Reproducibility in Computer Vision: Towards Open Publication of Image Analysis Experiments as Semantic Workflows

Ricky J. Sethi
Fitchburg State University
Fitchburg, MA 01420
rickys@sethi.org

Yolanda Gil
USC Information Sciences Institute
Marina del Rey, CA 90292
gil@isi.edu

Abstract**—Reproducibility of research is an area of growing concern in computer vision. Scientific workflows provide a structured methodology for standardized replication and testing of state-of-the-art models, open publication of datasets and software together, and ease of analysis by re-using pre-existing components. In this paper, we present initial work in developing a framework that will allow reuse and extension of many computer vision methods, as well as allowing easy reproducibility of analytical results, by publishing datasets and workflows packaged together as linked data. Our approach uses the WINGS semantic workflow system which validates semantic constraints of the computer vision algorithms, making it easy for non-experts to correctly apply state-of-the-art image processing methods to their data. We show the ease of use of semantic workflows for reproducibility in computer vision by both utilizing pre-developed workflow fragments and developing novel computer vision workflow fragments for a video activity recognition task, analysis of multimedia web content, and the analysis of artistic style in paintings using convolutional neural networks.**

1. Introduction

The limited reproducibility of scientific research is a growing cause for concern, sometimes described as a "reproducibility crisis" [1], [2], [3], [4]. Lack of reproducibility may lead to unreliable comparisons to other work, insufficient verification of the state-of-the-art, inadvertent errors in publications, and a general inability to efficiently build upon previous work.

The importance of reproducible computational research has come to the forefront in computer vision, as evidenced by premier conferences like Computer Vision and Pattern Recognition (CVPR) requiring reviewers to comment on the reproducibility of papers[1], and the International Conference on Image Processing (ICIP) having round tables on reproducibility [5]. In the closely related field of machine learning, prominent members have argued for the need to share software and datasets to facilitate experimentation and learning [6]. In order to combat the growing problems with reproducibility, some researchers have suggested computational solutions for specific areas [7], [8], [9]; but these are often ad hoc programs written in specific languages that do not accommodate well applications which combine pre-existing, heterogeneous software such as computer vision.

An elegant solution to the problem of creating standardized, reproducible research is to utilize a scientific workflow system [10]. Sharing workflows in addition to data and software is part of the recently proposed Reproducibility Enhancement Principles (REP) [11]. Scientific workflows have been used for reproducible and reusable research in many science fields but have not been previously applied to computer vision. Computer vision presents a unique challenge due to its inherently multi-disciplinary methodologies, widely heterogenous codebases, and dearth of pre-existing foundational computer vision workflows.

This paper describes a collection of workflows for computer vision applications, which we have developed in the WINGS semantic workflow system [12], [13]. The paper illustrates the use of these workflows for reproducible computer vision research in three case studies: activity recognition in video, multimedia analysis of web site content, and analysis of artistic style in paintings. The paper also discusses how fragments of the workflows can be reused for new tasks, not only to save effort but to build on well-tested and proven analysis methods.

In addition to reproducibility and efficiency, our goal is to enable non-experts to reuse these workflows. In particular, we are interested in allowing geoscientists to analyze images collected in the field [14], art scholars to do computational analysis of artistic style [15], and allowing students with limited or no programming background to learn the capabilities offered by image analysis [16].

We begin with an overview of the workflows that we have created. We then describe the use of these workflows in the three case studies to demonstrate reproducibility and reuse. We conclude with a summary of the benefits of workflows to support reproducible research.

1. http://tab.computer.org/pamitc/archive/cvpr2010/submission/

2. Semantic Workflows in WINGS for Computer Vision Tasks

Computational workflows capture an end-to-end analysis composed of individual analytic steps as a dependency graph that indicates dataflow links as well as control flow links among steps. They represent complex applications as a dependency network of individual computations linked through control or data flow. Each workflow step is a *software component* that can be implemented in a different language from others, each with one or more data inputs and one or more data outputs.

