A Textural Approach to Snow Depth in the Weddell Sea

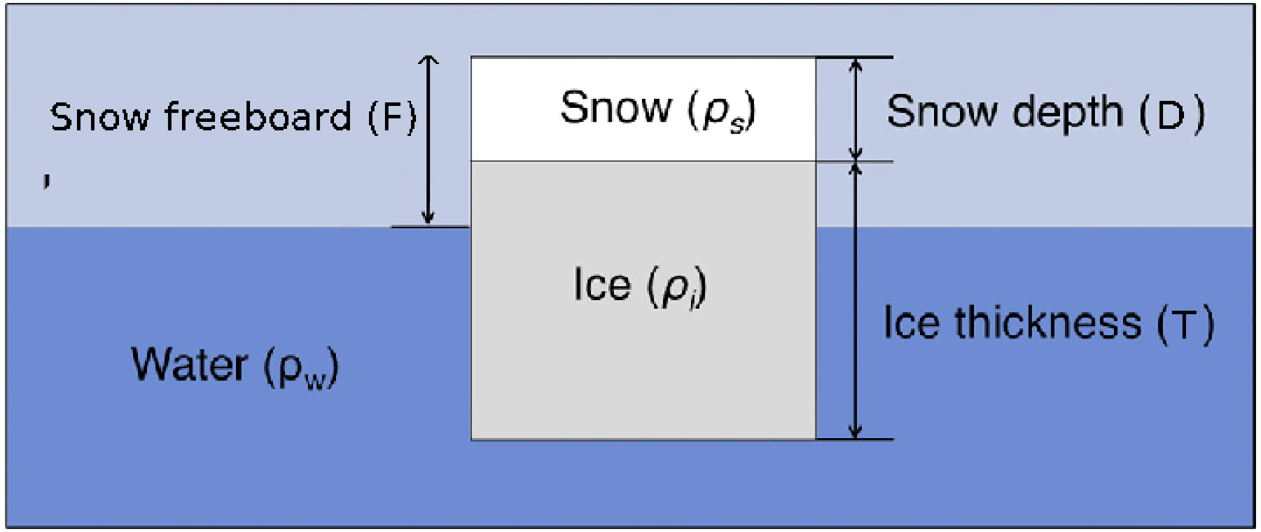
Jeffrey Mei & Ted Maksym
Woods Hole Oceanographic Institution

Contents

Introduction

- **What is the relationship between sea ice thickness and surface morphology?**
- Prior work from Mei et al. (2019) showed that sea ice thickness could be predicted from the surface elevation (lidar scan), at local (25 m) scales with 20% error.
- This is better than linear regressions that fit sea ice thickness to surface elevation directly, which may have errors up to 50% when applied to large-scale data from ICE-Sat (Kern & Spreen 2015).
- Can we generalize this to larger data sets?

Introduction



$$T = \frac{\rho_w}{\rho_w - \rho_i} F + \frac{\rho_s - \rho_w}{\rho_w - \rho_i} D$$

If we can predict the snow depth *D* from the snow freeboard *F*, we can essentially predict SIT directly (with the caveat that the density of snow is not necessarily constant, nor well-constrained).

Data

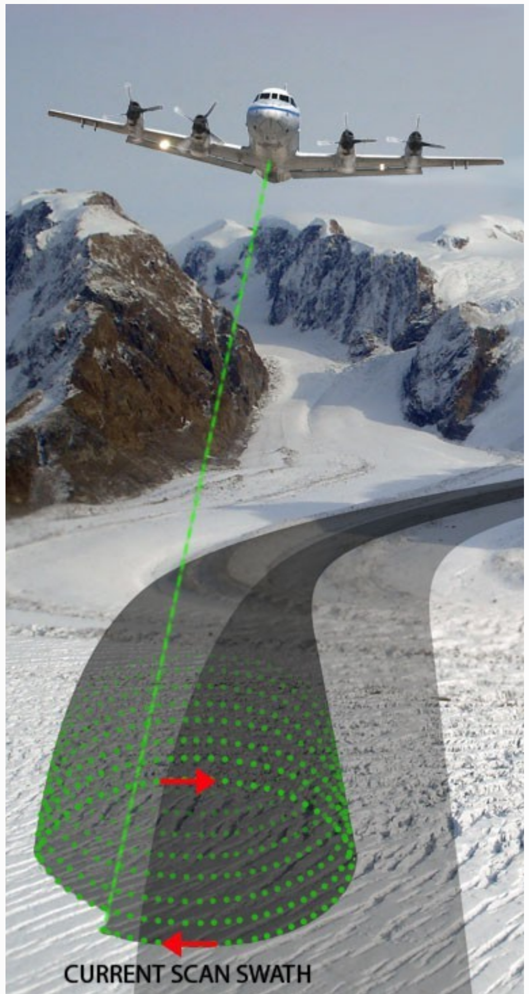


Fig 1. Operation IceBridge lidar surface elevation scan, showing the conical scanning lidar. Image credit: NASA

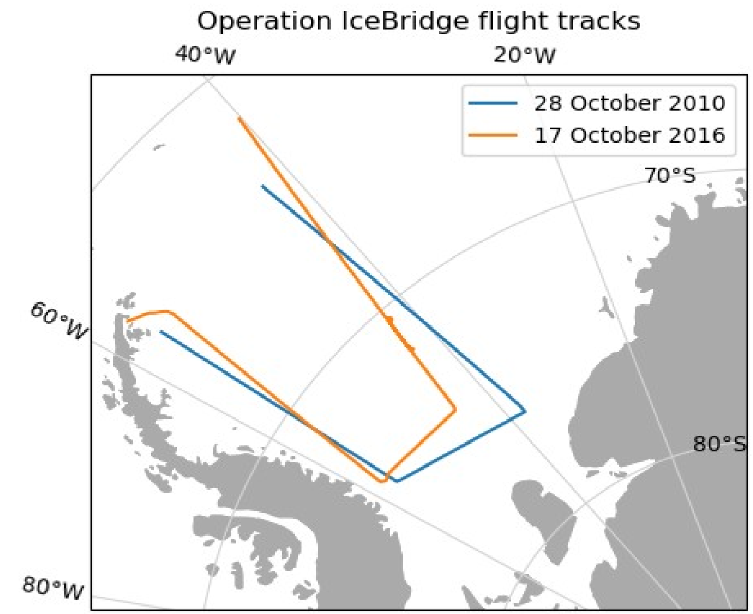


Fig. 2. Flight tracks used in this study, in the Weddell Sea.

