Towards Automated Quantification of Arteriole Formation

Boris Babenko, Jessica A. DeQuach, Anthony J. Monteforte, Clifford P. Mao, Raymond Pasemen, Karen L. Christman, and Serge Belongie

***Abstract*—Tissue engineering is an interdisciplinary field that offers the promise of improving, repairing and/or replacing damaged tissue in the human body. Research in this area involves the development of various biomaterials and processes that facilitate the fabrication of such tissue. Quantifying vascular cell infiltration and new vessel formation is one technique for gauging the success of tissue regeneration and biocompatibility. This task is typically done manually by a trained expert or technician but requires an intensive amount of time and meticulous effort. Automation of this procedure would be beneficial for faster data turnaround time and would also be useful for a broad range of applications. In this project we aim to ease the burden of doing such analysis via modern computer vision techniques. While the state of the art in computer vision still requires expert oversight, the long-term goal of our work is to automate this process as much as possible.**

***Index Terms*—Neovascularizaton, tissue engineering, object detection, blood vessel**

I. Introduction

Tissue engineering is a fast growing, interdisciplinary field that aims to develop technology for improving, repairing and/or replacing damaged tissue in the human body. Research in this area involves the development of various biomaterials as scaffolds to generate in vitro tissue, which is subsequently implanted, or as in situ forming scaffolds to promote endogenous repair in situ. A number of methods exist to determine whether the material is helping the damaged area:

- Functional improvement
- Cellular infiltration
- Inflammatory response
- Neovascularization (see Fig. 1).

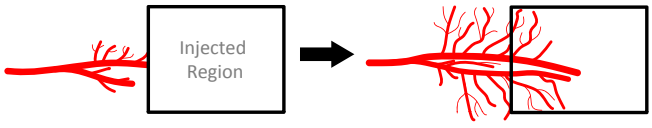


Fig. 1. New blood vessels may infiltrate the tissue engineering scaffold.

In this work we focus on the latter method, which consists of counting the blood vessel density in the biomaterial. After a sample is sectioned and stained, blood vessels appear as oval shapes (i.e. a slice through the vessel, which is tubular); see Fig. 2. This form of evaluation is used in a number of studies [1]–[8] and is advantageous to determine blood vessel density, and hence, vascularization of the biomaterial and potential integration with the host. Nevertheless, manually counting vessels is an extremely labor intensive task – a typical experiment can involve going through up to a thousand images, each of which must be scanned carefully as vessels are a range of size from 10 μm - hundreds of microns. Furthermore, since there are many ambiguous cases and each expert annotator is inevitably biased, all images in an experiment must be labeled by a single person for the sake of consistency. In other words, this task cannot be split amongst many people.

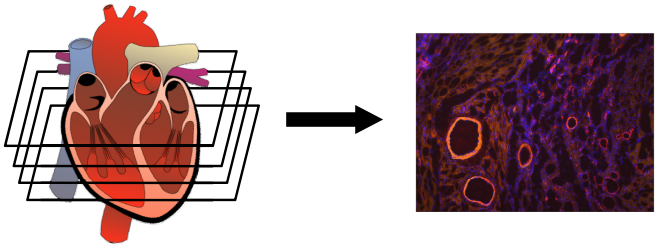


Fig. 2. Tissue is sectioned, stained with fluorescent antibodies and captured.

An automated system is extremely appealing in the above scenario: not only would such a system dramatically reduce the amount of work a researcher must do, a deterministic algorithm is perfectly consistent. There has been some success in developing algorithms for detecting and counting simple structures like cell nuclei [9], [10]. However, such algorithms typically rely on hand-tuned rules and simple image operations like thresholding that are not applicable for more complicated structures. While systems have been proposed for outlining blood vessels in profile views (e.g. in retinal images [11], [12]), to the best of our knowledge no system exists for detecting and counting of vessels in a cross section view.

The contributions of this paper are twofold: first, we collect manual annotations from several experts for a large collection of images. With this data, we evaluate the human error on the blood vessel labeling task, establishing a baseline of performance to which we can compare automated systems. We intend to make this data (images and ground truth annotations) public, enabling other computer vision researchers to develop algorithms for this domain. Second, we propose a detection system composed of state-of-the-art computer vision techniques, and evaluate its performance. In scenarios where a high degree of accuracy is required, it may be necessary for an expert to clean up the results of the automated system; we

B. Babenko and S. Belongie are with the Department of Computer Science and Engineering, UC San Diego.

J. DeQuach, A. Monteforte, C. Mao, R. Pasemen and K. Christman are with the Department of Bioengineering, UC San Diego.

explore this scenario and measure the utility of our system in reducing the amount of work a human annotator must do.

II. Data Collection

In this section we describe the images we collected for our study, and the process through which we collected annotations for these images.

