CycleSTTN: A Learning-Based Temporal Model for Specular Augmentation in Endoscopy

Rema Daher[1(✉)], O. León Barbed[2], Ana C. Murillo[2], Francisco Vasconcelos[1], and Danail Stoyanov[1]

[1] University College London, London, UK
{rema.daher.20,danail.stoyanov}@ucl.ac.uk
[2] Universidad de Zaragoza, Zaragoza, Spain
acm@unizar.es

Abstract. Feature detection and matching is a computer vision problem that underpins different computer assisted techniques in endoscopy, including anatomy and lesion recognition, camera motion estimation, and 3D reconstruction. This problem is made extremely challenging due to the abundant presence of specular reflections. Most of the solutions proposed in the literature are based on filtering or masking out these regions as an additional processing step. There has been little investigation into explicitly learning robustness to such artefacts with single-step end-to-end training. In this paper, we propose an augmentation technique (CycleSTTN) that adds temporally consistent and realistic specularities to endoscopic videos. Such videos can act as ground truth data with known texture occluded behind the added specularities. We demonstrate that our image generation technique produces better results than a standard CycleGAN model. Additionally, we leverage this data augmentation to re-train a deep-learning based feature extractor (SuperPoint) and show that it improves. CycleSTTN code is made available here.

Keywords: Surgical Data Science · Surgical AI · Generative AI · Deep Learning · Endoscopy · Specularity

1 Introduction

During endoscopic procedures, such as colonoscopy, the camera light source produces abundant specular highlight reflections on the visualised anatomy. This is due to its very close proximity to the scene coupled with the presence of wet tissue. These reflections can occlude texture and produce salient artifacts, which may reduce the accuracy of surgical vision algorithms aiming at scene understanding, including depth estimation and 3D reconstruction [5,16]. To resolve

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43999-5_54.

H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14229, pp. 570–580, 2023.
https://doi.org/10.1007/978-3-031-43999-5_54

these challenges, a simple solution is to detect and mask out these regions before performing any other downstream visual task. However, this approach comes with limitations. In the context of sparse feature point detection, simply discarding points falling on specularities results in excessive filtering, often leading to an insufficient number of detected features. As for dense estimation problems, such as pixel depth regression and optical flow, masking out specular regions makes interpolation necessary.

Alternatively, videos can be pre-processed to inpaint specular highlights with its hidden texture inferred from neighbouring frames [7]. This allows running other algorithms with reflection-free data. However, this adds a significant computational overhead and requires the processing of temporal frame windows that restrict online inference applications. Given that state-of-the-art feature detection methods are deep learning models, an appealing approach would be to learn robustness to reflections during training, as this would result in an single-step end-to-end inference model without any pre or post-processing overhead.

In this paper, we propose to learn robustness to specular reflections via data augmentation. We use a CycleGAN [24] methodology that takes advantage of a pre-trained specular highlight removal network in adversarial training. Our proposed generator network, based on STTN [7,23], performs video-to-video translation. In doing so, we create a cycle structure for STTN, that we call CycleSTTN. To demonstrate the effectiveness of our approach, we use it to improve the performance of the SuperPoint feature detector/descriptor. We combine the proposed method with that of [7] to add and remove specularities as data augmentation. The contributions of this paper can be summarised as:

- We propose the CycleSTTN training pipeline as an extension of STTN to a cyclic structure.
- We use CycleSTTN to train a model for synthetic generation of temporally consistent and realistic specularities in endoscopy videos. We compare results of our method against CycleGAN.
- We demonstrate CycleSTTN as a data augmentation technique that improves the performance of SuperPoint feature detector in endoscopy videos.

2 Related Work

Barbed et al. investigated the challenge of specular reflections when performing feature detection in endoscopy images [3]. As a solution, they design a loss function to explicitly avoid specular regions and focus on detecting points in the remaining parts of the image. However, this ignores the fact that features extracted from these points will still be contaminated by specularity pixels in their neighborhood. Following a different strategy, we propose data augmentation as a way to induce specularity robustness in both point detection and feature extraction.