A *workflow fragment* consists of one or more components, together with their data inputs and outputs. A *full workflow* combines multiple workflow fragments connected together by having one or more data outputs of one fragment becoming data inputs of another fragment. An example is shown in Figure 1, where several fragments have been combined to create a full workflow to analyze the movement over time of groups of objects in a video.

Several workflow systems have been developed with a variety of capabilities, and used in many areas of science [10]. However, none have been applied to computer vision. In our work, we use the WINGS semantic workflow system as it has three key features to support reproducibility: semantic constraints to validate a workflow for data [13], abstractions to represent classes of components [12], and the ability to publish workflows as web objects following linked open data principles [17]. A detailed discussion on how WINGS supports reproducibility and validation with examples from clinical omics can be found in [18].

An overview of the collection of workflow fragments for computer vision tasks we have created is shown in Table 1. Some of the fragments were built by us, but most are built using exisitng open source packages that are popular in computer vision. We have fragments built with OpenCV for image filtering, segmentation, and edge detection. We also have fragments for Latent Dirichlet Allocation (LDA), some of them implemented in the Mallet software. We also include fragments that build on the recently released Google TensorFlow machine learning framework for deep learning. Such predefined workflow fragments make complex analytics expertise readily available to new users, who can compose them to create new full workflows. The components that make up workflow fragments can be written in heterogeneous languages: e.g., some components are in Java, others in MATLAB, and still others in C++, but the language of choice is irrelevant as the components are integrated into the workflows without reliance upon their individual implementation idiosyncrasies. This is possible because each individual program is converted into a workflow component via a short wrapper shell script (usually 3-5 lines of code).

WINGS adds semantic constraints to the workflows to express the requirements of the different workflow components. Users can export these workflows and make them available as part of a workflow library like the WINGS standard repository so that other researchers can directly utilize any single component (or the entire component collection) in their own workflows by simply importing those web objects. The exported workflows can also be adapted by adding or changing any component. Thus, these *workflows web objects* allow full reproducibility of the original experiments by examining or executing them online or downloading and importing them locally.

We show both the re-use of existing workflow fragments from the WINGS standard repository as well as the development of novel computer vision workflow fragments in the analysis of three fundamental computer vision tasks: a video activity recognition application, an analysis of multimedia web site content, and the analysis of artistic style using convolutional neural networks. Workflows for all these tasks are covered in detail in the following sections.

3. Case Study: Video Activity Recognition Task

For the video activity recognition task, we utilize the the *Group Transition Ratio,* G_{tr}, from [19] to quantitatively define a group as well as the Atomic Group Actions. The G_{tr} is defined as $G_{tr} = \frac{L}{\lambda}$, where λ is the mean free path and L is the characteristic length. The characteristic length is usually a convenient reference length that is a constant of a given configuration.

We use the G_{tr} to identify when a collection of objects can be considered as separate individuals, a group, or a crowd. In addition, we use the time variance of the G_{tr} to determine *when* a collection of objects transitions between being individuals, groups, or crowds. We implemented the G_{tr} in MATLAB as a component in the WINGS workflow system and tested it against the Atomic Group Actions dataset [20]. The workflow is shown in Figure 1. The Evaluation component represented there is a composite representation of multiple statistical evaluation components from Table 1. The re-use of pre-created workflow fragments like the statistical fragments shown there allowed for a rapid development cycle in addition to providing the ability to

Table 1. WORKFLOW FRAGMENTS CREATED FOR COMPUTER VISION TASKS.

Category	Workflow Fragments
Computer Vision	OpenCV components (Optical Flow, Kalman tracker, Mixture of Gaussians, Particle filter, etc.), N-Cuts, PhaseSpace, G_{tr}, $Relative_{Velocity}$, $Relative_{Distance}$, Image Extractor, Background/Foreground Extraction, Neural Algorithm for Artistic Style (Lua/Torch), Neural Algorithm for Artistic Style (TensorFlow)
General Machine Learning	K-Means, Latent Dirichlet Allocation, Mallet, libSVM, Caffe, Convolutional Neural Networks (Lua/Torch), TensorFlow, Adam Optimizer, Recurrent Neural Networks
Statistical Evaluation	Confusion Matrices, Heatmaps, Precision-Recall Curves, ROC Curves, AUC Curves, Equal Error Rate, F-Measure

