SET Brazil

SMPTE

SINCE 1916

SMPTE Motion Imaging Journal

Covering emerging technologies in film, broadcast, and the connected media ecosystem

Automating Metadata Logging Through Artificial Intelligence

By Christopher Witmayer

Abstract

In 2007, NASCAR designed and implemented a media asset management solution for their sport. Now, in 2018, the library has grown to more than 500,000 hr of content containing video, audio, and still images dating back to 1933—one of the most vast sports libraries in the world. Since the inception of the library, NASCAR has employed a staff to manually apply metadata to each video frame, amassing 10 million entries. While the logging by our staff has been impressive, the incoming data avalanche cannot be addressed by using the current system, let alone the glacier of data hiding in the archive. At present, we have calculated that it would take our existing staff nearly 150 years to log all of the historical content as it stands today. Over the past ten years, we have extensively analyzed open-source and proprietary tools aimed at dealing with the data logging gap and have determined that machine learning is the ideal solution to address metadata logging on large-scale media libraries. As machine learning has become more accessible through the scalability of cloud computing, training data, and implementing convolutional neural networks, it is now within the reach of media production companies and asset stakeholders. NASCAR is on the verge of revolutionizing how all data asset management systems can be restructured in the future to integrate machine learning to harness efficiencies in metadata logging.

NASCAR is on the verge of revolutionizing how all data asset management systems can be restructured in the future to integrate machine learning to harness efficiencies in metadata logging.

Keywords

Artificial intelligence (AI), convolutional neural networks (CNNs), machine learning, metadata tagging, tensorflow, transfer learning, video logging

Introduction

In our endeavor, we will seek to replace humans or enhance their abilities to apply visual metadata to video and image assets through machine learning. Given the recent growth and availability of cloud computing and machine learning tools, we believe we can prove that computers are capable of applying metadata to video assets on a mass scale.

It is our goal to log every frame of video and audio from our ever-expanding media library. Designed and built in 2007, the library currently holds more than 500,000 hr of video content and continues to grow at a rate of 1,200 hr/month. The library has employed between ten and 20 people at any one time, tasked with adding metadata to the media files. Currently, our logging staff is using desktop software—visually identifying the start and end point of a clip, and adding relevant metadata drawn from a defined series of dropdown menus. A given clip contain a minimum of ten spatial metadata tags (location, camera angle, etc.). Clips can receive additional visual metadata tags such as the driver, car number, sponsor, and so on. Since the inception of the library our logging team has manually added more than 10 million metadata points. Although this number is certainly impressive, it only accounts for less than 80,000 hr of content, which is a mere 16% of the library. Given that we have roughly 420,000 hr of video content still to log and that the library grows daily, the task ahead seems daunting and, frankly, unachievable without vast investments of time and money.

The NASCAR Library undoubtedly contains iconic images that enhance the stories of the drivers, the machines, and the passion of our sport. Perhaps, one day, we will find a story we did not know existed in our archives. One cannot know the value of his archive if he does not know what it contains. This black box of our library led us to explore machine learning as a solution.

Understanding Artificial Intelligence

To further comprehend how we can utilize artificial intelligence (AI) for logging our media assets, we must first understand the technology scope and limitations. The term *artificial intelligence (AI)* was coined by John McCarthy in 1956 when he gathered fellow scholars for

Digital Object Identifier 10.5594/JMI.2019.2932163
Date of publication: 23 September 2019

the Dartmouth Summer Research Project on Artificial Intelligence. The goal of the conference was "to proceed on the basis of the conjecture that every aspect of learning or any feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."[1] AI can be divided into two categories: General and Narrow.[i] At the highest level, General AI would state that the computer system is equivalent to that of human intelligence in that it can understand language, objects, planning and generally everything a human brain can comprehend. We are focusing on the second area—Narrow AI—where our machine will have a limited scope of understanding and only know how to identify objects that we have taught it to recognize. Objects and images outside of our trained scope will simply be ignored—hence our narrow intelligence. It should be noted that at this point there have been no true implementations of General AI. Even the most seemingly complicated systems, such as Google Translate, are only operating within the scope of Narrow AI.