Methods – lead processing

The data used here is the L1b Airborne Topographic Mapper from Operation IceBridge. This data is referenced to the WGS84 ellipsoid, so we first have to reference these to local sea surface height. To do this, we developed a lead finding algorithm that uses the orthorectified and geolocated camera imagery from OIB.

The lead heights are then used to correct the surface elevation data using an inverse distance weighting. The typical error for the lead finding is less than 3 cm.

For a demo of the lead finding algorithm click here.

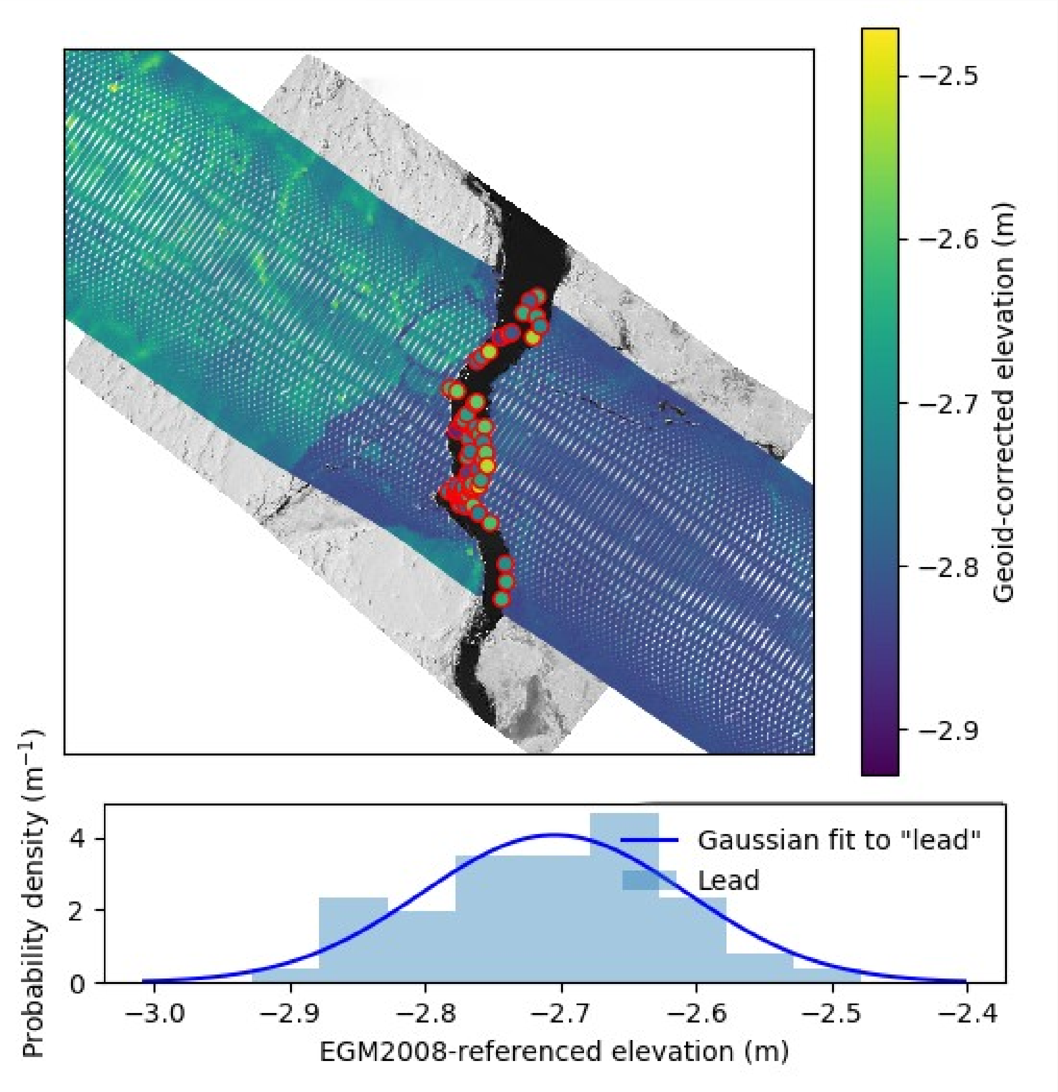


Fig 3. Leadfinding algorithm, in order to reference F to sea level correctly. Geoidal model (EGM2008-1) is only accurate to 1-4 m.

Preprocessing

This lead-referenced surface freeboard data is then interpolated onto a 180 x 180 grid (resolution 1m) using natural neighbor interpolation, as shown in Fig. 4. The snow depths for that segment are also shown (circles) – note that it is a nadir-looking data so the data is 1D. The mean snow depth of this lidar window is NOT necessarily the mean of the raw snow depth points for this window.