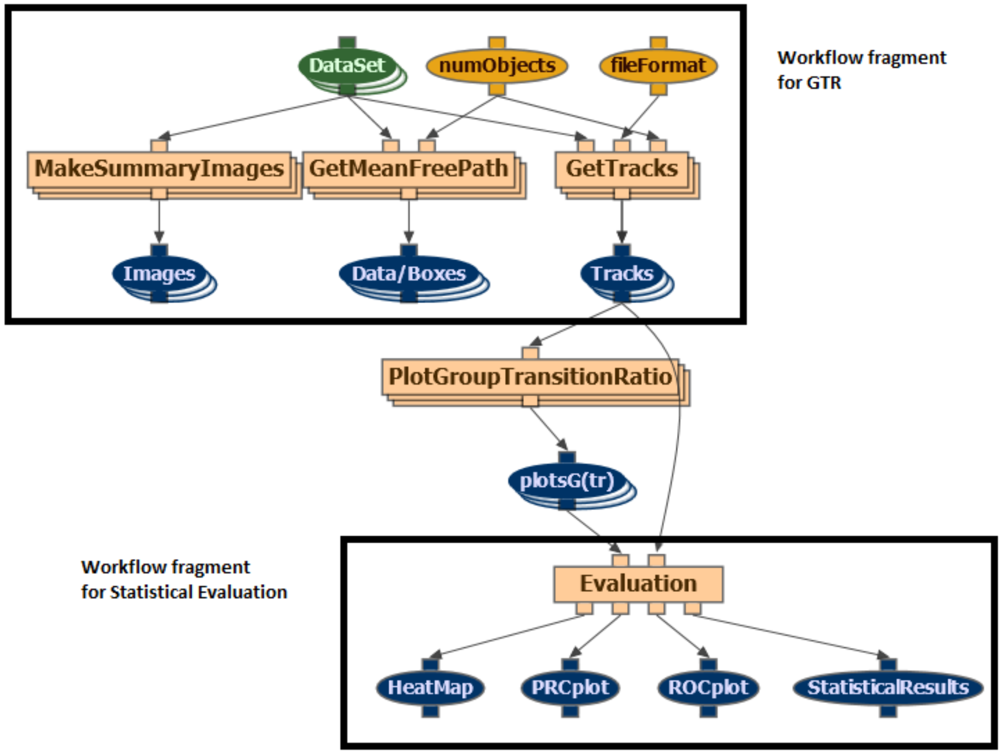
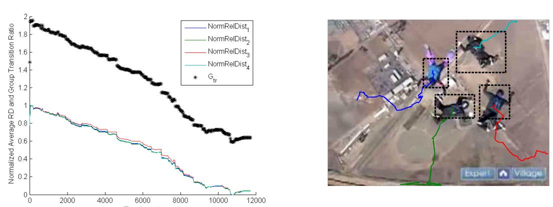
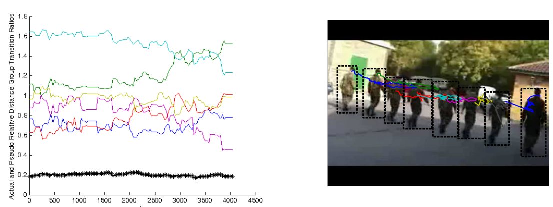


Figure 1. Workflow for the Group Transition Ratio (G_{tr}) Model for the Video Activity Recognition Task.



(a) *Group Formation*: G_{tr} decreases from about 2 to about 0.5. Sample image shows individuals converging and forming a group.



(b) *Group Movement*: G_{tr} stays steady at about 0.2. Sample image shows group moving together.

Figure 2. Workflow Generated Results for the Video Activity Recognition Task.

export the entire analysis and dataset as web objects for reproducibility.