Machine Learning

A further subset or implementation of Narrow AI is machine learning. The concept and term *machine learning* was coined by Arthur Samuel in 1959, who stated that machine learning is a "field of study that gives computers the ability to learn without being explicitly programmed."[2] It is within Samuels's statement that we garner the true value of machine learning. We, as end users, are merely defining the parameters. In our case, we are identifying a logo or car number, and the machine eventually learns a pattern that it can apply to detections. The more images that the computer analyzes and understands, the better the accuracy on future detections—the machine is now learning. It is within machine learning where we will focus on teaching our computer how to detect objects that are important to NASCAR, or any other organization.

[i]There is the newer concept of Artificial Super Intelligence stating that computers will be smarter than the average human brain, though this is conceptual and beyond the scope of this whitepaper.

Convolutional Neural Networks

With a broad understanding of AI and machine learning, we must next explore how a computer can recognize images similar to the relationship between the eye and brain. The human brain is complex, comprising more than 100 billion neurons.[3] These neurons layer upon each other to create a deep network controlling our everyday lives from basic human function, to thinking, and of course vision. In 1962, D. H. Hubel and T. N. Wiesel published their landmark paper documenting how the visual cortex works within mammals—more specifically, within cats. Through their research with cats, they were able to understand how light received by the visual cortex activated different neurons in the brain. Specific neurons responded based on the orientation of the light that was received. The scientists noted that as more cells were activated along the visual processing pathway, the object would become more clearly identified by the cat, thus it was able to recognize objects. It was this biological process that inspired the convolutional neural networks (CNNs) of today.[4]

Further studies would reveal that visual cortex neurons overlap, allowing multiple interpretations of the same region of vision. This convolving of responses from each neuron allows for more accurate identification should any one neuron not respond correctly or have a different interpretation.

Through understanding these layered features of how the visual cortex operates and applying them to machine learning, the concept of the CNN has emerged. Image data is input into the CNN where it is divided into smaller regions to be analyzed. The responses from each virtual neuron help to build out a digital picture, allowing it to be compared with previously known models.

The tipping point for machine learning, as it pertains to image detection, came in 2012 with the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The event asked competitors to use machine learning to identify objects within images based on a prebuilt set of data. While the scope of the objects within the images was limited, the error rate of detection was 16%—a significant reduction from the 2010 inaugural event results of 25%. Through the use of CNNs, the error

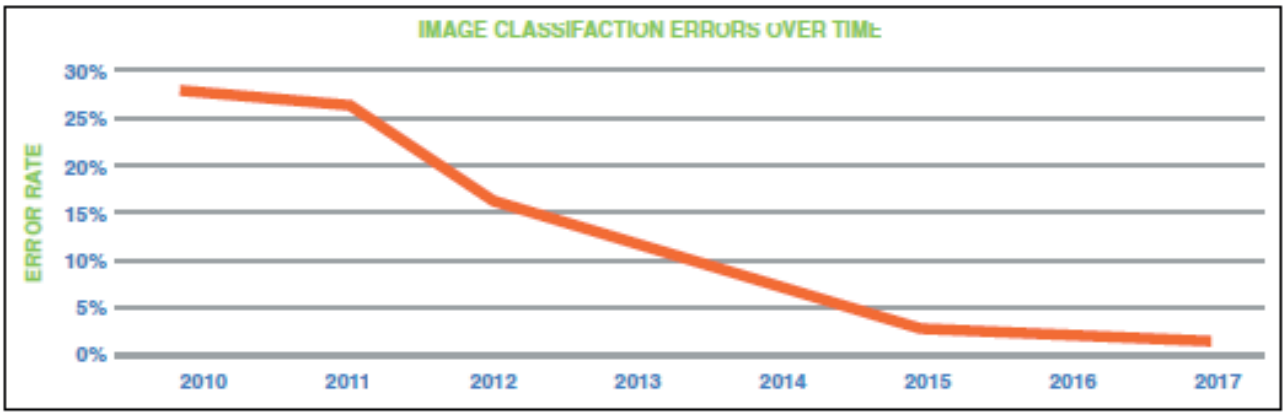

FIGURE 1. Image classification error rate from 2010 to 2017.

