12], [4] from the atmospheric science literature) to model the fractal nature of Y after adjusting for pixel averaging. Indeed, one of the advantages of the the Matérn autocovariance function is the flexibility it provides for modeling the fractal behavior of Y.

The Matérn autocovariance function $\mathcal{K}$ is defined by

$$\mathcal{K}(r) = \frac{\sigma}{2^{\nu-1}\Gamma(\nu)} \left(\frac{2\nu^{1/2}r}{\rho}\right)^\nu \mathcal{K}_\nu \left(\frac{2\nu^{1/2}r}{\rho}\right),$$

where $\mathcal{K}_\nu$ is the modified Bessel function (see [15]). We model Y, the continuous cloud image, as a two dimensional Gaussian random field with covariance structure given by $\mathrm{cov}(Y(s), Y(t)) = \mathcal{K}(|s - t|)$ for all $s, t \in \mathbb{R}^2$ with parameters $(\sigma, \rho, \nu) = (1, 4, 4/3)$. The parameter σ is the variance $\mathrm{var}[Y(t)]$ at any fixed point $t \in \mathbb{R}^2$. The parameter ρ serves as the range where the correlation

of the random field effectively becomes zero. The ν parameter controls the smoothness of the process.

In our analysis the range parameter ρ is set to 1100 meters (4 pixel widths) and the variance parameter σ is 1. Different values of ρ, σ don't severely effect our height estimates with the exception of either very small or very large ρ. In the case of very large ρ it seems that the local patch size becomes too small, compared to the range, for appropriate modeling. When ρ is very small, the estimation of height and wind becomes ineffective since most of the interlaced images are modeled as nearly uncorrelated.

Here we derive our justification for $\nu = 4/3$ as a plausible value for the smoothness parameter. In our analysis, the vector $\boldsymbol{y}^{(k)}$ represents log of registered bidirectional reflectance factor (BRF) values from the kth camera. Let I_{BRF} denote the latent continuous BRF image so that $Y = \log(\varphi * I_{\text{BRF}})$, where the pixelation is represented by the convolution kernel $\varphi = \mathbb{I}_{[-1/2,1/2]^2}$ for the indicator function $\mathbb{I}$. Let $\mathcal{K}_{\text{BRF}}$ be the autocovariance function for the continuous BRF image I_{BRF}. The two dimensional Fourier transform of $\mathcal{K}_{\text{BRF}}$ gives the spectral density

$$f_{I_{\text{BRF}}}(\omega) := \frac{1}{(2\pi)^2} \int_{\mathbb{R}^2} \exp(-ix^T\omega)\mathcal{K}_{\text{BRF}}(x)dx,$$

where $x = (x_1, x_2)$ and $\omega = (\omega_1, \omega_2)$ are both two dimensional vectors. There is evidence to suggest that Kolmogorov's 5/3 scaling law holds for $f_{I_{\text{BRF}}}$ (see [3], [1], [16], [12], [4]). Under the additional assumption of isotropy, the scaling law implies the two dimensional spectral density satisfies $f_{I_{\text{BRF}}}(\omega) \asymp \frac{1}{|\omega|^{5/3+1}}$ as $|\omega| \to \infty$. Therefore, the spectral density of the pixelated process $\varphi * I_{\text{BRF}}$ is given by

$$(7) \qquad f_{\varphi*I_{\text{BRF}}}(\omega) = |\hat{\varphi}(\omega)|^2 f_{I_{\text{BRF}}}(\boldsymbol{\omega}) \asymp \text{sinc}^2\left(\frac{\omega_1}{2}\right)\text{sinc}^2\left(\frac{\omega_2}{2}\right)\frac{1}{|\omega|^{5/3+1}},$$

as $|\omega| \to \infty$. Let f_Y denote the spectral density for the Matérn autocovariance model for Y. We believe a plausible value for the parameter ν, for the random field $Y = \log(\varphi*I_{\text{BRF}})$ corresponds to matching the power-law decay of f_Y to that of $f_{\varphi*I_{\text{BRF}}}$, in the coordinate directions. In particular, by (7) and the properties of the Matérn spectral density,

$$f_{\varphi*I_{\text{BRF}}}(\omega) \asymp \frac{\sin^2(\omega_1/2)}{|\omega_1|^{5/3+3}}, \qquad f_Y(\omega) \asymp \frac{1}{|\omega_1|^{2\nu+2}},$$

fixing ω_2 and letting $\omega_1 \to \infty$. Therefore, to match the decay, we set the smoothness parameter ν to $(5/3 + 1)/2 = 4/3$.