We first show some qualitative results for the G_{tr} analysis shown in Figure 1 that were generated automatically from the workflow. In Figure (2), we show how the G_{tr} varies with time to characterize the action in a video; results for all categories of the G_{tr} graph along with representative frames from the video are overlaid to show the action. The G_{tr} decreases with time for *group formation* in (a), increases with time for *group dispersal*, and stays the same for *group movement* in (b). The thresholds were empirically determined automatically by the workflow to be $G_{tr} \lesssim 0.1$ for crowds, $0.1 \lesssim G_{tr} \lesssim 1.5$ for groups, and $G_{tr} \gtrsim 1.5$ for individuals. More details about this task are described in [20].

4. Case Study: Multimedia Analysis of Web Content

We demonstrate the utility of image analysis workflow fragments by combining them with pre-existing text analysis workflow fragments in order to analyze multimedia content of web sites that contain ads for services in order to detect human trafficking. This project analyzes posts on various sites on the Internet in order to determine if the subject of that post may be a victim of human trafficking and alert law enforcement. Using both the text and image content of posts helps to make a stronger determination of whether or not the subject of the post was trafficked. The combined workflow is described in [21], we focus here on the image analysis aspects.

The initial development of the project had progressed to creating a crawler, which downloads posts from various posting sites, and an extractor, which extracts the text and images and stores them in a database. However, there had been no substantial analysis of the posts in this nascent project. We extended the project to examine both the text of the post (using the text analytics workflow fragments we had already developed, described in [22]), as well as the associated images (using the image analysis workflow fragments we developed); a final determination about trafficking of the subject of the post was made by fusing the results of the text and image analyses via a fusion workflow fragment we developed. Thus, the re-use and re-purposing of workflow fragments allowed a multimedia analysis spanning data domains of text and image analysis, including the fusion of their results in the final determination.

The fragments of the workflow that focus on computer vision are shown in Figure 3. The initial image analysis is done via N-cuts followed by an unsupervised Mallet LDA analysis and a supervised analysis using SVM in a bag-of-words model.

The original development of the HTD crawler/extractor had taken several months; the conversion to WINGS took roughly two days. The extension of the other components took approximately one day, saving effort estimated to be on the order of 300 man-hours of work. Summary results showed an Equal Error Rate of 0.37 and F-Measure of 0.47 for the fusion module that combines image and text analysis results. More details about this workflow can be found in [21].

5. Case Study: Analysis of Artistic Style using Convolutional Neural Networks

In this section, we show the use of workflows for image analysis and processing of paintings. We implemented the newly released and very popular neural algorithm for artistic style developed in [23]. In order to allow maximal flexiblity in the use of the neural algorithm of artistic style [23], we developed two separate implementations as shown in the two workflow fragments shown in Figure 4, which uses the Lua scripting language with the Torch machine learning framework, and Figure 5, which uses Python and Google's TensorFlow machine learning framework. Both of these use fragments for pre-processing that were also developed in Table 1.

The Neural Algorithm of Artistic Style uses deep neural networks (specifically, a Convolutional Neural Network, CNN) to separate the style and content of an image. It designates one image as a style image and one as target image. It then extracts the style from the style image and applies it to the content of the target image to create a new image in the style of the style image.

For example, Figure 6 shows the target image of a scene from Tubingen as presented in the original paper [23]. The algorithm then extracts the style from the Starry Night painting of Van Gogh. This style is applied to the Tubingen image to create a new image of Tubingen in the style of Van Gogh. Similarly, they extract the style of Munch using his painting, The Scream, and apply this to the Tubingen scene, as well.

We reproduce these results in Figure 6 using the workflow fragments shown in Figure 4.

CNNs pass filters of shared weights across overlapping image patches to learn convolutions, also called feature maps. Following the example of the paper, we utilize the publicly available and trained VGG network. The method for determining the style is based on texture extractions using correlations between feature-maps within a layer. The final reconstructed image is created by first randomly generating a white-noise image and then using gradient descent for the optimization. In the workflows in Figures 4 and 5, we use Adam Optimization, an algorithm for first-order gradient-based optimization. Finally, the loss function in the article is a weighted sum of the style and content mean squared error loss functions and is used to generate the final new image. We can re-use the Adam Optimization module in the workflow in Figure 5 if we did not want to use the optimization provided in TensorFlow.

