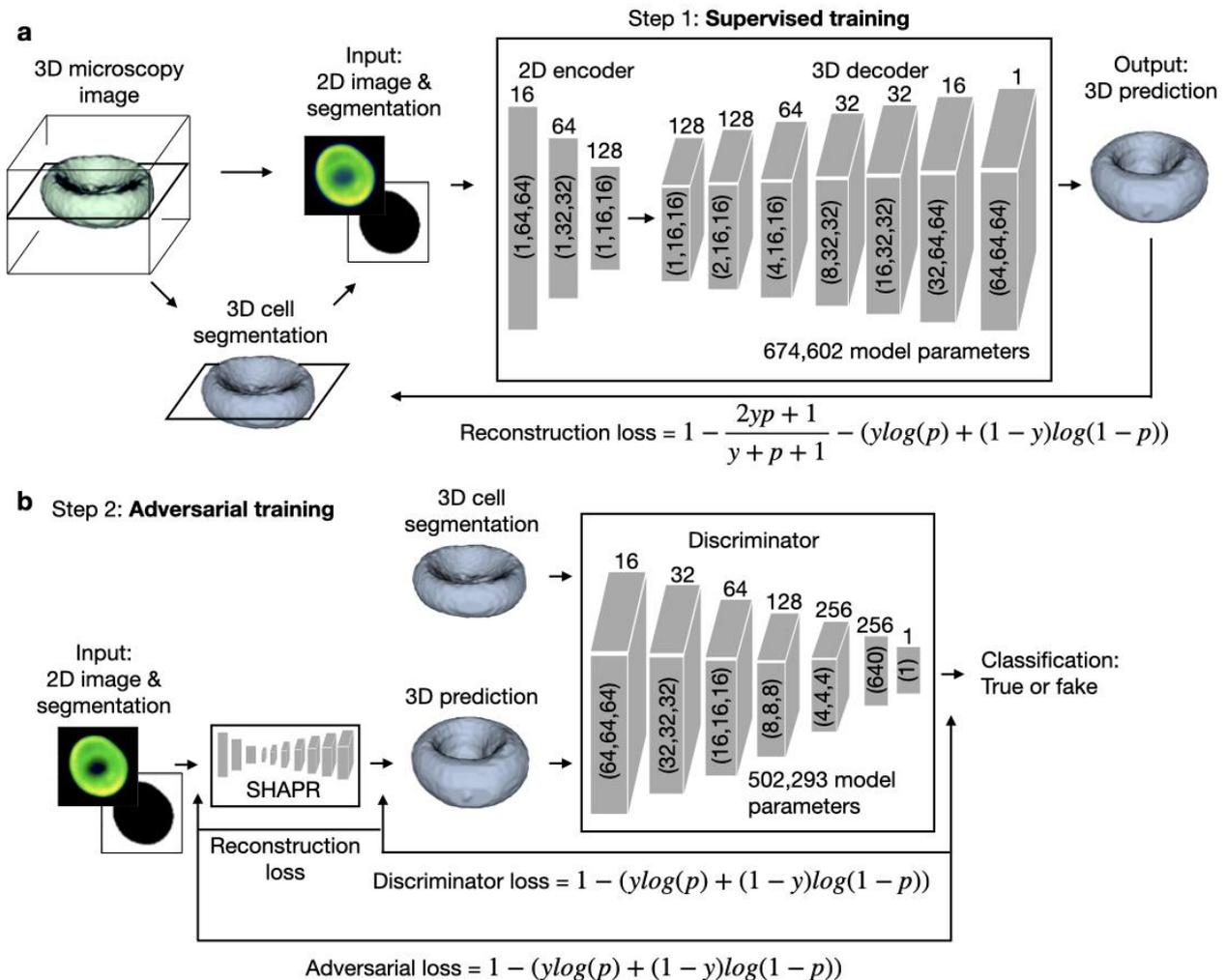**Supplemental information**

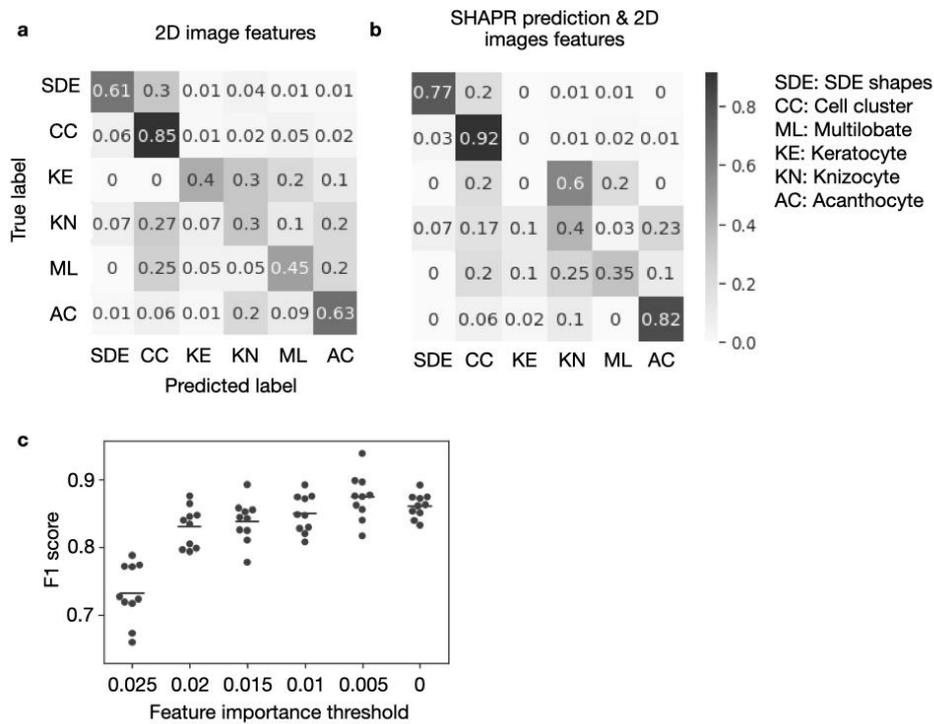# SHAPR predicts 3D cell shapes

from 2D microscopic images

Dominik J.E. Waibel, Niklas Kiermeyer, Scott Atwell, Ario Sadafi, Matthias Meier, and Carsten Marr

SHAPR Supplementary material

Supplementary Figure 1 | SHAPR utilizes a two step training process. a, SHAPR consists of a 2D encoder, which embeds 2D images into a 128-dimensional latent space, and a 3D decoder, which reconstructs 3D shapes from the latent space representation. To train SHAPR we segment 3D microscopy images (we show an exemplary single red blood cell). We pair a 2D segmentation with the microscopy image of the same slice to enter the encoder as input. During supervised training (Fig. 1, step 1), we minimize the reconstruction loss (see Methods), which is the sum of the Dice loss and the binary cross entropy loss between the 3D segmentations y and SHAPR predictions p. For an input image of 64 x 64 pixels, we provide the pixel sizes for each layer in the gray boxes and the filter sizes on top of each box. **b**, In the second step, we fine-tune SHAPR by adding a discriminator. The discriminator is trained to differentiate between SHAPR output p ground truth segmentation r and minimize the adversarial loss. It thereby challenges SHAPR to output realistic 3D objects (related to Fig. 1b).

Supplementary Figure 2 | **Red blood cell classification is significantly improved when morphological features extracted from SHAPR predicted cell shapes are added to features derived from 2D images. a,b,** Confusion matrix for red blood cell classification using 2D images features (a) and 2D images features combined with features from SHAPR 3D prediction (b). SHAPR increases the mean true positive rate in a 5-fold cross-validation for 3 out of 5 classes. For keratocytes and knizocytes the mean true positive rate is the same. **c,** Our random forest trained on the combination of 3D SHAPR features, 2D image features, and 2D segmentation features yields the highest F1 score if we remove features that are less important than a threshold of 0.005 before retraining the random forest (related to Fig. 1e, results).

Mean±std.dev. [%]	3D red blood cell fit (n = 825)		
	SHAPR	Cylindrical model	Ellipsoid model
Intersection over union	**63±12**	30±28	54±13
Relative Volume error	**20±18**	33±22	37±23
Relative Surface area error	**15±11**	22±15	40±18
Relative surface roughness error	**11±7**	38±9	28±13