3.1.2. *Low cloud likelihood.* Here we give some of the computational techniques for dealing with the nuisance parameters when estimating the height of low textured clouds. We start by constructing a matrix L ('L' for low) which filters out the dependence of the observations on A_k

and b_k. Then we construct an approximate profile likelihood to handle the multiplicative nuisance parameters σ_k.

For low textured clouds we begin by constructing n matrices $L_1, \ldots, L_n$ which annihilate monomials of order at most 1 so that $L_k \boldsymbol{y}^{(k)} \sim \mathcal{N}(0, \sigma_k^2 \Sigma_k)$. In particular, L_k is a $(m-3) \times m$ matrix with rows composed of linearly independent vectors in the kernel space $\{v: [\phi_0(\boldsymbol{x}^{(k)}) \ \phi_1(\boldsymbol{x}^{(k)}) \ \phi_2(\boldsymbol{x}^{(k)})]^T v^T = \boldsymbol{0}\}$, where ϕ_0, ϕ_1 and ϕ_2 are the the monomials of order at most 1. Now define the matrix $L = \mathrm{diag}(L_1, \ldots, L_n)$ so that $L\boldsymbol{y} \sim \mathcal{N}(0, \Delta_\sigma \tilde{\Sigma} \Delta_\sigma)$, where $\tilde{\Sigma} := L\Sigma L^T$, $\Sigma := (\mathcal{K}(|\boldsymbol{x}_i - \boldsymbol{x}_j|))_{ij}$ and $\Delta_\sigma := diag(\sigma_1 I_{m-3}, \ldots, \sigma_n I_{m-3})$. Therefore the likelihood of the observation vector $L\boldsymbol{y}$, as it depends on $(h, \boldsymbol{v})$ and $\boldsymbol{\sigma} := (\sigma_1, \ldots, \sigma_n)^T$, can be written

$$\mathcal{L}(h, \boldsymbol{v}, \boldsymbol{\sigma}|L\boldsymbol{y}) = \frac{1}{|2\pi \tilde{\Sigma}|^{1/2}} \frac{1}{|\Delta_\sigma|} \exp\left[-\frac{1}{2}(L\boldsymbol{y})^T \Delta_\sigma^{-1} \tilde{\Sigma}^{-1} \Delta_\sigma^{-1}(L\boldsymbol{y})\right].$$

At this point there are a few options to dispense with the dependence of the above likelihood on the nuisance parameters $\boldsymbol{\sigma}$. The most desirable option would be to marginalize out $\boldsymbol{\sigma}$ by integrating $\int_{\mathbb{R}_+^n} \mathcal{L}(h, \boldsymbol{v}, \boldsymbol{\sigma}|L\boldsymbol{y}) d\boldsymbol{\sigma}$. For even moderate block size m and $n = 2$, this becomes computationally formidable. The second option would be to maximize a profile likelihood $\mathcal{L}(h, \boldsymbol{v}|L\boldsymbol{y}) := \max_{\boldsymbol{\sigma} \in \mathbb{R}_+^n} \{\mathcal{L}(h, \boldsymbol{v}, \boldsymbol{\sigma}|L\boldsymbol{y})\}$. This option is somewhat more computationally tractable but still problematic since there is no closed form for the profile likelihood. The easiest option is to estimate σ_k on each patch separately, $\hat{\sigma}_k^2 := (L_k \boldsymbol{y}^{(k)})^T (L_k \Sigma_k L_k^T)^{-1}(L_k \boldsymbol{y}^{(k)})/n$, then maximizing a plug-in version of the likelihood $\mathcal{L}(h, \boldsymbol{v}, \hat{\boldsymbol{\sigma}}|L\boldsymbol{y})$, where $\hat{\boldsymbol{\sigma}} := (\hat{\sigma}_1, \ldots, \hat{\sigma}_n)^T$. Notice, however, that under the correct $(h, \boldsymbol{v})$ the patches $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(n)}$ are highly correlated, and therefore information is lost by estimating σ_k separately on each patch. In an attempt to alleviate this problem, we explore a compromise between the full profile likelihood and the overly simplistic case of the plug-in likelihood with $\hat{\boldsymbol{\sigma}}$.