Data augmentation for lighting conditions in surgical data has been explored extensively before. Classical techniques for modelling specular highlights include the use of a parametric model to add specularities and illumination to real

images [1,13]. Another technique for augmentation uses synthetic data or phantoms, where different light sources and environment variables can be captured. However, synthetic data lacks real textures and artifacts. Thus many image-to-image translation techniques have been developed to add more realistic textures to synthetic data [15,17,21,22]. These methods rely on CycleGAN [24] for unsupervised learning and thus, are able to map from real to synthetic domain and vice-versa. This allows for the generation of real images with the same structure but different lighting, texture, and blurriness. To have more control over the augmentations, some methods add a controllable noise vector to the network input that modifies the image lighting, specular reflections, and texture. However, this vector does not have a physical meaning, and it is thus challenging to independently control the different environment variables by directly manipulating its values. To address this, [14] uses two separate noise inputs, one controlling texture and specular reflection and another controlling color and light. However, all these approaches still use multiple steps to finally create new real data, which might lead to loss of important information in the process.

Single-step approaches have also been developed that augment real data directly, but the generated images have different structures and thus, do not create paired data [9,20]. A CycleGAN model has been proposed to map from images with specularities to images without specularities [11] using manually labelled patches cropped from frames, however, this work focuses on specular highlight removal, and does not test the data augmentation capabilities of generating synthetic specularities. In [12] a classification model is used to categorize data for unpaired training of CycleGAN. From the output of CycleGAN, they generate a paired dataset, however, only quality metrics are used to filter out images in the generated paired dataset. Furthermore, most of these approaches are applicable to single frames and are not able to generate synthetic videos with temporally consistent specularities. While some other methods use a temporal component for endoscopic video augmentation [17,21], they do not have a single-step structure and have not been applied to generate/remove specular highlights.

3 Methods

We use the STTN model as our video-to-video translation architecture and T-PatchGAN [6] as the discriminator. STTN contains an encoder followed by a spatio-temporal transformer and a decoder [7,23]. While STTN was originally proposed to remove occlusions in videos, in this paper, we use two instances of this model, $STTN_R$ and $STTN_A$, to respectively remove and add specular occlusions. We start by pre-training these models separately using their respective discriminators D_R and D_A. Then, we continue training them simultaneously in an adversarial manner with a CycleGAN methodology. We denote the complete training pipeline as CycleSTTN (Fig. 1), which is divided into 3 sections:

1. ***Paired Dataset Generation*** We generate a dataset of paired videos with and without specularities. We train a generator for specularity removal fol-

lowing the same methodology as in [7]. This model is denoted as ($STTN_{R0}$, D_{R0}), where $STTN$ is the generator and D is the discriminator. For a set of real endoscopic videos with specularities V_A and specularity masks M, we run $STTN_{R0}$ to generate their inpainted counterparts without specularities V_R (Fig. 1 - Step 1).

2. ***$STTN_A$ Pre-training*** Using the paired dataset (V_A, V_R) we train a new model to add specularities. We denote it as $STTN_{A0}$ with D_{A0} as its discriminator. This is shown in Fig. 1 as step 2.
3. ***$STTN_R, STTN_A$ Joint Training*** By initializing with the models from Step 1 and 2 (($STTN_{R0}$, D_{R0}) and ($STTN_{A0}$, D_{A0})), we continue training $STTN_R$ and $STTN_A$ simultaneously in an adversarial manner with a CycleGAN methodology. We denote the final removal and addition models as ($STTN_{R1}$, D_{R1}) and ($STTN_{A1}$, D_{A1}), respectively. This is also shown in Fig. 1 as step 3.