Supplementary Table 1 | SHAPR outperforms naïve stereological models on a red blood cell dataset. Using SHAPR, we obtain a higher intersection over union, a lower volume, surface, and roughness estimation predicting the 3D shape of red blood cells compared to predicting the quantities using a cylindrical or ellipsoid fit (Fig. 2b). Best results are highlighted in boldface (related to Fig. 1d).

Mean±std.dev. [%]	3D nuclei cell fit (n = 887)		
	SHAPR	Cylindrical model	Ellipsoid model
Intersection over union	**46±16**	41±19	31±16
Relative Volume error	**33±41**	44±25	62±19
Relative Surface area error	**32±36**	36±22	54±16

Supplementary Table 2 | SHAPR outperforms naïve stereological models on a nuclei dataset. Using SHAPR, we obtain a higher intersection over union, a lower volume, surface, and roughness estimation predicting the 3D shape of nuclei as compared to predicting the quantities using a cylindrical or ellipsoid fit (Fig. 3f). Best results are highlighted in boldface (related to Fig. 2d).

3D features (126)	2D segmentation features (9)	2D image features (5)
Volume	Mean pixel value	Mean pixel value
Surface	Surface	Standard deviation of pixel value
Mean	Boundary of outline	Gabor feature
Shape index	Roughness	Gray-level co-occurrence dissimilarity and correlation (2 features)
Roughness	Convexity	
Gaussian roughness	Gray-level co-occurrence dissimilarity and correlation (2 features)	
Convexity	Moment center row and center column (2 features)	
3D boundary		
Gabor feature of z-projection		
Gray-level co-occurrence dissimilarity and correlation (2 Features)		
Mean, standard deviation, and maximum of z-projection (3 features)		
Convexity of central slices (3 features)		
Boundary of central slices (3 features)		
Inertia eigenvalues (3 features)		
Mean, median, and Standard deviation of triangular faces (calculated with Lewiner marching cubes algorithm) (3 features)		
Mean, median, standard deviation of vertices (calculated with Lewiner marching cubes algorithm) (3 features)		
Mesh volume, mesh convex hull volume (2 features)		
Number of vertices and faces (2 features)		
Moments (8 features)		
Mesh inertia eigenvalues (9 features)		
Mesh principals (12 features)		
Gabor filter of each z-slice (64 features)		

Supplementary Table 3 | List of features extracted for different image modalities. For 3D images, a total of 126 features were acquired by using built-in functions of the python Skimage package. Since there are fewer features in 2D than in 3D, only 7 features were extracted from the binary mask and 5 features from the image (related to Fig. 1e).