6. Conclusion

The inability to reproduce results of computational algorithms is a significant issue in science. This paper describes a collection of workflow fragments for computer vision and their use in four different image analysis tasks. By using the power of semantic workflows, we provide a framework to export workflows and their associated datasets as web objects, thus allowing for easy and full reproducibility of research. Using such a workflow framework allows for quick deployment and comparison of various approaches and algorithms. It also eases development by incorporating heterogeneous codebases, re-using components and providing standard implementations of popular components, and allowing for easy extension of pre-existing workflows. We are already using some of these workflows to teach data science to non-programmers [16]. In the future, we plan to use these workflows to allow non-experts to apply state-of-the-art computer vision methods. This includes allowing geoscientists to analyze images collected in the field [14] and art scholars to do computational analysis of artistic style [15].

7. Acknowledgments

This research was supported in part by the US National Science Foundation (NSF) with grants ICER-1440323 and ICER-1632211, in part under grant CNS-1019343 to the Computing Research Association for the CIFellows Project, and in part under the National Endowment for the Humanities (NEH) Grant under Award HD-248360-16. We would like to thank Taylor Alarcon, Catherine A. Buell, Alyssa Deng, Hyunjoon Jo, Andrew Philpot, and William P. Seeley for their discussions and their assistance in developing some of the workflow fragments.