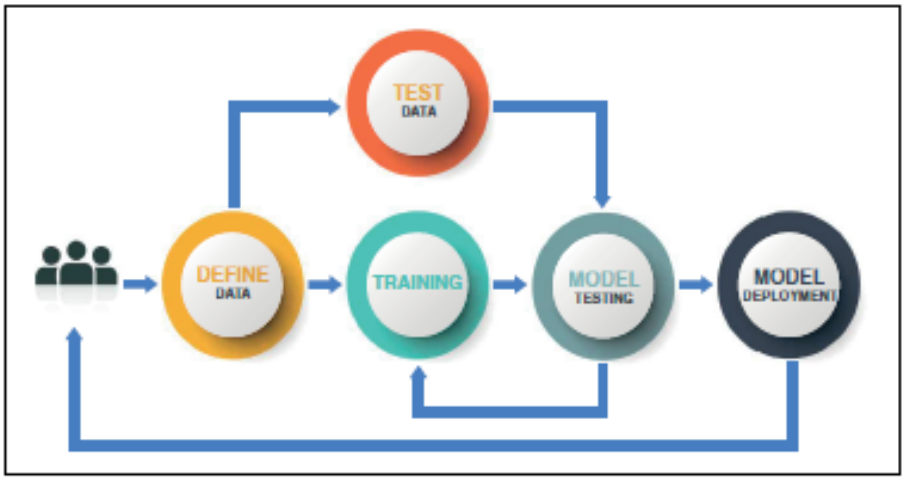

FIGURE 2. The path to machine learning.

rate continued to decline and, in 2015, achieved historic results by dropping below 5%—which is the error rate of humans identifying the same objects (**Fig. 1**).[5]

Decades of research in machine learning algorithms and the advancement of computational power have converged to allow for image recognition to be successful. It is this research that we will be building upon to detect objects important to NASCAR.

Machine Learning Tools

Over the past few years, there has been a surge of new tools and platforms made available for machine learning, both locally and in the cloud. During our exploration, we concluded that any toolset we selected should meet the following minimum requirements:

- Open Source—nonproprietary tool and vendor agnostic
- Portable—able to move from local to cloud with minimal issues
- Common Language Support—widespread user-based, such as Python
- Scalable—support for various hardware technologies such as CPU and GPU.

While we tested many tools and frameworks, we found that TensorFlow met all of our criteria and provided a good starting point.[6]

The Path to Machine Learning

Although there are many intricacies and variations in how to solve problems with machine learning, we will offer a template for general understanding and a path for success with a focus on integrations involving images and video, also known as Classification. The process of designing and building solutions for machine learning for image recognition will be relatively similar, regardless of dissimilar content, chosen framework, or deployment (**Fig. 2**). At the end of the day, we are simply teaching a computer to recognize patterns and make a prediction that the image contains something it has seen before.

Define the Data

The first and most critical step in machine learning is defining the data set. There are two types of data—Generic and Custom. A Generic data set (or model) could include general objects without specific detail—such as a tree, but not a type of tree. Generic image data sets can often be found on the internet through projects such as Open Images data set,[ii] which contains roughly 9 million images. Generic models are often offered as a pretrained model through cloud providers and are available through an application programming interface (API). In our use case, we would like to identify objects that are specific to NASCAR, and therefore, we build Custom Data Models. Because we are the rights holder for NASCAR content, we have a vast library of images which we can utilize for building our models. In our use case, we have selected roughly 500 images of the iconic #43 car for our training data. The images selected do not have to be pristine—in fact, the more variations, the better, as this is how the object appear in real life (**Fig. 3**).

Although models can be built on any number of images, the lack of data will affect the accuracy of the model. However, when building models, the law of diminishing returns does take effect. Earlier analyzed images will offer more value than those analyzed later. Depending on the variations of the images in a data set, the point of diminishing returns will vary.

[ii]Open Images Data set can be found here: https://storage.googleapis.com/openimages/web/index.html. Other data sets are available, however be sure to verify the licensing agreements.

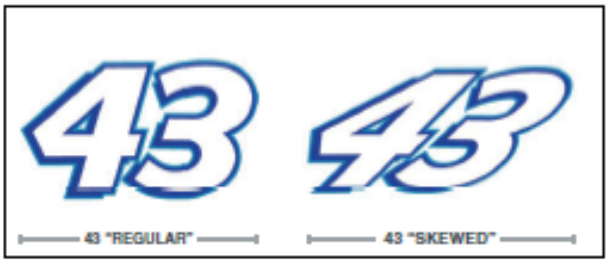

FIGURE 3. Images should vary in size, shape, and skew.

FIGURE 4. Richard Petty and the icon #43 car. Note the green bounding box around the #43.