A. Images

The images for our experiments come from two different studies: one using a scaffold in cardiac tissue and one with a scaffold injected into skeletal muscle (details below); see Fig. 5 for examples of images. These studies aimed to measure the efficacy (via blood vessel count) of an injectable biomaterial in helping to alleviate damage from ischemia. The experiments were done on rats. Note that the purpose of our study is to measure accuracy of vessel detection and the particular groups that our images come from are not relevant to the study; therefore, we collected a mix of images from both the control and experimental groups from these studies. In total, we collected 149 images from the Skeletal Muscle experiment and 148 images from the Cardiac Muscle experiment. The three color channels of the images (red, green and blue) corresponded to the following stains: anti-smooth muscle actin (red) to label arterioles, isolectin (green) to label endothelial cells, and Hoechst to label nuclei (blue). The red channel is primarily useful for detection of vessels, but the other channels can occasionally help in determining corner cases. In all images we first labeled the injection region by drawing a polygon on each image (more details in Section II-B).

1) Skeletal Muscle: Ischemia was induced by removal of the femoral artery. One week post-injury the rats were injected with a biomaterial, and then sacrificed at several time points post injection. The treated muscle was excised and sectioned with a spacing of 500 μm apart. The injection region of the sample was determined by hematoxylin and eosin staining, five evenly spaced slides from the injection region were then stained with the stains listed above. Images throughout the injection region were taken at 200x using a Carl Zeiss Observer D.1. with Axiovision software.

2) Cardiac Muscle: A scaffold was utilized in a rat myocardial infarction (MI) model. Ischemia reperfusion was performed, and one week post-MI the scaffold was injected into the heart. Staining was done same as above. As a clear injection region could not be determined, five images throughout the infarction were taken at 200x.

B. Annotation

To facilitate the annotation of the images we developed a Graphical User Interface (GUI) which allows a user to navigate through images and draw ellipses around blood vessels(the GUI is an extension of the bounding box labeler in [13]); see Fig. 3. The interface also allows the user to zoom in and out, change the contrast of the image, and turn various channels on and off. We recruited a total of 5 annotators (whom we refer to as A_0, A_1, A_2, A_3 and A_4) to go through the SM dataset and 4 annotators (similarly named) to go through the CM dataset; all annotators were instructed on how to annotate these images. In particular, the instructions explained that vessels appear as near-elliptical regions that were smooth muscle actin positive, exhibited a clearly defined lumen, and had an average diameter greater than $10\mu m$. For some of the experiments we had some of our annotators review and fix the output of our automated system. In these cases the interface displayed the detected vessels and the annotator was allowed to delete, move or add new vessels to the image; we instructed the annotators to only fix or remove clear mistakes (e.g. if a detection was reasonably close, we asked them to leave it as is).

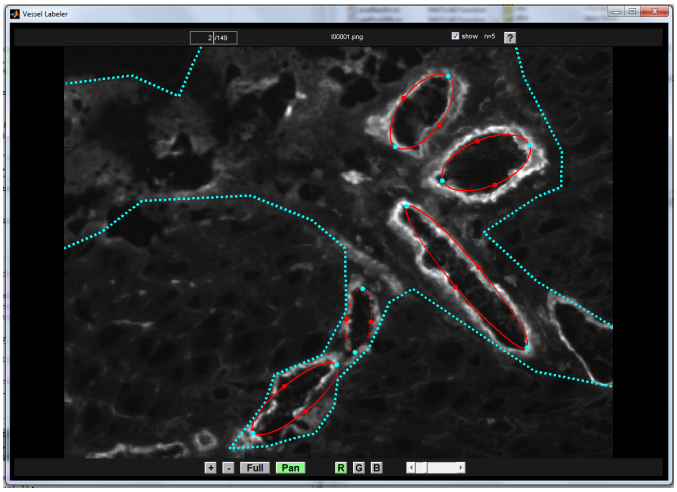


Fig. 3. A screenshot of the Graphical User Interface (GUI) we used for labeling blood vessels. The dashed cyan polygon represents the injection region; our annotators were asked to only label vessels in this region. Each red ellipse corresponds to a labeled vessel.

Fig. 6 shows some basic statistics of the collected data; Section IV contains more detailed analysis. The annotation GUI, images and our collected annotations will be made available on the web.