Notice that the quadratic term $(L\boldsymbol{y})^T \Delta_\sigma^{-1} \tilde{\Sigma}^{-1} \Delta_\sigma^{-1}(L\boldsymbol{y})$ can be written

$$\|\tilde{\Sigma}^{-1/2} \Delta_\sigma^{-1} L\boldsymbol{y}\|^2 = \left\|\frac{R_1 L_1 \boldsymbol{y}^{(1)}}{\sigma_1} + \cdots + \frac{R_n L_n \boldsymbol{y}^{(n)}}{\sigma_n}\right\|^2 = \boldsymbol{\sigma}^{-T} \tilde{R} \boldsymbol{\sigma}^{-1},$$

where the matrices $R_1, \ldots, R_n$ decompose $\tilde{\Sigma}^{-1/2}$ into block form so that $\tilde{\Sigma}^{-1/2} = (R_1, \cdots, R_n)$ and $\tilde{R}$ is the matrix of inner products $\left(\left\langle R_i L_i \boldsymbol{y}^{(i)}, R_j L_j \boldsymbol{y}^{(j)}\right\rangle\right)_{ij}$. Therefore the log likelihood can be written

$$\ell(h, \boldsymbol{v}, \boldsymbol{\sigma}|L\boldsymbol{y}) = c_1 - \frac{1}{2}\log|\tilde{\Sigma}| - (m-3)\sum_{k=1}^n \log \sigma_k - \frac{1}{2}\boldsymbol{\sigma}^{-T} \tilde{R} \boldsymbol{\sigma}^{-1},$$

where c_1 is a constant. For a fixed $(h, \boldsymbol{v})$, maximizing ℓ over $\boldsymbol{\sigma}$ is a convex problem whose stationary point is characterized by

$$(8) \qquad\qquad\qquad \tilde{R}\boldsymbol{\sigma}^{-1} - (m-3)\boldsymbol{\sigma} = 0.$$

This equation defines the profile likelihood but unfortunately has no closed form solution for general n. However, there is considerable research on investigating iterative algorithms for solving (8) (see [9],[8], [11] for example). The problem with iterative algorithms is that the likelihood computation will be done many times while searching through height and wind over sliding blocks. Therefore, including an iterative algorithm in the likelihood calculation presents some computational problems. As a compromise we define a "one Newton step" estimate of the stationary point of (8) with an initial starting point $\hat{\boldsymbol{\sigma}}$, the maximum likelihood estimate $\sqrt{(L_k \boldsymbol{y}^{(k)})^T (L_k \Sigma_k L_k^T)^{-1} (L_k \boldsymbol{y}^{(k)})/n}$ from each camera separately.

The Newton step is constructed as in [8] and defined for $\boldsymbol{\sigma}^{-1}$ rather than $\boldsymbol{\sigma}$. In particular, define $F(\boldsymbol{\sigma}^{-1}) = \tilde{R}\boldsymbol{\sigma}^{-1} - (m-3)\boldsymbol{\sigma}$ and $\tilde{\Delta}_{\boldsymbol{\sigma}} := diag(\sigma_1, \ldots, \sigma_n)$. Now $F(\boldsymbol{\sigma}^{-1} + \boldsymbol{\tau}) = F(\boldsymbol{\sigma}^{-1}) + \tilde{R}\boldsymbol{\tau} + (m-3)\tilde{\Delta}_{\boldsymbol{\sigma}}^2 \boldsymbol{\tau}$ plus higher order terms in $\boldsymbol{\tau}$. Setting this linear approximation to zero and solving for $\boldsymbol{\tau}$ gives the Newton step $\boldsymbol{\sigma}^{-1} + \boldsymbol{\tau}$. We use the initial starting point $\hat{\boldsymbol{\sigma}}^{-1}$. Notice the Newton step may result in a negative variance. In which case, the original estimate $\hat{\boldsymbol{\sigma}}$ is used instead of the Newton step. In summary, to estimate $(h, \boldsymbol{v})$ we define the "one Newton step"

$$(\hat{\boldsymbol{\sigma}}_{\text{newton}})^{-1} := \begin{cases} \hat{\boldsymbol{\sigma}}^{-1} + (\tilde{R} + (m-3)\tilde{\Delta}_{\hat{\boldsymbol{\sigma}}}^2)^{-1}((m-3)\tilde{\Delta}_{\hat{\boldsymbol{\sigma}}}^2 - \tilde{R})\hat{\boldsymbol{\sigma}}^{-1}, & \text{if positive;} \\ \hat{\boldsymbol{\sigma}}^{-1}, & \text{otherwise,} \end{cases}$$

and maximize the plug-in log-likelihood

$$\widehat{(h, \boldsymbol{v})} := \arg\max_{(h, \boldsymbol{v})} \ell\left(h, \boldsymbol{v}, \hat{\boldsymbol{\sigma}}_{\text{newton}} \Big| L\boldsymbol{y}\right).$$