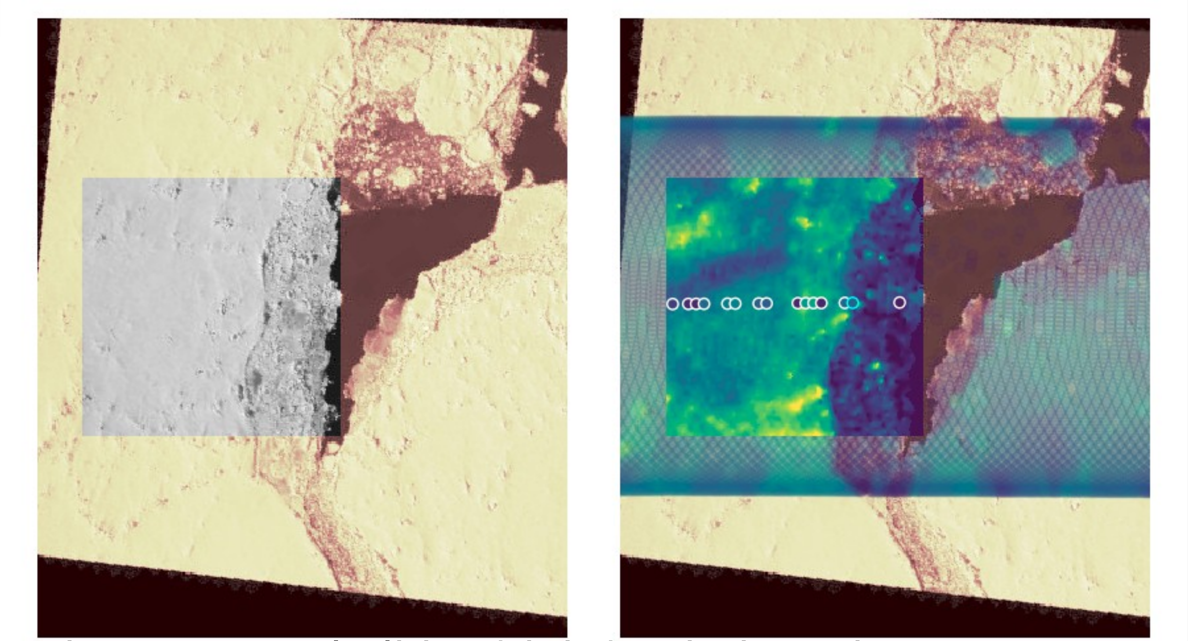


Fig. 4. Example lidar (right), windowed to 180 x 180 m, with snow depths (circles), and aerial photography (left). The **snow depths** are **only 1D.**

Methods - segmentation

To extrapolate the snow depths, we first have to segment the lidar scans into area of similar texture. We are assuming that the snow depth (in particular, the snow/ice ratio) is similar for texturally-similar areas). For example, we may expect that ridged surfaces to have a different snow distribution than flat surfaces, but to have a similar snow distribution to another (nearby) ridge.

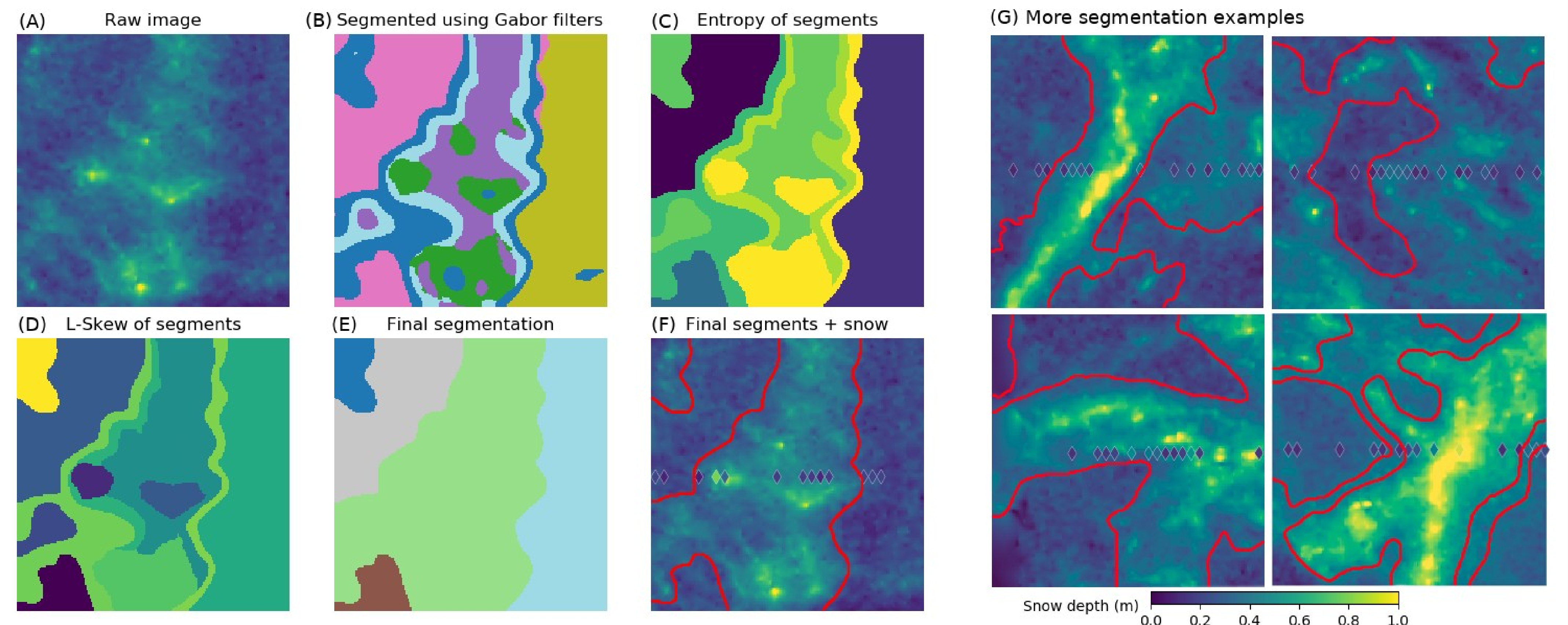


Fig 5. Segmentation algorithm. The lidar window is first segmented using Gabor filters (B), then adjacent segments with similar (within 2%) entropy** (C) and skew** (D) are merged together, until the no new mergings can be completed (E). This gives the final segments (F).

** these metrics were chosen by using a decision tree with manually-classified lidar+images.

Methods - extrapolation

Now we can use the segments to extrapolate the snow line, in order to get a more accurate mean snow depth corresponding to each 180 x 180 m lidar window.