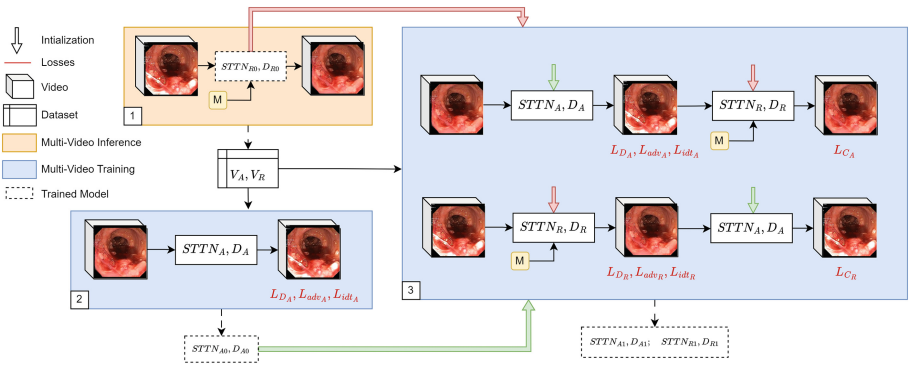

Fig. 1. CycleSTTN training pipeline with 3 main steps: 1 Paired Dataset Generation, 2 $STTN_A$ Pre-training, and 3 $STTN_R, STTN_A$ Joint Training.

3.1 Model Inputs

The STTN architecture receives as input a paired sequence of frames and masks. Originally masks are meant to represent occluded image regions that should be inpainted, however, in this work, we do not always use them in this way. When training $STTN_R$ we define the mask inputs as regions to be inpainted (to remove specularities). However, when training $STTN_A$, input masks are set to 1 for all pixels, since we do not want to enforce specific locations for specularity generation; we want to encourage the model to learn these patterns from data.

3.2 Losses

The loss function of the T-PatchGAN discriminators are shown below, such that $\mathbb{E}$ is the expected value of the data distributions as done in [23]:

$$L_{D_R} = \mathbb{E}_{V_R \sim p(V_R)}[ReLU(1 - D_R(V_R))] + \mathbb{E}_{Fake'_R \sim p(Fake'_R)}[ReLU(1 + D_R(Fake'_R))] \quad (1)$$

$$L_{D_A} = \mathbb{E}_{V_A \sim p(V_A)}[ReLU(1 - D_A(V_A))] + \mathbb{E}_{Fake_A \sim p(Fake_A)}[ReLU(1 + D_A(Fake_A))] \quad (2)$$

where $Fake_A = STTN_A(V_R)$ represents fake videos with added specularities, and analogously $Fake_R = STTN_R(V_A)$ represents fake videos with removed specularities. Further, we also define $Fake'_R = M.Fake_R + V_A(1 - M)$ for the discriminator loss, where inpainted occluded regions from $Fake_R$ are overlaid over V_A. M denotes masks with 1 values in specular regions of V_A and 0 otherwise.

For the generators, an adversarial loss was used as done in [7]:

$$L_{adv_R} = -\mathbb{E}_{Fake'_R \sim p(Fake'_R)}[D_R(Fake'_R)]; \quad L_{adv_A} = -\mathbb{E}_{Fake_A \sim p(Fake_A)}[D_A(Fake_A)] \quad (3)$$

The identity loss was the only loss modified from the original STTN model [7]. The identity loss for $STTN_R$ and $STTN_A$ are:

$$L_{idt_R} = \|V_R - STTN_R(V_R)\|_1; \quad L_{idt_A} = \|V_A - Fake_A\|_1 \quad (4)$$

Here L_{idt_R} ensures that if a video does not have specularities, it would stay the same when fed into the model that removes specularities. Whereas, L_{idt_A} ensures predicted videos with specularities, $Fake_A$, resemble real specular videos V_A.

Finally, we added cycle loss terms:

$$L_{c_R} = \mathbb{E}_{V_A \sim p(V_A)}[\|V_A - STTN_A(Fake_R)\|_1]; \quad L_{c_A} = \mathbb{E}_{V_R \sim p(V_R)}[\|V_R - STTN_R(Fake_A)\|_1] \quad (5)$$

The total generator losses L_R and L_A for removing and adding specularities are shown below, such that the loss weights λ are all set to 1 except for λ_{adv}, which is set to 0.01 as advised by [23]:

$$L_R = \lambda_{adv} L_{adv_R} + \lambda_{idt} L_{idt_R} + \lambda_c L_{c_R}; \quad L_A = \lambda_{adv} L_{adv_A} + \lambda_{idt} L_{idt_A} + \lambda_c L_{c_A} \quad (6)$$

In summary, we adopted the original STTN model [7] and changed the training pipeline, model inputs, and losses as shown in Fig. 1. In particular, (a) the training pipeline was transformed into a multi-task one of adding and removing specularities, where $STTN_{R0}$ from [7] was used as an initialization model. (b) For $STTN_A$, specularity masks were removed from model inputs. (c) Identity losses and cycle losses were also added while masked based losses were removed.