Train the Model

The next step in the process of machine learning is training the computer to recognize a specific object and generate a resulting model. There are two main types of training—supervised and unsupervised. In unsupervised training, the machine is expected to find patterns leading to identifying an object, which will eventually be labeled and used for future detection. Unsupervised learning takes significantly more data and time and is not ideal for image detection as it pertains to our usage in tagging assets in media libraries. For our use case, we use Supervised Training as it allows us to use our domain-specific knowledge to identify objects. Again, the process for training may differ based on the tools selected; however, the concepts remain the same. The first step in training is to identify the target/object in the image that the computer should learn. While all of the other items in the frame be processed, the patterns within the defined location be used specifically for training. In the process, we create a bounding box of the area and apply a label to what is within the area, with the label being the title for the object (**Fig. 4**). Behind the scenes, the software is defining a series of coordinates for the bounding box and will save them to an Extensible Markup Language (XML). This process is completed for each image. In our example, we would like to identify the car number 43, so our label is "43."

There are many tools available for annotating, both locally and in the cloud, from open source to crowd-sourced. For our annotation, we opted to use the open-source tool, LabelImg, as it allowed our existing logging staff to identify objects using local infrastructure.[7]

Model training can be time and computationally intensive, taking days or weeks to train a single model. In an effort to better understand the process and invest a limited amount of resources into machine learning research, we opted to utilize a concept known as transfer learning. Transfer learning allowed us to take a publicly available model and retrain it with our images, essentially transferring all of the previously learned image features. Because we are focusing on TensorFlow, we were able to utilize the Inception Version 3 module which is a neural network that has been trained on the ISLVCR data set (ImageNet).[8] This process allowed us to train our models in a few hours on a standard desktop without any additional hardware.

Test the Model

Prior to deployment, it is critical to test the model and confirm that it can indeed recognize the intended objects with a high level of accuracy. The remaining images from our initial pool of data were used for this testing. In addition, we made certain not to use any images that were part of the initial training process, as the model would be able to detect them with pinpoint precision since it had previous knowledge.

During the testing process, we look for the optimal fit to our model. A fit model can be defined as one that is able to accurately detect objects within the image. Models can be overfit, where they match only exact examples of the object it was trained against. Conversely, a model can also be underfit and not have enough data to make an accurate detection. If the optimal fit is not achieved, more data, or perhaps more diverse data, may be required to help round out the model and identify the object correctly. At the completion of our testing, we were able to move our model to the deployment stage.

Deployment

There are three potential machine learning deployment strategies: local, Cloud Infrastructure as a Service (IaaS), and Cloud Platform as a Service (PaaS). Each of these deployment methodologies has its own unique requirements and cost considerations.

Local Deployment

Although we certainly highlight the advantages of cloud computing, there is still a value in local deployment of trained models. A local deployment would consist of either a single computer or cluster of machines in the company datacenter. Local deployments allow training and testing of models without incurring additional operational or capital costs. In our initial example, we used local machines to build and test our models allowing us to see the potential for success.[iii]

IaaS Cloud Deployment

In the IaaS deployment model, we can utilize cloud-based servers with additional resources such as GPUs and custom machine learning hardware such as the tensor processing units (TPUs).[iv] Migrating to IaaS was the next logical progression for our testing, allowing for better performance, though it added complexity in management of the servers.

[iii] It should be noted that in our examples we utilized transfer learning, and thus did not have the same computational requirements that may be needed for fully training a model from scratch.

[iv] TensorFlow processing units are currently only available on Google Cloud Platform (GCP).