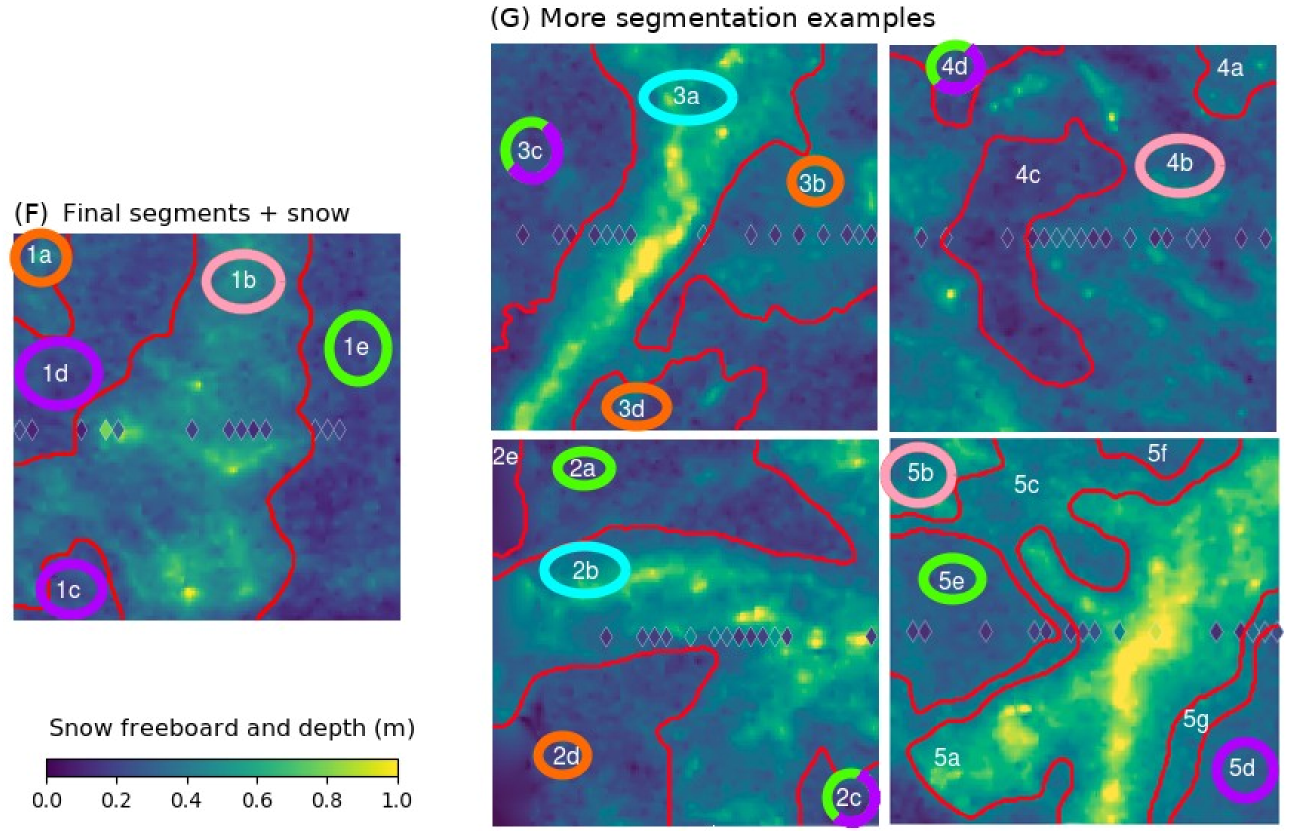


Fig 6. Segment-matching, color-coded for reference. For example, 1b is matched with 4b and 5b. A weighted average of the mean F/D ratios for 4b and 5b is taken (weighted by similarity and number of snow depth samples), and is applied to the mean F of 1b to get the extrapolated snow depth. Note that these are just randomly sampled lidar windows, as the full algorithm looks for all segments within ±10 km (up to 120 windows).

For a working example of the extrapolation method, see here.

Results

To check for biases and errors, we first try the extrapolation algorithm on the set of segments that do have snow depths on them (the ‘snow line’). For segments on this line, those that are successfully matched to another segment are called ‘completions’.

There are two main biases: **Extrapolation** bias & **Sampling** bias:

Extrapolation bias: red vs blue line – difference in mean, 0.5 cm. So the extrapolation algorithm is not that biased.

Sampling bias: blue vs green line – difference in mean, 4.4 cm. So, the snow depth sampled by OIB are likely biased (consistent with findings from Kwok & Maksym 2014).

The mean snow of the non-completions is 16 cm higher – probably because thicker surfaces are fewer in number (harder to match) and also are likely to be rougher, and less likely to give a snow depth return (Farrell 2012, Kwok 2011).

The mean relative error of the snow depth extrapolation is 22.5%. Now we can see if the mean snow depth can be related to the snow surface.

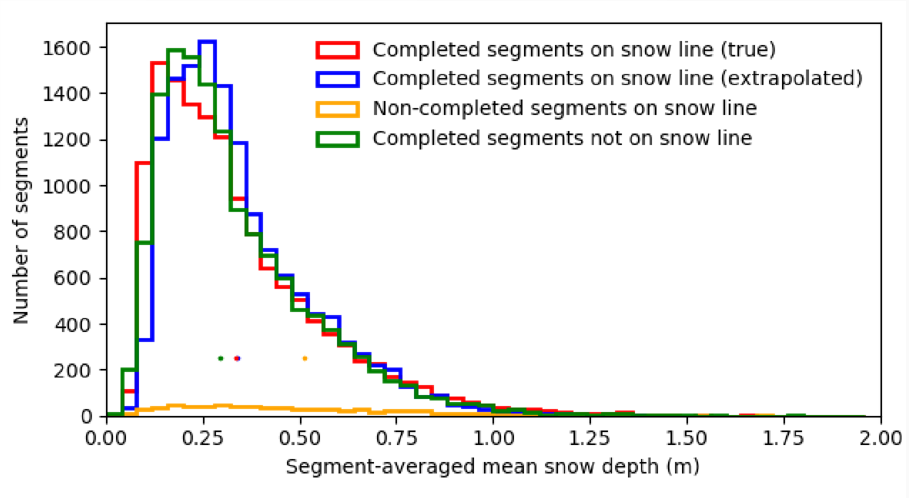


Fig 7. Distributions in snow depth for different groups of segments.

Learning D as a function of F

Now we know the 'true' snow depth (extrapolated to the full lidar window), we can use that as the output (to be predicted) from the input (lidar windows).

Following Mei et al (2019), we turn to convolutional neural networks, as they best use spatial information.