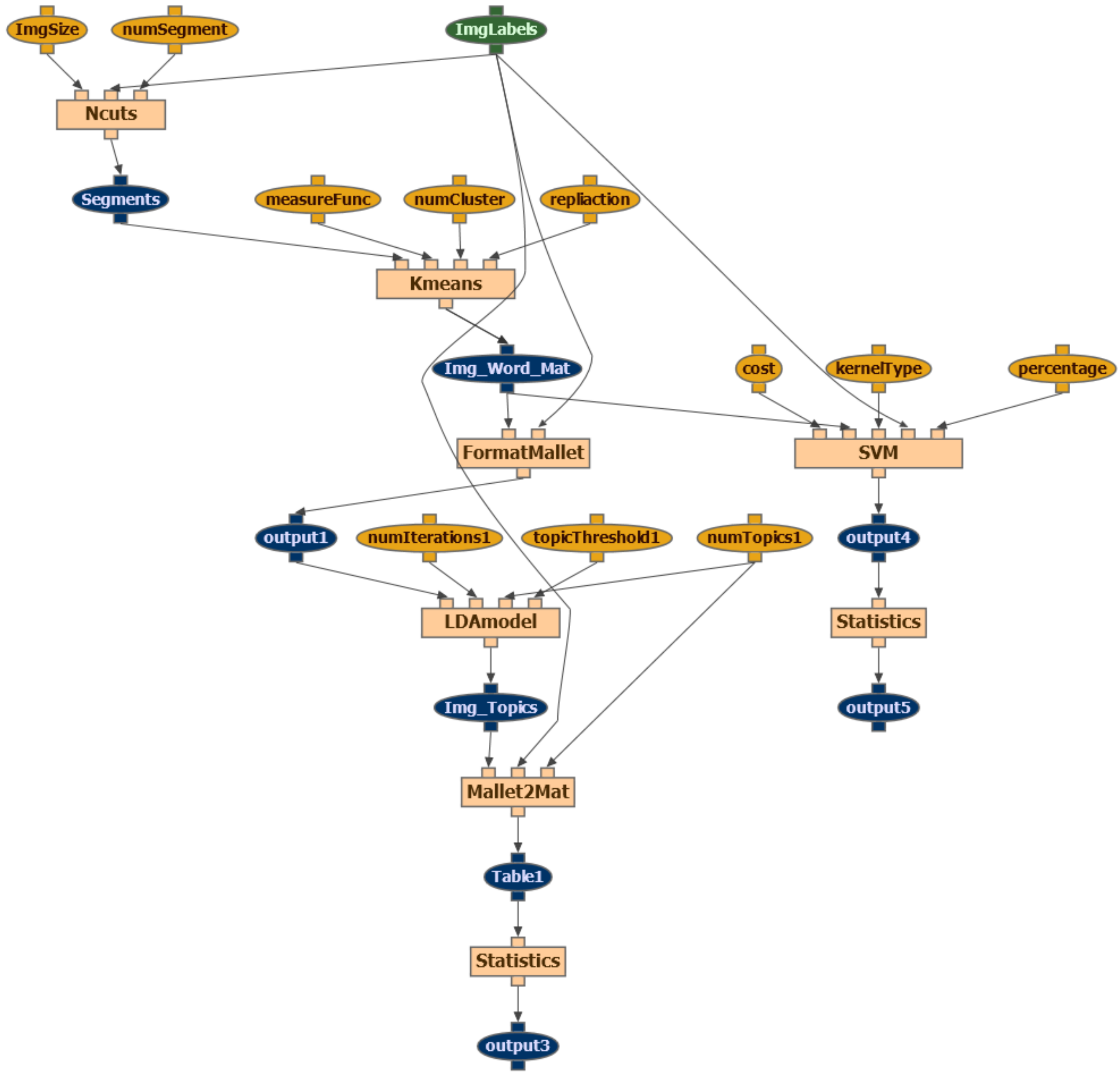


Figure 3. Workflow for the Multimedia Web Content Analysis Task.

References

[1] P. Vandewalle, J. Kovacevic, and M. Vetterli, "Reproducible research in signal processing," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 37–47, May 2009.

[2] D. L. Donoho, A. Maleki, I. U. Rahman, M. Shahram, and V. Stodden, "Reproducible Research in Computational Harmonic Analysis," *Computing in Science & Engineering*, vol. 11, no. 1, pp. 8–18, Jan. 2009.

[3] S. Fomel and J. F. Claerbout, "Reproducible Research," *Computing in Science & Engineering*, vol. 11, no. 1, pp. 5–7, 2009.

[4] R. De Veaux and et al., "Curriculum guidelines for undergraduate programs in data science," *Annual Review of Statistics, forthcoming*, 2016.

[5] "ICIP Reproducibility Round Table," in *http://www.icip2011.org/index.php/Main/RoundTable*, 2011.

[6] S. Sonnenburg, M. L. Braun, and et al., "The need for open source software in machine learning," 2007.

[7] R. D. Peng and S. P. Eckel, "Distributed Reproducible Research Using Cached Computations," *Computing in Science & Engineering*, vol. 11, no. 1, 2009.

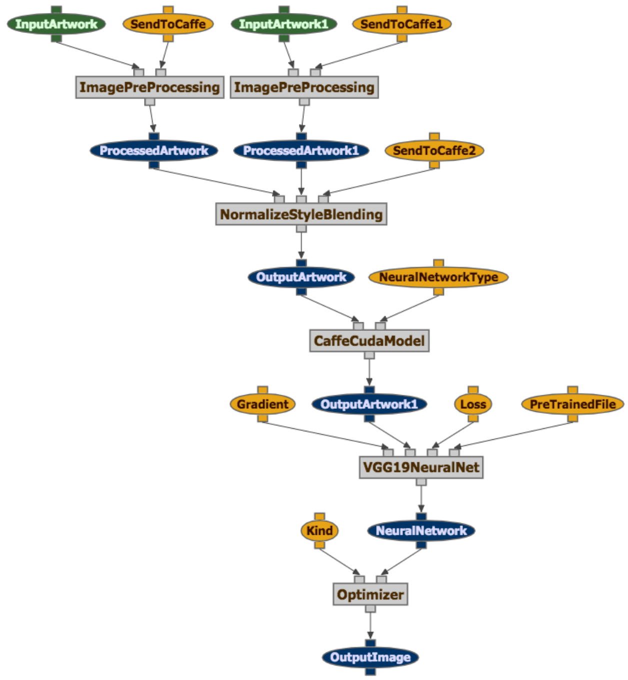


Figure 4. Workflow for the Neural Algorithm of Artistic Style using Lua and Torch.