3.1.3. *High cloud likelihood.*   Here we give some of the computational techniques for dealing with the nuisance parameters when estimating the height of high thin clouds. Since these heights are notoriously hard to estimate, we attempt a brightness stabilization on the whole cloud image which allows more local information for estimating cloud height. After this stabilization, we construct a matrix H ('H' is for high) which filters out the common value of the stabilized nuisance parameters A_k and b_k. For the brightness corrections σ_k, our stabilization allows us to marginalize out the common value of σ_k under a uniform prior which is far more desirable than the approximate profile likelihood techniques needed for low clouds.

One of the difficulties of high clouds is that they are frequently optically thin or partially transparent. This makes estimating parallax difficult because the dominant signal is frequently the background rather than the cloud itself. In an attempt to overcome this difficulty we estimate a linear transformation on $\boldsymbol{y}^{(k)}$ for each camera $k = 1 \ldots, n$ to stabilize the brightness corrections modeled by σ_k, A_k and b_k. A more in-depth discussion on this issue is presented in Section 4. Once stabilization is achieved we have $\boldsymbol{y} \sim \mathcal{N}(A\boldsymbol{x} + b, \sigma^2 \Sigma)$, where σ, A and b are the common values of σ_k, A_k and b_k respectively and $\Sigma := \big(\mathcal{K}(|\boldsymbol{x}_i - \boldsymbol{x}_j|)\big)_{ij}$. This allows us to find a matrix H for which $H\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \sigma H \Sigma H^T)$. H is defined as the matrix with rows composed of linearly independent vectors in the kernel space $\{v \colon [\phi_0(\boldsymbol{x}) \ \phi_1(\boldsymbol{x}) \ \phi_2(\boldsymbol{x})]^T v^T = \boldsymbol{0}, \}$. Notice that now H is not block diagonal as it was for L. As a consequence $H\boldsymbol{y}$ is a vector of length $nm - 3$ versus

$(m-3)n$ (for the low cloud estimates) where n and m denote the number of cameras and patch size respectively. This provides more information for matching in an attempt at recovering the high cloud heights.

The likelihood of $h, \boldsymbol{v}$ and σ, given the observation vector $H\boldsymbol{y}$, has the following form

$$\mathcal{L}(h, \boldsymbol{v}, \sigma | H\boldsymbol{y}) = \frac{1}{\sigma^{nm-3}|2\pi\tilde{\Sigma}|^{1/2}} \exp\left[-\frac{\boldsymbol{y}^T H^T \tilde{\Sigma}^{-1} H\boldsymbol{y}}{2\sigma^2}\right],$$

where $\tilde{\Sigma} := H\Sigma H^T$. In contrast with the low cloud likelihood, we can remove the dependence on the unknown parameter σ by marginalizing out under a uniform improper prior

$$\mathcal{L}(h, \boldsymbol{v} | H\boldsymbol{y}) \propto \int_{\mathbb{R}_+} \mathcal{L}(h, \boldsymbol{v}, \sigma | H\boldsymbol{y}) \, d\sigma = \frac{2^{(nm-4)/2}}{2|2\pi\tilde{\Sigma}|^{1/2}} \frac{\Gamma\left(\frac{nm-4}{2}\right)}{\left[\boldsymbol{y}^T H^T \tilde{\Sigma}^{-1} H\boldsymbol{y}\right]^{\frac{nm-4}{2}}}.$$

The integral is obtained by the change of variables $x = \sigma_k^2$ and recognizing an un-normalized inverse gamma density. Therefore the log-likelihood which is maximized over $(h, \boldsymbol{v})$ is

$$(9) \qquad \ell(h, \boldsymbol{v} | H\boldsymbol{y}) = c_2 - \frac{1}{2}\log|\tilde{\Sigma}| - \frac{nm-4}{2}\log\left[\boldsymbol{y}^T H^T \tilde{\Sigma}^{-1} H\boldsymbol{y}\right],$$

for some constant c_2.