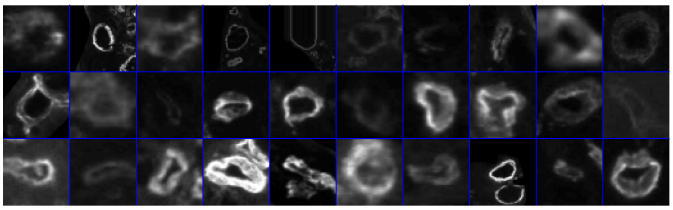
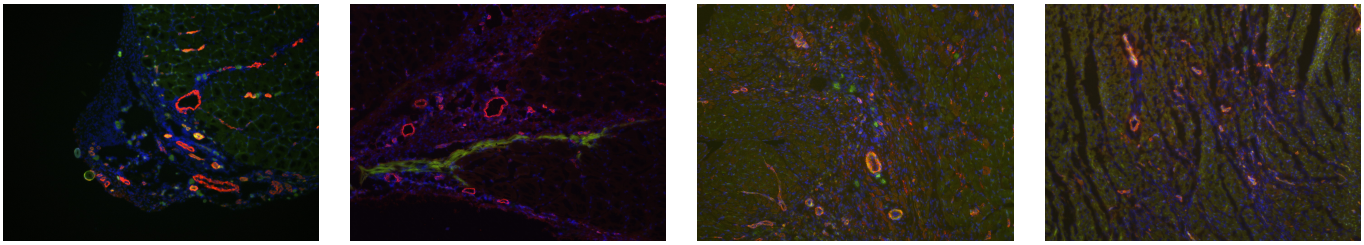


Fig. 4. Blood vessels after cropping and resizing from training data; these and other image patches are fed into the AdaBoost training procedure to train our detector. Note the large variability in brightness, shape, wall thickness, and contrast.

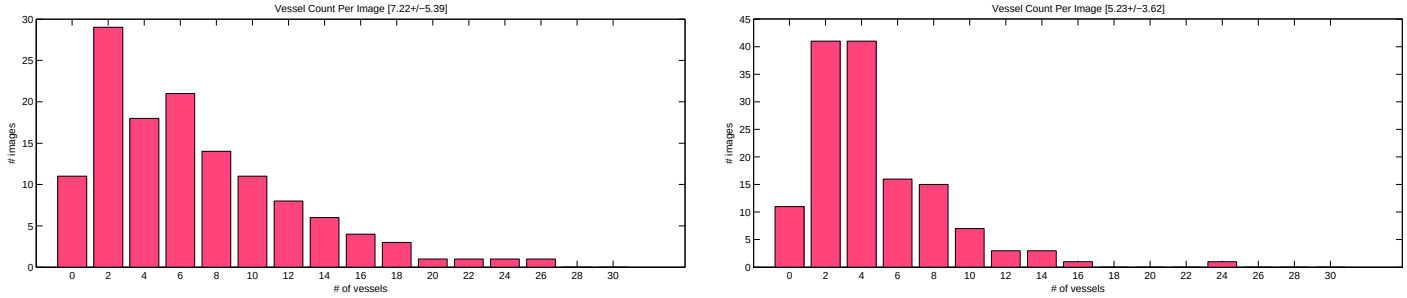
III. Proposed System

The goal of our system is to detect blood vessels in a given input image; the output of the system can be then used as is, or can be checked and fixed by a human annotator. Since blood vessels in the tissue are tubular and our images capture slices through the tissue, vessels appear roughly elliptical (though due to noise, folding and tearing of tissue, the exact shapes



Skeletal Muscle Data (SM) Cardiac Muscle Data (CM)

Fig. 5. Example images from the two datasets. The three color channels of the images (red, green and blue) corresponded to the following stains: smooth muscle actin (red), isolectin (green) and Hoechst (blue). The red channel is the primary channel used for detecting arterioles, which appear as red ovals in the images. Notice that the red stain also attaches to much of the background causing noise, especially for the CM images.



Skeletal Muscle Data (SM) Cardiac Muscle Data (CM)

Fig. 6. Dataset Statistics: histograms of vessel counts for the two datasets.

can vary; c.f. Fig. 4). Thus, each detected blood vessel will be characterized by an ellipse (i.e. there are five parameters: x and y location of the center, major and minor axis lengths and orientation). Such precise localization information allows for a a number of statistics to be computed, ranging from simple (e.g. number of vessels per image) to more complex (e.g. average size of vessels). Furthermore, we found that for the purposes of having a human annotator fix the output of the system, simpler forms of output (e.g. only x and y location) cause more confusion as to which detections should be fixed.

Over the last decade object detection has seen great success in the computer vision community (see e.g. [14]). Popular applications of such techniques include face detection [15] and pedestrian detection [16], [17], although generic object categories have also been studied [14]. The success of these approaches is mainly due to the use of statistical machine learning, versus the more traditional rule-based methods. Thus, the most common approach to object detection is to train a discriminative classifier such as AdaBoost [18] or a Support Vector Machine [19] using labeled examples, and apply the trained classifier to a novel image in a sliding window fashion. To avoid many overlapping detections in the output, this step is typically followed by non-maximal suppression (NMS). For multi-scale detection, the classifier is applied on a scale pyramid of the image. Thanks to a number of engineering tricks, many of the most recent systems are able to perform multi-scale detection at a fraction of a second per image [20].