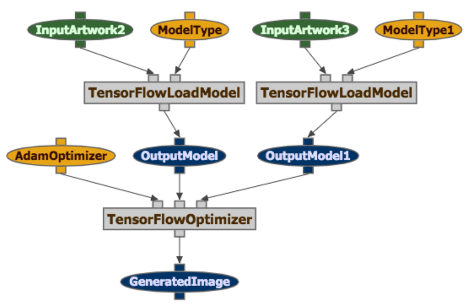


Figure 5. Workflow for the Neural Algorithm of Artistic Style using Python and Google's TensorFlow.

Figure 6. Reproducing the results from [23] using the workflows in Figure 4.

[8] R. LeVeque, "Python tools for reproducible research on hyperbolic problems," *Computing in Science & Engineering*, pp. 19–27, 2009.

[9] J. Buckheit, "Wavelab and reproducible research," *LECTURE NOTES IN STATISTICS*, 1995.

[10] I. J. Taylor, E. Deelman, D. B. Gannon, and M. Shields, *Workflows for e-Science: Scientific Workflows for Grids*, I. J. Taylor, E. Deelman, D. B. Gannon, and M. Shields, Eds. Springer-Verlag, 2006.

[11] V. Stodden, M. McNutt, D. H. Bailey, E. Deelman, Y. Gil, B. Hanson, M. A. Heroux, J. P. Ioannidis, and M. Taufer, "The reproducibility enhancement principles for disclosing computational methods," *Science, forthcoming*, 2016.

[12] Y. Gil, V. Ratnakar, J. Kim, P. A. Gonzalez-Calero, P. Groth, J. Moody, and E. Deelman, "Wings: Intelligent workflow-based design of computational experiments," *IEEE Intelligent Systems*, vol. 26, no. 1, 2011.

[13] Y. Gil, P. A. Gonzalez-Calero, J. Kim, J. Moody, and V. Ratnakar, "A semantic framework for automatic generation of computational workflows using distributed data and component catalogs," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 23, no. 4, 2011.

[14] Y. Gil and S. Pierce, "Final report of the 2015 nsf workshop on intelligent systems for geosciences," 2015, national Science Foundation, Arlington, VA.

[15] R. J. Sethi, C. A. Buell, W. P. Seeley, and Y. Gil, "Democratizing the visual stylometry of art: Analysis of artistic style through computational workflows," *Communications of the ACM, forthcoming*, 2016.

[16] Y. Gil, "Teaching big data analytics skills with intelligent workflow systems," in *Proceedings of the Sixth Symposium on Educational Advances in Artificial Intelligence (EAAI), colocated with the National Conference of the Association for the Advancement of Artificial Intelligence (AAAI)*, Phoenix, AZ, 2016.

[17] D. Garijo, Y. Gil, and O. Corcho, "Abstracting, linking, publishing, exploiting: Workflow sharing and reuse as linked open data," *Future Generation Computer Systems, forthcoming*, 2016.

[18] C. L. Zheng, V. Ratnakar, Y. Gil, and S. K. McWeeney, "Use of semantic workflows to enhance transparency and reproducibility in clinical omics," *Genome Medicine*, vol. 7, no. 73, 2015.

[19] B. Song, R. J. Sethi, and A. K. Roy-Chowdhury, "Robust Wide Area Tracking in Single and Multiple Views," in *Visual Analysis of Humans*, T. B. Moeslund, L. Sigal, V. Krüger, and A. Hilton, Eds. Springer-Verlag, 2011, pp. 1–18.

[20] R. J. Sethi, "Towards defining groups and crowds in video using the atomic group actions dataset," in *IEEE International Conference on Image Processing (ICIP)*, 2015.

[21] R. J. Sethi, Y. Gil, H. Jo, and A. Philpot, "Large-scale multimedia content analysis using scientific workflows," in *ACM International Conference on Multimedia (ACM MM)*, 2013.

[22] M. Hauder, Y. Gil, and Y. Liu, "A framework for efficient text analytics through automatic configuration and customization of scientific workflows," in *In Proceedings of the Seventh IEEE International Conference on e-Science*, Stockholm, Sweden, 2011.

[23] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," Aug 2015.