Remark: Instead of marginalizing out the unknown σ one could instead maximize a profile log-likelihood as a function of $(h, \boldsymbol{v})$, where the profiling is over the unknown σ. The resulting profile log-likelihood is $c_3 - \frac{1}{2}\log|\tilde{\Sigma}| - \frac{nm-3}{2}\log\left[\boldsymbol{y}^T H^T \tilde{\Sigma}^{-1} H\boldsymbol{y}\right]$ which, it appears, fails to take into account the loss of degrees of freedom associated with the unknown σ.

4. Test Results for Cloud Height Retrieval. This section shows some test results for estimating cloud height in a multi-layer cloud region. The test region is a cropped image from a MISR swath corresponding to orbit number 029145, path 031, and MISR blocks 56-67. Using the conventional grid, the upper left corner is located (row, column)=(801,1701) and the lower right corner is located (row, column)=(1400,2100) for a total image size of 600×400. Figure 3 shows the cloud images from the Bf and Cf cameras. This particular region was selected because of the clear nature of the two layer cloud ensemble. Near the right edge of the image, the region is dominated by optically thin, high clouds. The left edge contains mostly low, textured clouds. As one moves from left to right, the high clouds become more optically thick and the low clouds become sparse. This makes the region particularly useful for testing the ability to estimate the height of two distinct cloud layers.

In addition to the two layer structure, there is evidence to suggest minimal wind in both the along-track and across-track direction. This allows one to only estimate the height parameter and set the wind vectors to zero. In all of the test cases, the interlacing likelihood is maximized over a grid with resolution 100 meters over the interval $(0, 3 \times 10^4)$ meters. All of the following height
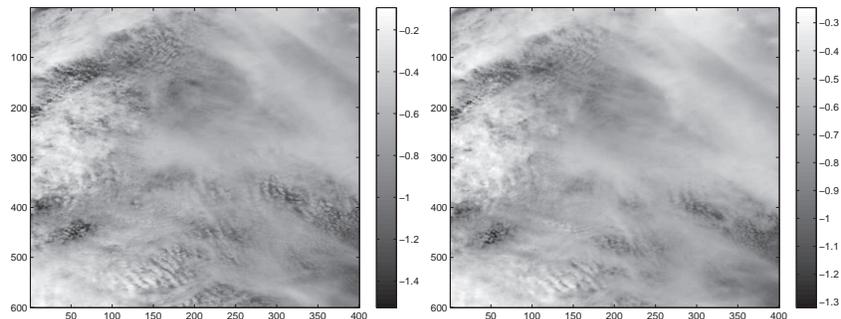
FIG 3. *Log BRF from the the Bf camera (left) and Cf camera (right).*

estimates are based on 15-by-16 sliding local patches, with the exception of the 15-by-15 patch size used in the MISR software estimates and the likelihood profiles found in Figure 7. The MISR estimates use an implementation of the M2 stereo matcher which is the same algorithm used in the current MISR standard product (details can be found in [10]).

Figure 4 shows the cloud height estimates based on the low cloud likelihood (top row) and the high cloud likelihood (bottom row) using two cameras (left column) and three cameras (right column). Figure 5 shows two of our best estimates of height along with the estimates from MISR's M2 algorithm. The left plot in Figure 5 shows the high cloud estimates based on the three cameras Bf, Cf and Df. The middle plot shows the low cloud estimates based on the three cameras Aa, An and Af. The right plot shows the MISR's M2 estimates based on two cameras Bf and Cf. Compared with MISR's M2 software one can see that the coverage (in other words, the percentage of good height estimates) is comparable. It is also apparent that the three estimates recover different components of the cloud layers. The most striking difference is the amount of high clouds recovered in the left plot compared to the other two. Presumably, a major factor in this recovery is the brightness stabilization across cameras.

Figure 6 shows the mean and variance, taken over each column of the transformed images, from cameras Bf, Cf and Df. The stabilizing transformations are found by visually matching the means and variances of the high clouds near the right edge of the test image. From these plots it seems that, at least for the test region, the high cloud region can indeed be stabilized as seen by similarity of the first two moments near the right edge, where the high clouds dominate. There is some reason to believe that such a stabilizing transformation exists for general cloud images. The radiance contribution from the high clouds is almost exclusively due to initial scattering, whereas the radiance contribution from the low clouds must pass though the high cloud region getting a reduction in both mean and variance depending on the optical thickness of the top layer. It is not so clear, however, whether the stabilizing transformation can be estimated from a multi-layer scene. In our test case we can take advantage of the fact that high clouds dominate the right hand edge and therefore the overall change in mean and variance can be estimated.