4 Experiments and Results

4.1 Datasets and Parameters

To evaluate our pipeline, we use 373 videos from the Hyper Kvasir dataset [4] to generate our paired dataset (V_R, V_A) as described in Sect. 3 and Fig. 1. 343 video

pairs were used for training and 30 for testing with an upper limit of 927 frames per video. Models were trained on NVIDIA A100-SXM4-40GB GPUs. We use the same training parameters as [7,23] with the exception of batch size, which we changed from 8 to 3 for training ($STTN_{A1}$, $STTN_{R1}$). CycleGAN models were trained with suggested parameters in CycleGAN's public repository[1], with the exception of batch size, which was changed from 1 to 3.

In our experimental analysis, we use our proposed models shown in Fig. 1 along with CycleGAN models, which use ResNet with 9 residual blocks as the generator. All these models are listed in Table 1. We note that even though $STTN_{R1}$ was trained with masks, it seems it was affected by the cycle loss and was only able to give decent results without a mask as input.

Table 1. Trained models used in our analysis input type and training iterations.

Proposed Models	Input	Iterations	CycleGAN Models	Identity Loss	Input	Iterations
$STTN_{R0}$	videos + masks	90,000	($ResNet_{A0}$, $ResNet_{R0}$)	Original [24]	videos	285,600
$STTN_{A0}$	videos	30,000	($ResNet_{A1}$, $ResNet_{R1}$)	L_{idt_A} - Eq. 4	videos	285,600
($STTN_{A1}$, $STTN_{R1}$)	videos	20,000				

4.2 Pseudo Evaluation Experiments

We input the pseudo ground truth V_R to our models, $STTN_{A0}$ and $STTN_{A1}$. We compare the output $Fake_A$ to real videos V_A. We conduct non-temporal testing by using single frame inputs, as opposed to video inputs, to demonstrate the temporal effect. We report the Peak Signal to Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Mean Square Error (MSE) metrics.

We show visual results for different models in Fig. 2. When V_R (videos with no specularities) are used as input, our CycleSTTN model $STTN_{A1}$ shows the highest similarity to the ground truth (V_A). With V_A (videos with specularities) as input, $STTN_{A1}$ is able to add more realistic specularities that flow smoothly from one frame to another. CycleGAN based models were not able to add new specularities with V_A as input. For $ResNet_{A0}$, this is expected, due to the original CycleGAN identity loss. When changing the identity loss, $ResNet_{A1}$ only intensifies specularities and darkens the background texture. This was further validated through results shown in Table 2, where $STTN_{A1}$ has the best SSIM, PSNR and MSE values. We can also see that $STTN_{A1}$ results are only slightly

[1] https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix/tree/pytorch0.3.1.