To engineer our system we would like to follow in the footsteps of the above work and utilize statistical learning tools (details on how we gathered training data are in Section II). However, most the of mentioned object detection frameworks output only 3 parameters for each detection: x, y and scale; this is insufficient to meet our goals. We therefore approach our problem in a coarse-to-fine manner: first we run a multi-scale detector to retrieve rough blood vessel locations, and for each detected vessel we perform more thorough pose estimation (see Fig. 7 for a diagram of our system). In the following sections we describe these two steps in more detail.

A. *Detection*

To perform multi-scale detection we turn to a recently proposed system that achieves state of the art results on many of the popular pedestrian detection benchmarks [17]. This method trains a classifier via the AdaBoost algorithm [18], where each weak classifier is a decision tree of depth 2; for efficiency, a soft cascade is used [21]. The features used are Haar-like features (i.e. sums inside rectangles [15]) computed over a number of channels. In our system we implemented and used the following channels for these features:

- grayscale image
- gradient magnitude
- gradient orientation histogram channels (6 orientations)
- local standard deviation, computed over 5×5 window

All of the above were computed only over the red channel, which corresponds to the smooth muscle stain. This is the primary stain that experts use to annotate the images; the use of the other channels could easily be built into this framework, but we leave this for future work. A large pool of Haar-like features (each consisting of a number of rectangles with an associated channel and rel-valued weight) were generated randomly and plugged into the AdaBoost training procedure which implicitly performs feature selection. While these particular features were designed and optimized for the pedestrian

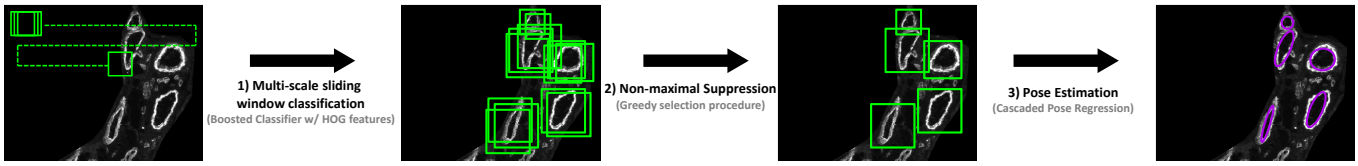


Fig. 7. System Pipeline: multi-scale sliding window detection, followed by pose regression.

detection task, we found that the system worked surprisingly well when applied to our domain. We therefore made very few modifications to this pipeline, though designing more domain-specific features remains a possible future work.

To train the classifier we cropped out square image patches around the annotated blood vessels in the training data and resized them all to 50×50 pixels (see Fig. 4). For each vessel we also added its reflection around the x and y axes to the training data. Negative data included all non vessel area in the labeled images; since training with all negative data is intractable, we used a bootstrapping strategy (with a total of 3 bootstrapping rounds).

For non-maximal suppression we used a simple greedy heuristic (as done in [17], [22]): go through all detections in order of their score and for each one that has not yet been suppressed, suppress its neighbors. To find the neighbors of a detected window, we used the PASCAL criteria to measure overlap [14] and threshold this at one half.

B. Pose Estimation

Once the rough location of vessels have been determined, we seek to fit an ellipse to the vessel more precisely. Various methods for such a task exist in the literature; for example, the Hough transform [23] is a popular method of detecting various shapes in images. However, due to the amount of variability and noise in our data, we once more turned to state of the art computer vision methods. We adopted a recently published learning based method for pose estimate called Cascaded Pose Regression (CPR) [24]. As is the detection method, this method for pose estimation is also based on statistical machine learning and is trained using labeled examples. Specifically, the algorithm places a random guess of the pose and uses an iterative procedure to improve the pose; since the initial guess may be bad, multiple starting positions are used and the final results are clustered to produce the final pose. In principle this method is not unlike Active Shape models [25]; the main difference is that the iterative steps are informed by trained rules rather than gradient descent on a designed objective function. At each iteration, the procedure computes *pose indexed* features at the current estimated pose, and feeds these into a simple fern regressor, which predicts how the pose should change to best fit the image. We refer the reader to [24] for more details on CPR.