Remark: One possible way to automate the estimation of the stabilizing transformation is to use classification algorithms to find regions dominated mostly by high clouds, estimate the transformation on these regions, then extrapolate to the whole multi-layer scene. Indeed, there is existing literature on cloud detection which could potentially be adapted for finding these high cloud dominated regions in a multi-layer scene (see [14]).

To investigate the difference between the the low cloud likelihood and the high cloud likelihood, Figure 7 shows the log likelihood profiles as a function of cloud height for two patches from our test scene (both patches constitute a 15×15 pixel region). The first patch is centered at row 500 and column 100 and is dominated by low textured clouds. The second patch is centered at row 400 and column 275 and constitutes a region with a two layer cloud ensemble. In Figure 7 the top row shows plots of the low cloud likelihood (left/right plot corresponds to the low/multi-layer cloud patch). The bottom row corresponds to the high cloud likelihood (left/right plot corresponds to the low/multi-layer cloud patch). The discontinuous nature of the graphs is due to the fact that a fixed interlacing was used and therefore sub-grid-scale displacement can not be detected. The significance of these plots is that the likelihood profile changes significantly when using the likelihood for the high clouds (bottom row). The result is that the bottom layer height is estimated when using the low cloud likelihood and the top layer height is estimated when using the high cloud likelihood. One other interesting feature of these likelihood profiles is multi-modality. One explanation could be the presence of more than two cloud layers in the test region.

Summary.   We present a new depth-of-field algorithm which uses random field models and the concept of super-resolution. By viewing the multi-angle cloud images as discrete sub-samples of a continuous random random field, one can view depth-of-field estimation as a statistical parameter estimation problem. Under this paradigm new tools become available for recovering depth-of-field from multiple stereo images and, in some cases, improve sensitivity and allow fine tuning for different observation scenarios. We apply these techniques to the recovery of cloud height using the MISR instrument on the Terra spacecraft. In our application, we attempted to demonstrate the ease in which technical details of the stereo cameras and the scientific properties of the observations can be incorporated in the estimates. The main focus of the test case was to use the special nature of our new estimator to recover the heights of a two layer cloud ensemble: an optically thin high cloud layer and an optically thick, textured low cloud. We have shown that the recovery of two separate layers is indeed possible and could potentially be automated for cloud height estimates on a global scale. These results lay the foundation for future research on extending this framework for cloud height estimation on a global scale using all nine MISR cameras. Indeed current research is underway to speed up the likelihood estimates and to incorporate more information on the observational properties of the MISR cameras.

References.

[1] H. Barker and J. Davies. Solar radiative fluxes for stochastic, scale-invariant broken cloud fields. *Journal of the Atmospheric Sciences*, 49:1115–1126, 1992.
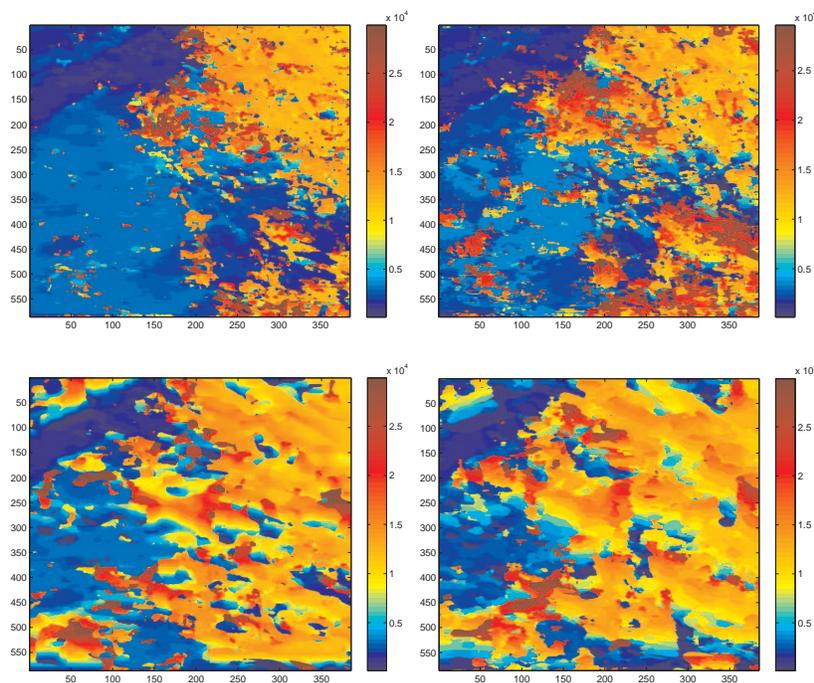
FIG 4. *Upper Left: Low cloud height estimates using the two cameras Bf and Cf. Upper Right: Low cloud height estimates using the three cameras Bf, Cf and Df. Lower Left: High cloud height estimates using the two cameras Bf and Cf. Lower Right: High cloud height estimates using the three cameras Bf, Cf and Df.*