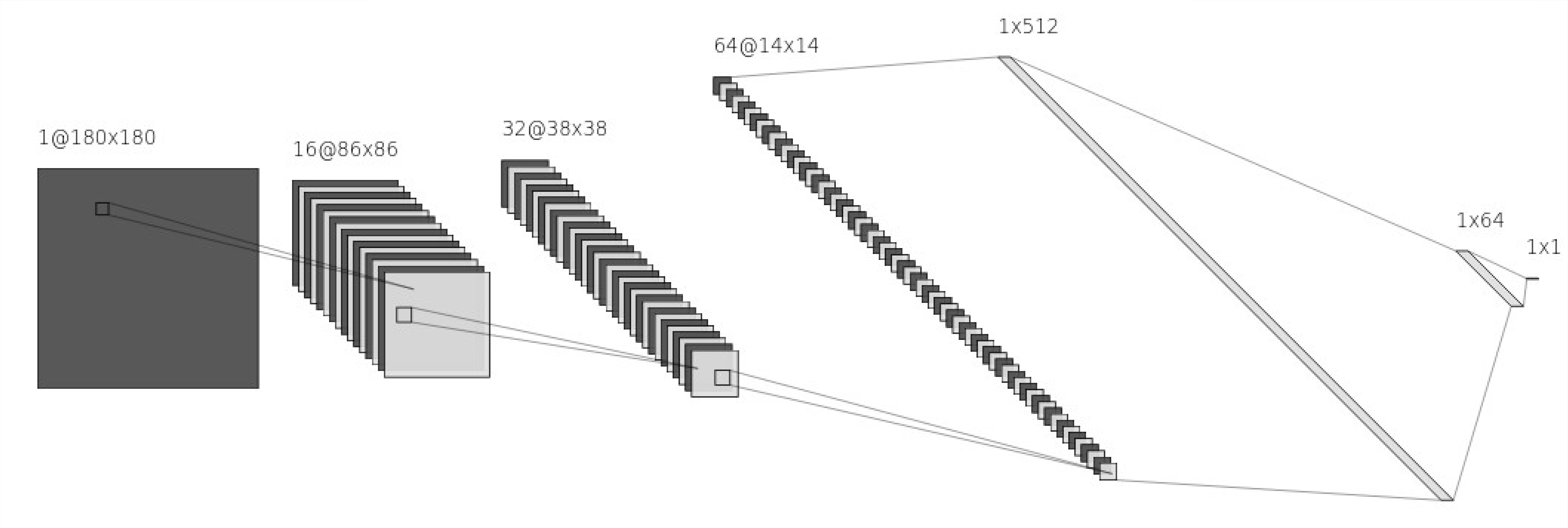


Fig. 8. ConvNet Architecture (LeNail 2019)

Results

Training + Validation data were taken from the 2010/10/28 flight, randomly split 80%-20%. Test data were taken from 2016/10/17 flight (so an independent sample). The linear fit (fitting D to F) is also fit over the training set.

Linear fit has similar fit and test errors but only because the fit is not particularly great. The ConvNet predicts the test set distribution almost perfectly, despite the test distribution being somewhat different to the training one. The linear fit gets the distribution of the test set markedly wrong.

The linear fit (by definition) would predict the training mean snow depth correctly, but clearly overestimates the test snow depth mean.

ConvNet results: predicting mean snow depth with **relative error 20%, RMSE 4.5 cm.**

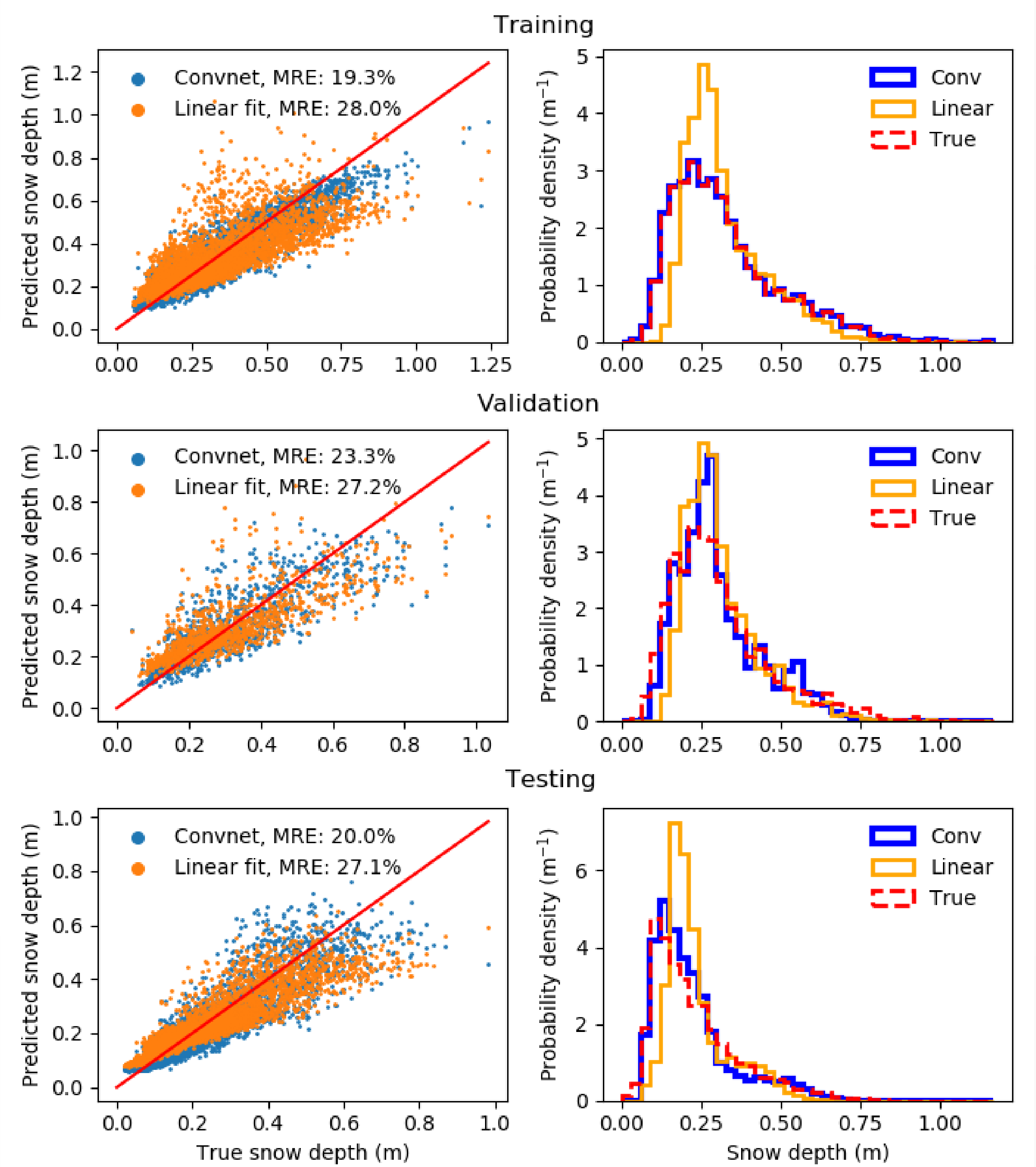


Fig. 9. Scatter plots of the predicted vs true snow depths (left) and their overall distributions (right) for the training/validation/test sets.