IV. Results & Analysis

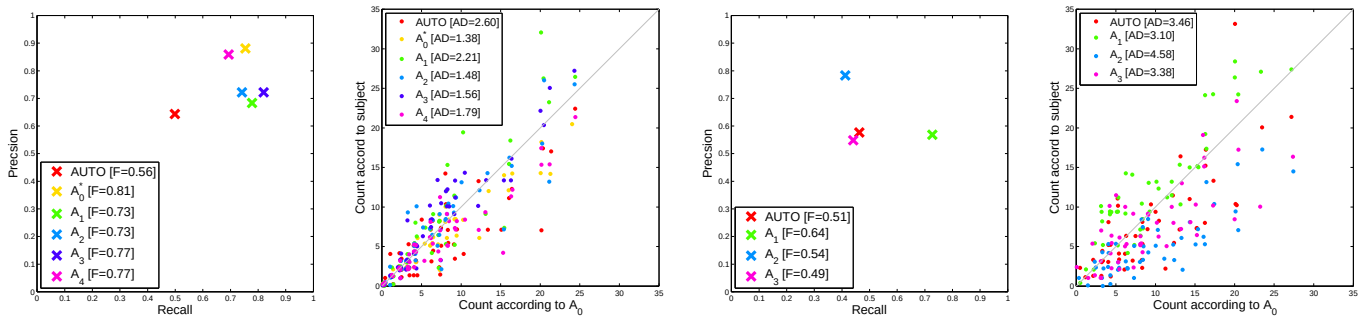
In this section we present an analysis of the annotations our experts provided, as well as the accuracy of our proposed automated system. We begin with an overview of our evaluation criteria.

A. Evaluation Criteria

For the purpose of evaluating accuracy, we arbitrarily chose one of the annotators to be the "ground truth" that all other annotators and the automated system are compared to (we refer to this annotator as A_0 in the plots). We compute two types of error metrics: 1) absolute count differences, and 2) precision-recall curves. The former metric takes into account *how many* vessels were detected, and ignores their location. This is a more lenient form of evaluation in the sense that mistakes (e.g. false detections of misses) get averaged out. However, if the primary goal of an automated system is to predict the count, this evaluation criteria is appropriate. The latter metric does take into account vessel locations: for each detected vessel in the ground truth annotation we look for a detected vessel in the test annotation that has a large spatial overlap. In particular, we use the PASCAL criteria to compute overlap (area of intersection divided by the area of the union). Note, however, that the PASCAL criteria were designed for axis-aligned bounding boxes, while we are dealing with elliptical regions. Since computing the intersection area of ellipses is non-trivial, we instead compute the intersection and union of the smallest axis-aligned bounding box around both ellipses. This approximation can result in underestimates of intersection area (e.g. when the ellipses are rotated at close to 45 degree angles); therefore, instead of the commonly used threshold of 0.5, we instead of a threshold of 0.3 to determine whether two vessel detections match. Each unmatched vessel detection in the "ground truth" is counted as a *false negative*, while each unmatched vessel detection in the test is counted as a *false positive*. Using these numbers we can calculate the precision (number of correctly detected vessels in the test annotations divided by the total number of detected vessels in the test annotations), and recall (number of correctly detected vessels divided by the total number of vessels in the ground truth). To summarize the precision-recall points in one number, we also calculate the F-score, $(2\frac{precision*recall}{precision+recall})$. Finally, we note that for both datasets we used the first 100 images for training and the rest for testing.

B. Analysis

1) Inter-annotator Error: Inter-annotator error refers to the difference in annotations between different human annotators. In Fig. 8 we see that expert annotators A_1, A_2, A_3 and A_4 achieved F-scores ranging between 0.73 and 0.77 on the SM dataset and between 0.49 to 0.64 on the CM dataset, and mean absolute difference scores ranging between 1.48 to 2.21 on the SM dataset and between 3.10 to 4.58 on the CM dataset. This suggests that the CM dataset is significantly more



Skeletal Muscle Data (SM) Cardiac Muscle Data (CM)

Fig. 8. Results for both datasets. In all cases A_0 refers to the annotator who we chose to be the ground truth; all other annotators (A_1, A_2, A_3 and A_4) are compared to A_0. *AUTO* refers to our automated system. A_0^* refers to A_0's second set of annotations (see text for details). In the legends, F refers to the F-score, and AD refers to the mean absolute difference.