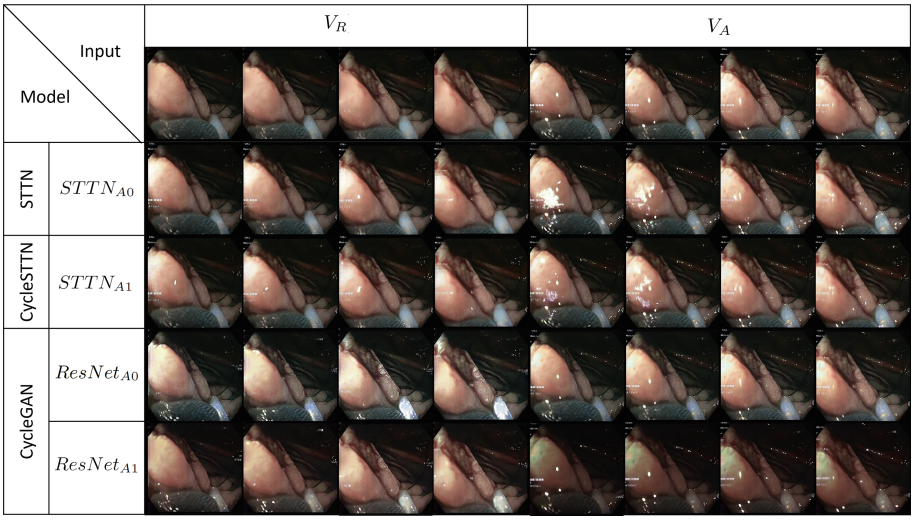

Fig. 2. Sample consecutive video frames from the model output using pseudo ground truth V_R as input in columns 1–4 and real videos V_A as input in columns 5–8.

Table 2. Mean SSIM, PSNR, and MSE values for model output videos using pseudo ground truth V_R as input. The output is compared to real videos V_A.

Method	Non-Temporal Testing						Temporal Testing					
	SSIM		PSNR		MSE		SSIM		PSNR		MSE	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
$STTN_{A0}$	0.802	0.040	26.20	2.31	231	98.0	0.808	0.039	26.33	2.37	226	91.9
$STTN_{A1}$	**0.826**	0.036	26.38	2.43	224	102.5	0.824	0.036	**26.49**	2.45	**219**	91.4
$ResNet_{A0}$	0.792	0.041	21.29	1.92	675	311	-	-	-	-	-	-
$ResNet_{A1}$	0.780	0.044	21.48	2.03	666	329	-	-	-	-	-	-

worse without temporal testing according to PSNR and MSE, yet slightly better according to SSIM. Thus, the temporal component only benefits training as a regularizer (i.e. our models outperforms CycleGAN based models). However, during inference, temporal testing is not necessary since it yields similar results to non-temporal testing.

The pseudo evaluation shows that generated specularities closely resemble real ones in appearance and location. While not fully enforcing physical realism, this augmentation improves upon traditional warping methods (Sect. 4.3).

4.3 Relative Pose Estimation Experiments

We use our models as data augmentation to re-train the feature detector proposed in [3], an adaptation of SuperPoint [8] to endoscopy. While [3] is originally trained with a specularity loss term that encourages the network to ignore specularity regions, we omit this term in our training. We want our models to be robust

to specularities, rather than just avoiding them. We use the pre-trained model in [3] to generate data labels and initialize our models. SuperPoint is trained by generating a sparse set of point matches in an image and its warped version via homography. Augmentations already used by SuperPoint include traditional random brightness, contrast, Gaussian noise, speckle noise, blur, and shade. We use our models as augmentations to original and warped images separately by randomly choosing between (no augmentation, specularity addition, and specularity removal). SuperPoint models with various augmentations are listed in Table 3. These models are trained using 131 randomly chosen videos from V_A (with cropped boundaries) and 14 for validation for 25,000 iterations with the same training parameters as [3]. Temporal (non-temporal) refers to feeding an image and its warped version together (separately) to the augmentation model.

Table 3. Pose estimation analysis for 12 SuperPoint models each trained with different specular augmentations. Metrics are described in Sect. 4.3.