Discussion – learned weights

Layer 1: gradients, i.e. edge detectors

Layer 2: steerable pyramid kernels (compare to Fig. 11)

Layer 3: textural components?

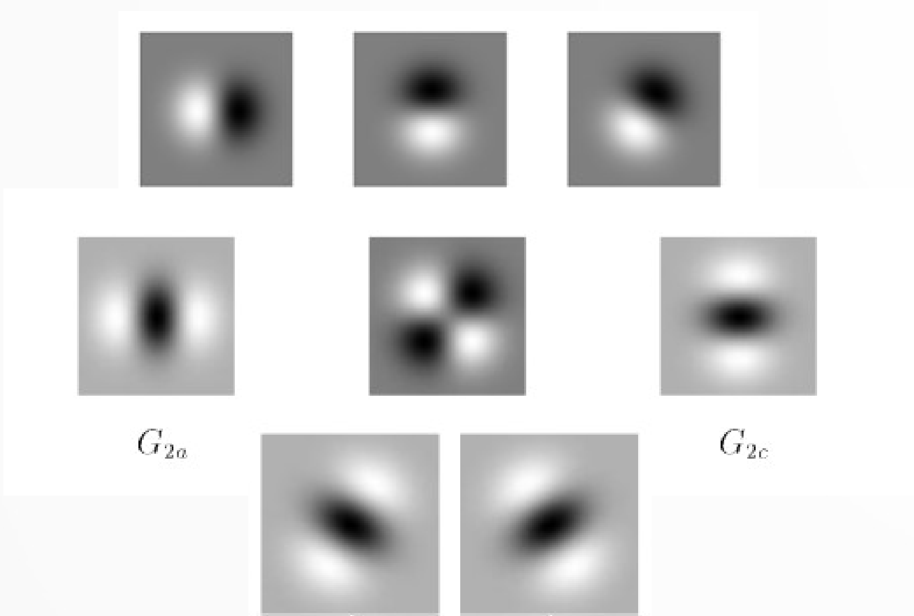


Fig 11. Examples of steerable pyramid kernels corresponding to first and second Gaussian derivatives (Freeman 1991)

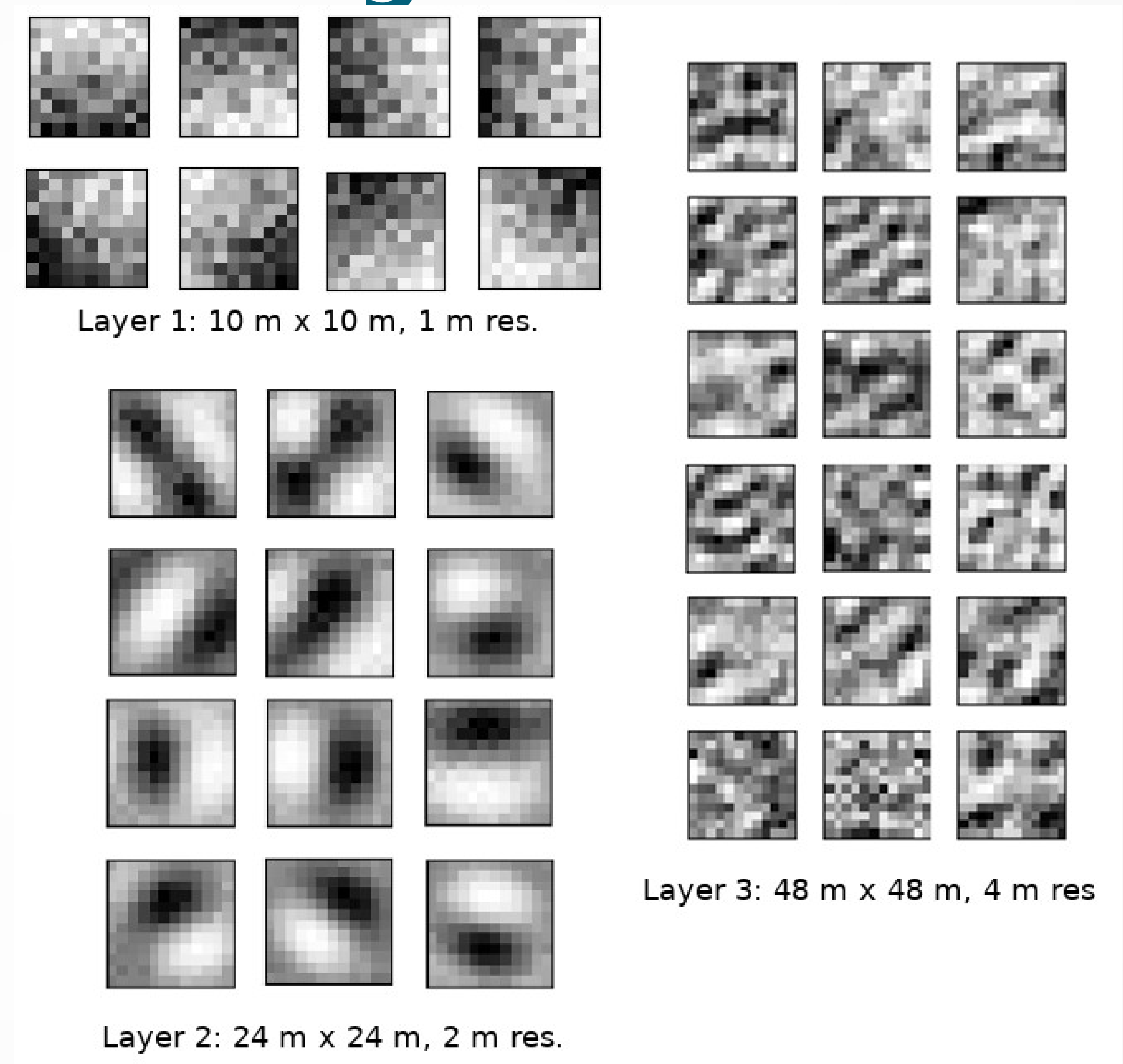


Fig 10. Learned weights for select filters for the first three convolutional layers in the ConvNet.

Discussion – larger scale errors

So we saw earlier that the mean snow depth at 180 m scale had a typical mean error of 20%. What about larger-scale errors?