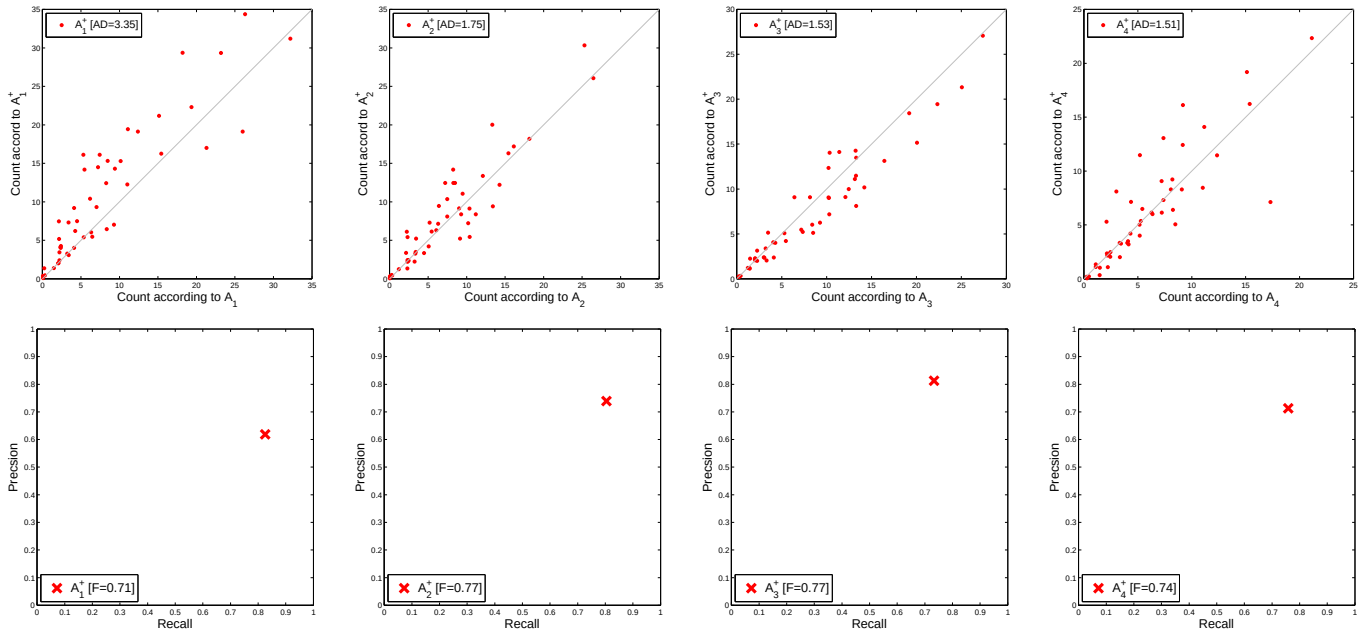


Fig. 9. Results: Proposed System + Annotator on the SM dataset. Here we compare the original annotations (i.e. from scratch) of each annotator with the result of each annotator fixing the output of our system.

ambiguous and will be more challenging for an automated system. Furthermore, these numbers establish a baseline to which we can compare an automated system.

2) Intra-annotator Error: Intra-annotator error refers to the difference in annotations for the same human annotator. That is, if an expert manually annotated the same images twice (with some time in between), we would like to know how much difference there would be between the two sets of annotations. To measure this, we had our ground truth annotator, A_0, annotate the SM dataset from scratch a second time, a month after the original annotations were collected. Fig. 8 shows the results (denoted as A_0^*). Not surprisingly, both the F-score and mean absolute difference metrics are the best for A_0^* when compared to the other annotators. This suggests a certain level of consistency in how a single expert annotates the images. However, it is also important to note that the expert is *not* perfectly consistent, and the intra-annotator error is indicative of the difficulty of this dataset.

3) Proposed System: The proposed system performance is also shown in Fig. 8, denoted as '*AUTO*'. On the SM dataset our system performs worse than the other expert annotators, though the mean absolute difference is quite competitive (recall that precision-recall analysis is a more strict form of evaluation). Surprisingly, our system achieves *better* performance than one of the expert annotators on the more difficult CM dataset. Note that the images in this dataset have much more ambiguity and noise than images in SM, and our system is trained on A_0's annotations. Hence, it is plausible that our system is able to better mimic A_0's behavior than the other human annotators.

4) Proposed System + Annotator: Since computer vision may not be mature enough to match human performance on these complex datasets, we envision that our system (and others like it) will be used in conjunction with expert oversight. To this end, we asked some of our annotators to go over the SM dataset one more time, but this time rather than annotating from scratch we pre-populated the images with

the annotations of our system and asked the annotators to fix any errors they see. Our goal with this experiment was to see how much the resulting annotations differ from the original annotations of these experts. These results are shown in Fig. 9, with A_1+, A_2+, A_3+, A_4+ denoting the results of the corresponding annotators fixing the output of our system. Overall we observe that the difference between annotations that were done from scratch to the annotations that were done with the aid of our system is similar to the inter-annotator differences. Furthermore, we wanted to measure how many vessel annotations had to be fixed as compared to how many vessels total were in the images. Table I shows the total number of vessels, the number of deletions and the number of additions that each annotator had to make (if the location of a vessel was changed significantly, it was counted as one deletion and one addition). Overall, using our system to pre-populate annotations saved a significant amount of time versus annotating from scratch.