[2] M. Brown, D. Burschka, and G. Hager. Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):993–1008, 2003.

[3] R. Cahalan and J. Snider. Marine stratocumulus structure. *Remote Sens. Environ.*, 28:95–107, 1989.

[4] A. Davis, A. Marshak, R. Cahalan, and W. Wiscombe. The landsat scale break in stratocumulus as a three-dimensional radiative transfer effect: Implications for cloud remote sensing. *Journal of the Atmospheric Sciences*, 54:241–260, 1997.

[5] U. Dhond and J. Aggarwal. Structure from stero—a review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, 1989.

[6] D. J. Diner, R. Davies, L. Di Girolamo, A. Horvath, C. Moroney, J.-P. Muller, S. R. Paradise, D. Wenkert, and J. Zong. Misr: Level 2 cloud detection and classification. Technical report, Jet Propul. Lab., Pasadena, CA. JPL Internal Doc. D-11399., 1997.

[7] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar. Advances and challenges in super-resolution. *International Journal of Imaging Systems and Technology*, 14(2):47–57, 2004.

[8] L. Khachiyan and B. Kalantari. Diagonal matrix scaling and linear programming. *SIAM J. Optimization*, 2:668–672, 1992.

[9] A. W. Marshall and I. Olkin. Scaling of matrices to achieve specified row and column sums. *Numer. Math.*, 12:83–90, 1968.

[10] J. Muller, A. Mandanayake, C. Moroney, R. Davies, D. Diner, and S. Paradise. Misr stereoscopic image matchers: Techniques and results. *IEEE Transactions on Geoscience and Remote Sensing*, 40(7):1547–1559,
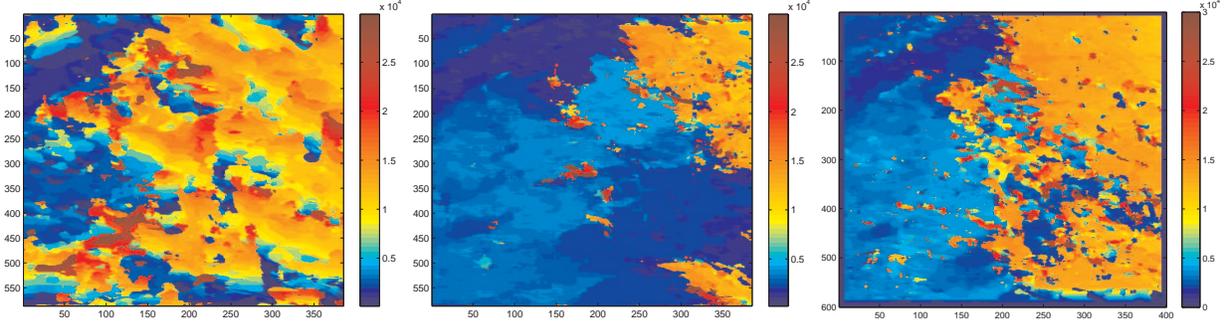
FIG 5. *Left: High cloud height estimates from three cameras Bf, Cf and Df. Middle: Low cloud height estimates from the three cameras Aa, An and Af. Right: Height estimates based on the MISR's M2 stereo matcher using cameras Bf and Cf.*
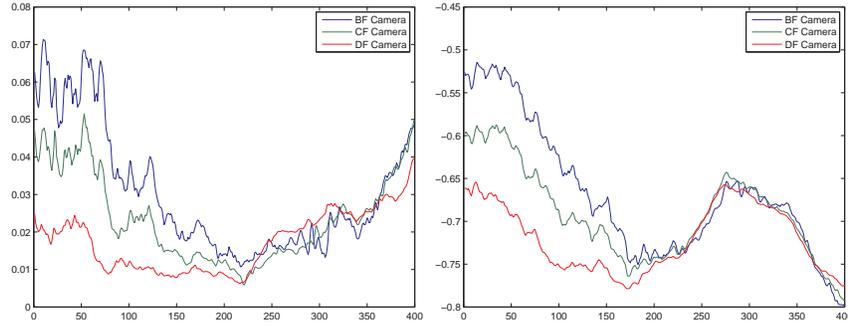
FIG 6. *Left: Variances of each column in the transformed images $\boldsymbol{y}^{(1)}$, $1.14\,\boldsymbol{y}^{(2)} + .03$ and $1.3\,\boldsymbol{y}^{(3)} + 0.09$ (where $\boldsymbol{y}^{(1)}$, $\boldsymbol{y}^{(2)}$, $\boldsymbol{y}^{(3)}$ denote the gray values from cameras Bf, Cf and Df respectively). Right: Mean of each column in the transformed image $\boldsymbol{y}^{(1)}$, $1.14\,\boldsymbol{y}^{(2)} + .03$ and $1.3\,\boldsymbol{y}^{(3)} + 0.09$.*