Vertical lines show the mean relative error. At 1.5 km, mean relative error is 14.0%.

As can be seen in Fig. 12, the mean error reduces as we average over longer segments, and the linear fit error approaches the ConvNet error.

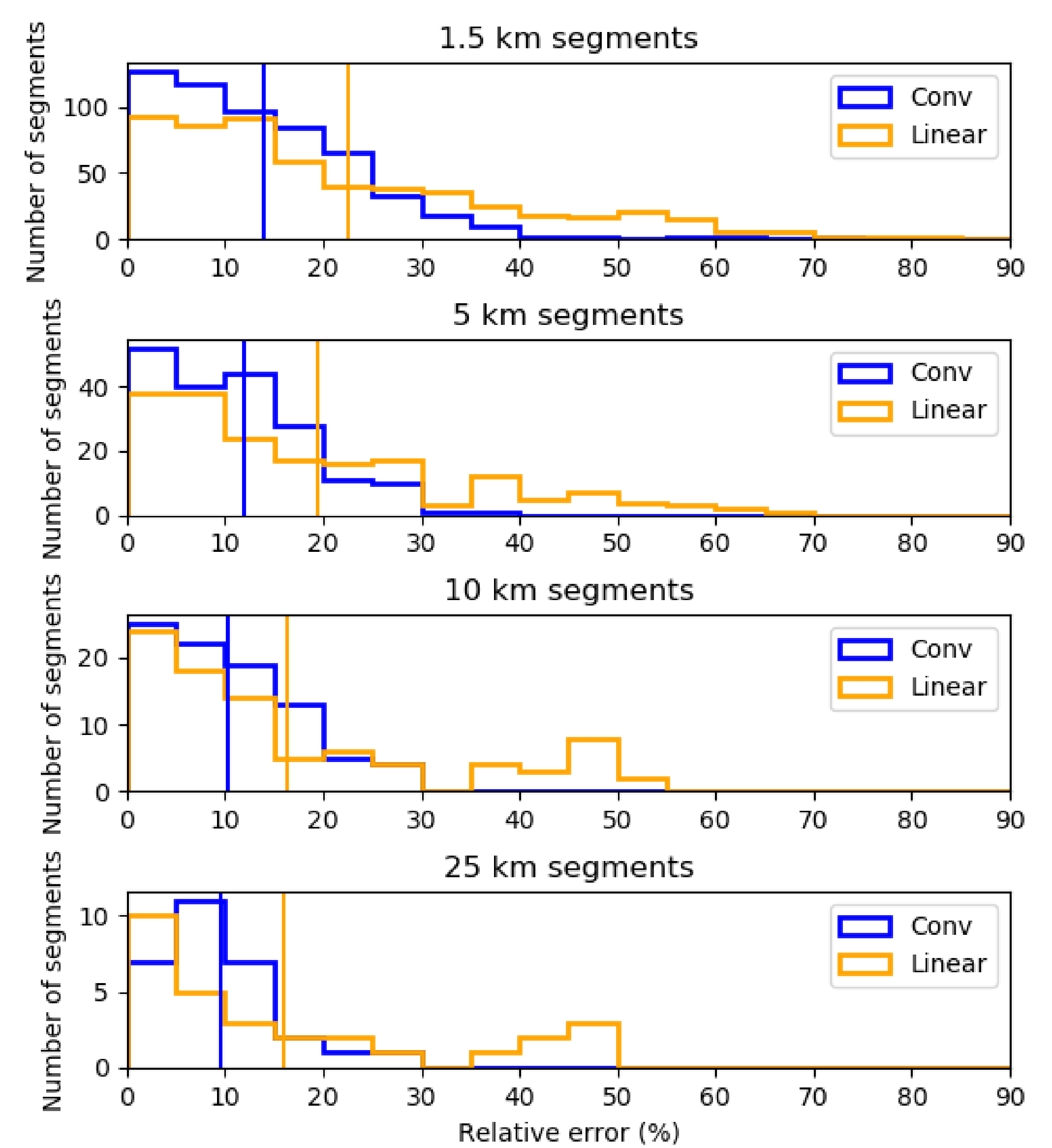


Fig. 12. Overall errors for segment-averaged snow depth, for 1.5-25 km segments, applied to the test set.

Everything together

1. F > D for measured points, but not when averaged → mean of raw snow points is biased

2. Linear underpredicts thick snow and overpredicts thin snow (i.e. regresses to mean)

3. D is not a function of F. Same F, different D (both extrapolated and measured) (see arrows)

4. Extrapolation agrees with raw mean when points have low vertical variability (x = 1.8), less so near ridges (x = 3.1 km) (see arrows)