TABLE I
PROPOSED SYSTEM + ANNOTATOR

	Total # of detected vessels	# Deletions	# Additions
A1+	507	46	295
A2+	372	61	175
A3+	340	63	145
A4+	280	53	75

V. CONCLUSION AND FUTURE WORK

Detecting and counting blood vessels in fluorescently stained images is an important form of analysis for tissue engineering, and in this paper we have studied how to automate this procedure. The system we proposed in this paper is made up of state-of-the-art computer vision techniques that are trained on labeled examples. We established an accuracy baseline by measuring the accuracy of expert annotators with respect to themselves, and then evaluated our proposed system. Furthermore, we analyzed the performance of our system when used in conjunction with human oversight. If the end goal is to simply count vessels in images, our system already achieves performance close to human annotators (measured by the absolute count difference), especially on the more ambiguous CM dataset. Alternatively, if the locations and size measurements of the vessel are important for analysis, our system can be used to pre-populate annotations, saving an expert time.

We hope that this work draws attention to the blood vessel detection problem and suspect that future research into features specific to this domain will improve the accuracy of the system. Another interesting avenue for future work is to combine the multiple expert annotations into a single ground truth, and use this as training data to achieve a system that is less biased towards one particular expert annotator.

REFERENCES

[1] K. Christman, Q. Fang, M. Yee, K. Johnson, R. Sievers, and R. Lee, "Enhanced neovasculature formation in ischemic myocardium following delivery of pleiotrophin plasmid in a biopolymer," *Biomaterials*, vol. 26, no. 10, pp. 1139–1144, 2005.

[2] P. Krishnamurthy, J. Rajasingh, E. Lambers, G. Qin, D. Losordo, and R. Kishore, "Il-10 inhibits inflammation and attenuates left ventricular remodeling after myocardial infarction via activation of stat3 and suppression of hur," *Circulation research*, vol. 104, no. 2, pp. e9–e18, 2009.

[3] W. Lee, H. Wei, W. Lin, Y. Yeh, S. Hwang, J. Wang, M. Tsai, Y. Chang, and H. Sung, "Enhancement of cell retention and functional benefits in myocardial infarction using human amniotic-fluid stem-cell bodies enriched with endogenous ecm," *Biomaterials*, 2011.

[4] Y. Yeh, H. Wei, W. Lee, C. Yu, Y. Chang, L. Hsu, M. Chung, M. Tsai, S. Hwang, and H. Sung, "Cellular cardiomyoplasty with human amniotic fluid stem cells: in vitro and in vivo studies," *Tissue Engineering Part A*, vol. 16, no. 6, pp. 1925–1936, 2010.

[5] J. Singelyn, J. DeQuach, S. Seif-Naraghi, R. Littlefield, P. Schup-Magoffin, and K. Christman, "Naturally derived myocardial matrix as an injectable scaffold for cardiac tissue engineering," *Biomaterials*, vol. 30, no. 29, pp. 5409–5416, 2009.

[6] S. Seif-Naraghi, M. Salvatore, P. Schup-Magoffin, D. Hu, and K. Christman, "Design and characterization of an injectable pericardial matrix gel: A potentially autologous scaffold for cardiac tissue engineering," *Tissue Engineering Part A*, vol. 16, no. 6, pp. 2017–2027, 2010.

[7] J. A. DeQuach, C. C. Joy E. Lin, D. Hu, M. A. Salvatore, F. Sheikh, and K. L. Christman, "Injectable skeletal muscle matrix hydrogel promotes neovascularization and muscle cell infiltration in a hindlimb ischemia model," *(submitted)*.

[8] K. Christman, A. Vardanian, Q. Fang, R. Sievers, H. Fok, and R. Lee, "Injectable fibrin scaffold improves cell transplant survival, reduces infarct expansion, and induces neovasculature formation in ischemic myocardium," *Journal of the American College of Cardiology*, vol. 44, no. 3, pp. 654–660, 2004.

[9] A. Carpenter, T. Jones, M. Lamprecht, C. Clarke, I. Kang, O. Friman, D. Guertin, J. Chang, R. Lindquist, J. Moffat *et al.*, "Cellprofiler: image analysis software for identifying and quantifying cell phenotypes," *Genome Biology*, vol. 7, no. 10, p. R100, 2006.

[10] J. Harada, K. Bower, A. Orth, S. Callaway, C. Nelson, C. Laris, J. Hogenesch, P. Vogt, and S. Chanda, "Identification of novel mammalian growth regulatory factors by genome-scale quantitative image analysis," *Genome research*, vol. 15, no. 8, p. 1136, 2005.

[11] T. Chanwimaluang and G. Fan, "An efficient blood vessel detection algorithm for retinal images using local entropy thresholding," in *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on*, vol. 5. IEEE, 2003, pp. V–21.

[12] X. Jiang and D. Mojon, "Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 131–137, 2003.