2002.

[11] D. P. O'Leary. Scaling symmetric positive definite matrices to prescribed row sums. *Linear Algebra and its Applications*, 370:185–191, 2003.

[12] L. Oreopoulos, A. Marshak, R. Cahalan, and G. Wen. Cloud three-dimensional effects evidenced in landsat spatial power spectra and autocorrelation functions. *Journal of Geophysical Research*, 105:14,777–14,788, 2000.

[13] S. Park, M. Park, and M. Kang. Super-resolution image reconstruction: a technical overview. *IEEE Signal Processing Magazine*, 20(3):21–36, 2003.

[14] T. Shi, B. Yu, E. Clothiaux, and A. Braverman. Daytime arctic cloud detection based on multi-angle satellite data with case studies. *Journal of The American Statistical Association. To appear.*

[15] M. L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging.* Springer, New York, 1999.

[16] T. Várnai. Influence of three-dimensional radiative effects on the spatial distribution of shortwave cloud reflection. *Journal of the Atmospheric Sciences*, 57:216–229, 2000.
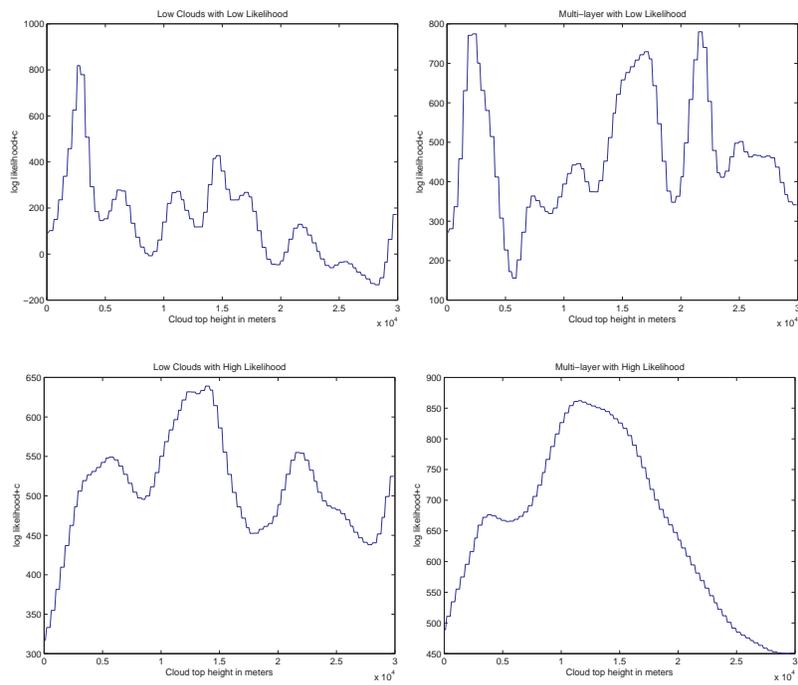
FIG 7. *Profiles of the log likelihood as a function of height. Rows correspond to different likelihood models (top: low cloud model, bottom: high cloud model) and the columns correspond to different patches (left: low cloud patch, right: multi-layer cloud patch).*

E. ANDERES
DEPARTMENT OF STATISTICS
UNIVERSITY OF CALIFORNIA AT BERKELEY
367 EVANS HALL
BERKELEY, CA 94720-3860
E-MAIL: anderes@stat.berkeley.edu