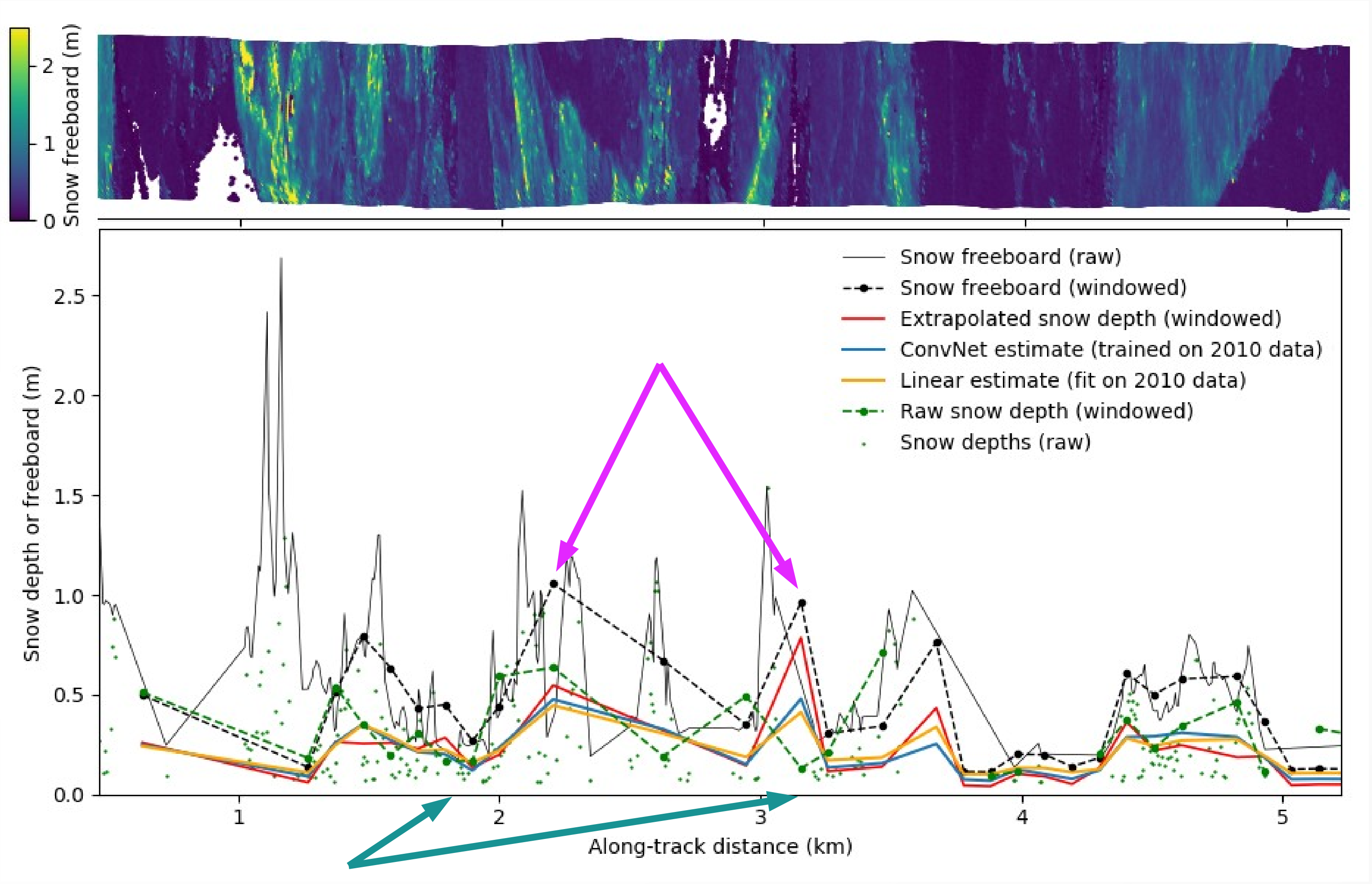


Fig. 13. An example flight segment, showing the lead-referenced snow freeboard (top) and the snow depth and snow freeboard measurements and estimates. ConvNet average error: 18%; linear fit: 29%.

Discussion: Implications for SIT

$$\begin{aligned}\epsilon_T^2 =& \epsilon_F^2\left(\frac{\rho_w}{\rho_w-\rho_i}\right)^2+\epsilon_D^2\left(\frac{\rho_s-\rho_w}{\rho_w-\rho_i}\right)^2+\epsilon_{\rho_s}^2\left(\frac{D}{\rho_w-\rho_i}\right)^2+\\ &+\epsilon_{\rho_w}^2\left(\frac{F-D}{\rho_w-\rho_i}-\frac{F\rho_w+D(\rho_s-\rho_w)}{(\rho_w-\rho_i)^2}\right)^2\\ &+\epsilon_{\rho_i}^2\left(\frac{F\rho_w}{(\rho_w-\rho_i)^2}+\frac{D(\rho_s-\rho_w)}{(\rho_w-\rho_i)^2}\right)^2\end{aligned}$$

First order error for thickness (ϵ_T) as a function of snow, ice, seawater densities, and F and D, from Giles (2007).

Using uncertainties from literature for $\epsilon(\rho_i)$=20 kgm^{-3}, $\epsilon(\rho_s)$=50 kgm^{-3}, $\epsilon(\rho_w)$=1 kgm^{-3}, we work out ϵ_D = 14% x 0.22 m = 0.033 m, ϵ_F = 0.016 m.

Using the mean F and D values from our 1.5 km segments of 0.44 m and 0.22 m respectively, typical densities of $\rho_w = 1024$ kg m^{-3}, $\rho_i = 915$kg m^{-3}, ρ_s=300 kg m^{-3}, this gives:

$$\begin{aligned}\epsilon_T^2 =& 88.3\times 0.016^2+44.1\times 0.033^2+4.07\times 10^{-6}\times 50^2\\ &+2.62\times 10^{-3}\times 1^2+6.01\times 10^{-4}\times 20^2\\ =& 0.023+0.048+0.010+0.003+0.240\end{aligned}$$

This gives a mean SIT of **2.79 ± 0.57 m** (20% error). The error is dominated by the uncertainty in ρ_i (49 cm); the uncertainty from ρ_s, D and F are comparable, and the uncertainty from ρ_w is negligible.

Conclusions

- Snow depth on sea ice can be predicted, at scales of 180 m – 1.5 km, with errors of 14-20%.

- Thisgives errors in sea ice thickness of ~20%.

- The learned filters support the idea that snow surface texture is related to snow depth (and hence sea ice thickness). These features may be generalized between different Weddell sea datasets from different years.

Future work

- More years, more regions – any textural classes? What is the variability temporally and regionally? Are the textural features generalizable?

- Can we resolve trends beyond interannual variability?

- Can any of these techniques be applied to ICESat-2 data? (13m footprint, heavily oversampled (0.7m along-track)

References

- Farrell, S.L., Kurtz, N., Connor, L.N., Elder, B.C., Leuschen, C., Markus, T., McAdoo, D.C., Panzer, B., Richter-Menge, J. and Sonntag, J.G., 2011. A first assessment of IceBridge snow and ice thickness data over Arctic sea ice. IEEE Transactions on Geoscience and Remote Sensing, 50(6), pp.2098-2111.
- Freeman, W.T. and Adelson, E.H., 1991. The design and use of steerable filters. IEEE Transactions on Pattern Analysis & Machine Intelligence, (9), pp.891-906.
- Kern, S. and Spreen, G., 2015. Uncertainties in Antarctic sea-ice thickness retrieval from ICESat. Annals of Glaciology, 56(69), pp.107-119.
- Kwok, R. and Maksym, T., 2014. Snow depth of the W eddell and B ellingshausen sea ice covers from I ce B ridge surveys in 2010 and 2011: An examination. Journal of Geophysical Research: Oceans, 119(7), pp.4141-4167.
- Mei, M.J., Maksym, T., Weissling, B. and Singh, H., 2019. Estimating early-winter Antarctic sea ice thickness from deformed ice morphology. Cryosphere, 13(11).