	Specularity Augmentation Models		Non-Temporal				Temporal			
					Rot. error				Rot. error	
	Addition	Removal	Matches	Precision	Mean	Median	Matches	Precision	Mean	Median
1	-	-	368.6	25.0	24.8	12.0	-	-	-	-
2	-	$STTN_{R0}$	537.4	29.2	21.4	10.1	538.2	29.2	21.2	9.8
3	$STTN_{A0}$	$STTN_{R0}$	402.1	27.8	21.7	10.7	404.2	27.5	22.0	10.6
4	$STTN_{A0}$	-	398.3	26.7	23.7	11.4	390.1	26.5	22.8	11.1
5	-	$STTN_{R1}$	373.3	26.0	22.6	10.3	542.8	28.6	23.4	11.1
6	$STTN_{A1}$	$STTN_{R1}$	360.3	27.3	21.3	9.6	548.5	28.7	23.1	11.0
7	$STTN_{A1}$	-	386.3	27.2	21.5	10.1	542.5	28.3	23.5	11.2
8	$STTN_{A1}$	$STTN_{R0}$	541.5	29.9	20.4	9.5	543.6	29.8	20.2	9.6
9	$STTN_{A0}$	$STTN_{R1}$	536.6	29.2	21.4	10.0	526.7	29.0	21.1	9.7
10	-	$ResNet_{R0}$	571.1	27.6	22.0	10.3				
11	$ResNet_{A0}$	$ResNet_{R0}$	531.0	26.8	22.5	10.6				
12	$ResNet_{A0}$	-	529.4	27.3	22.6	10.5				
13	-	$ResNet_{R1}$	571.6	28.2	22.7	10.5				
14	$ResNet_{A1}$	$ResNet_{R1}$	548.4	29.0	22.2	10.3				
15	$ResNet_{A1}$	-	527.3	29.0	22.4	10.5				

We evaluate the quality of point detections by using them to estimate relative camera motion in endoscopic sequences. We first apply brute-force matching of detected points in image pairs and then estimate motion via RANSAC [10]. The test data for this experiment is the same as in [3]. It includes 6 sequences (14191 frames) from the EndoMapper dataset [2] with a relative camera motion pseudo ground truth based on structure-from-motion (SFM: COLMAP [18,19]). Reported metrics include the precision of inlier points from RANSAC-estimated (threshold = 10 px) essential matrices as compared to inlier points using pseudo ground truth essential matrix (from COLMAP). To generate the pseudo ground truth inliers, the same distance metric used in RANSAC was applied to the pseudo ground truth essential matrix (from COLMAP). We also report the Rota-

tion error, which is the geodesic angle in degrees between the RANSAC-based pose estimation and the pseudo ground truth pose (from COLMAP).

In Table 3, all specularity augmentations (models 2–15) improve SuperPoint relative to not using them (model 1), which demonstrates their usefulness. Most STTN-based augmentations (models 2–9) produce better results than CycleGAN-based ones (models 10–15). Overall, the best performing augmentation is ($STTN_{A1}$, $STTN_{R0}$), showing the effectiveness of our system. This makes sense since the best removal and addition models are $STTN_{R0}$ and $STTN_{A1}$ according to the rotation error. However, it appears that non-temporal testing sometimes gives lower rotation errors than temporal testing. This could be due to the unrealistic nature of warped images used as consecutive frames. As also discussed in Sect. 4.2, temporal testing does not improve results, only temporal training does.

5 Conclusion

In conclusion, we introduce CycleSTTN, a temporal CycleGAN applied to generate temporally consistent and realistic specularities in endoscopy. Our model outperforms CycleGAN, as demonstrated by mean PSNR, MSE, and SSIM metrics using a pseudo ground truth dataset. We also observe a positive effect of our model as augmentation for training a feature extractor, resulting in improved inlier precision and rotation errors. However, our evaluation relies on SFM generated ground truth, different testing and training datasets, and indirect metrics, which may introduce some uncertainty. Nevertheless, augmentation shows great promise as an addition for training various endoscopic computer vision tasks to enhance performance and provide insights into the impact of specific artifacts.

Acknowledgments. This research was funded in part, by the Wellcome/EPSRC Centre for Interventional and Surgical Sciences (WEISS) [203145/Z/16/Z]; the Engineering and Physical Sciences Research Council (EPSRC) [EP/P027938/1, EP/R004080/1, EP/P012841/1]; the Royal Academy of Engineering Chair in Emerging Technologies Scheme; H2020 FET (GA863146); and the UCL Centre for Digital Innovation through the Amazon Web Services (AWS) Doctoral Scholarship in Digital Innovation 2022/2023. For the purpose of open access, the author has applied a CC BY public copyright licence to any author accepted manuscript version arising from this submission.