[13] P. Dollár, "Piotr's Image and Video Matlab Toolbox (PMT)," http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html.

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results." [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html

[15] P. Viola and M. Jones, "Fast multi-view face detection," in *CVPR*, 2001.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[17] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *BMVC*, 2009.

[18] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Comp. and Sys. Sci.*, vol. 55, pp. 119–139, 1997.

[19] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

[20] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *BMVC*, 2010.

[21] L. Bourdev and J. Brandt, "Robust object detection via soft cascade," 2005.

[22] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," *PAMI*, 2009.

[23] D. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[24] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression," in *CVPR*, 2010.

[25] T. Cootes, C. Taylor, D. Cooper, J. Graham *et al.*, "Active shape models-their training and application," *Computer vision and image understanding*, vol. 61, no. 1, pp. 38–59, 1995.

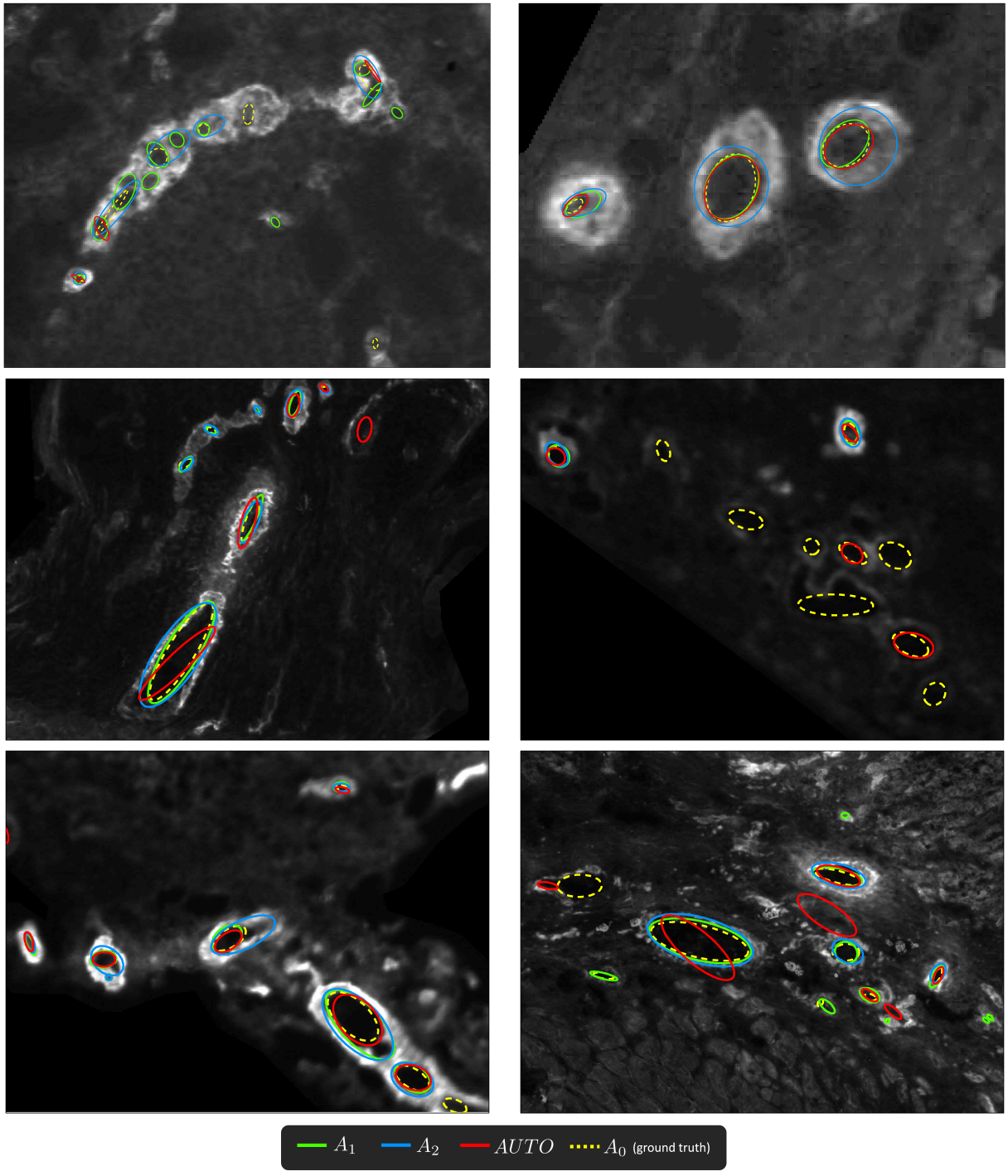


Fig. 10. Examples of annotations from a few of our annotators on the SM dataset, as well as the output of our system. These examples demonstrate the difficulty of the task, even for expert annotators.