References

1. Asif, M., Chen, L., Song, H., Yang, J., Frangi, A.F.: An automatic framework for endoscopic image restoration and enhancement. Appl. Intell. **51**(4), 1959–1971 (2021)
2. Azagra, P., et al.: Endomapper dataset of complete calibrated endoscopy procedures. arXiv preprint arXiv:2204.14240 (2022)

3. Barbed, O.L., Chadebecq, F., Morlana, J., Montiel, J.M.M., Murillo, A.C.: Superpoint features in endoscopy. In: Manfredi, L., et al. (eds.) ISGIE GRAIL 2022. LNCS, vol. 13754, pp. 45–55. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-21083-9_5
4. Borgli, H., et al.: Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. Sci. Data **7**(1), 1–14 (2020)
5. Chadebecq, F., Lovat, L.B., Stoyanov, D.: Artificial intelligence and automation in endoscopy and surgery. Nat. Rev. Gastroenterol. Hepatol. **20**(3), 171–182 (2023)
6. Chang, Y.L., Liu, Z.Y., Lee, K.Y., Hsu, W.: Free-form video inpainting with 3D gated convolution and temporal PatchGAN. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9066–9075 (2019)
7. Daher, R., Vasconcelos, F., Stoyanov, D.: A temporal learning approach to inpainting endoscopic specularities and its effect on image correspondence. arXiv preprint arXiv:2203.17013 (2022)
8. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: self-supervised interest point detection and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 224–236 (2018)
9. Diamantis, D.E., Gatoula, P., Iakovidis, D.K.: Endovae: generating endoscopic images with a variational autoencoder. In: 2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), pp. 1–5. IEEE (2022)
10. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
11. Funke, I., Bodenstedt, S., Riediger, C., Weitz, J., Speidel, S.: Generative adversarial networks for specular highlight removal in endoscopic images. In: Medical Imaging 2018: Image-Guided Procedures, Robotic Interventions, and Modeling, vol. 10576, pp. 8–16. SPIE (2018)
12. García-Vega, A., et al.: A novel hybrid endoscopic dataset for evaluating machine learning-based photometric image enhancement models. In: Pichardo Lagunas, O., Martínez-Miranda, J., Martínez Seis, B. (eds.) MICAI 2022. LNCS, vol. 13612, pp. 267–281. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19493-1_22
13. Hegenbart, S., Uhl, A., Vécsei, A.: Impact of endoscopic image degradations on LBP based features using one-class SVM for classification of celiac disease. In: 2011 7th International Symposium on Image and Signal Processing and Analysis (ISPA), pp. 715–720. IEEE (2011)
14. Mathew, S., Nadeem, S., Kaufman, A.: CLTS-GAN: color-lighting-texture-specular reflection augmentation for colonoscopy. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13437, pp. 519–529. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16449-1_49
15. Mathew, S., Nadeem, S., Kumari, S., Kaufman, A.: Augmenting colonoscopy using extended and directional cyclegan for lossy image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4696–4705 (2020)
16. Ozyoruk, K.B., et al.: Endoslam dataset and an unsupervised monocular visual odometry and depth estimation approach for endoscopic videos. Med. Image Anal. **71**, 102058 (2021)
17. Rivoir, D., et al.: Long-term temporally consistent unpaired video translation from simulated surgical 3D data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3343–3353 (2021)

18. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4104–4113 (2016)
19. Schönberger, J.L., Zheng, E., Frahm, J.-M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 501–518. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_31
20. de Souza Jr, L.A., et al.: Assisting barrett's esophagus identification using endoscopic data augmentation based on generative adversarial networks. Comput. Biol. Med. **126**, 104029 (2020)
21. Xu, J., et al.: OfGAN: realistic rendition of synthetic colonoscopy videos. In: Martel, A.L., et al. (eds.) MICCAI 2020. LNCS, vol. 12263, pp. 732–741. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59716-0_70
22. Yamane, H., et al.: Automatic generation of polyp image using depth map for endoscope dataset. Procedia Comput. Sci. **192**, 2355–2364 (2021)
23. Zeng, Y., Fu, J., Chao, H.: Learning joint spatial-temporal transformations for video inpainting. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12361, pp. 528–543. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58517-4_31
24. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV) (2017)