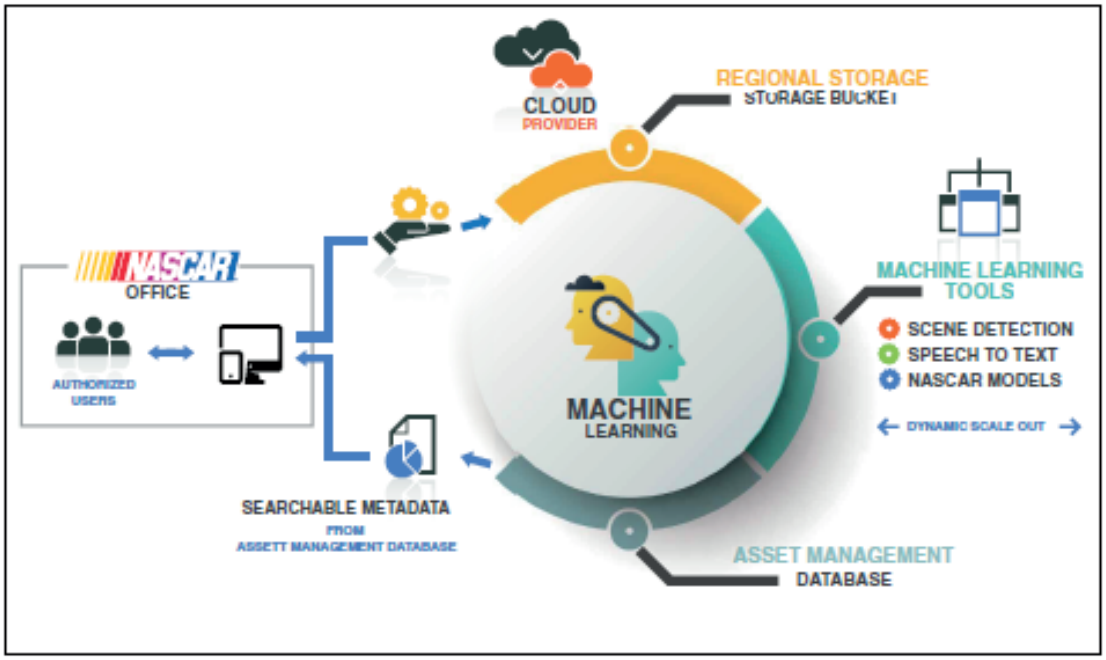

FIGURE 5. Final diagram of our workflow integrating custom models as well as generic cloud APIs.

PaaS Cloud Deployment

Recently, cloud providers have been offering machine learning platforms in which we can bring our own custom models. The PaaS deployment allows us to focus on the model and data, abstracting the hardware and scaling process.

While we have tested all three deployment models, we currently are operating locally as we are still expanding our model development. As we move our models to production and begin processing more data, the PaaS will allow us to scale dynamically with our limited resources.

Delivery

The final step in our machine learning workflow is executing the models and delivering the metadata into our media asset management system (**Fig. 5**). Our workflow for this project was to analyze individual assets as they are restored from the tape archive to spinning disk. As the model is run against the video file, the identified metadata is delivered via API to our asset management database.

At the start of this project, we set out to use machine learning to replicate or perhaps improve the existing metadata tagging process. During the process, we discovered that, while we are now able to tag every frame of video, a question arises as to whether or not tagging every frame is necessary as the human counterpart tags groups of frames or clips. In addition, we discovered that not only were we able to offer frame-accurate identification for archival assets, we were also able to apply metadata to live assets with minimal delay.

Rinse and Repeat

While machine learning can be highly accurate, there should be acknowledgement that it can deliver incorrect identification. These inaccurate predictions are called *false positives* and *false negatives* (**Fig. 6**). A false positive, also known as a type 1 error, occurs when the label is incorrectly matched to an object, thereby showing that an object exists in an image even though it does not. In our example, we are showing that the label incorrectly tagged a NASCAR driver. A false negative, or a type 2 error, is when the object is not identified or more correctly identified and omitted as the correct object. In our example, the NASCAR driver is acknowledged, but it is not labeled.

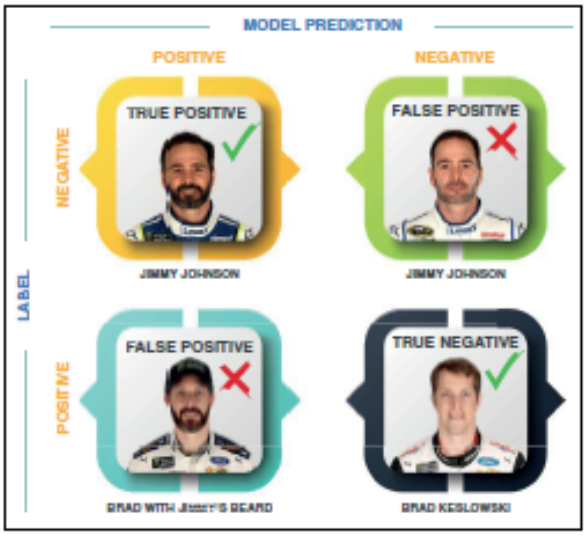

FIGURE 6. Example diagram showing four possible outcomes of prediction.

In instances where false positives or false negatives are encountered, the model should be retrained with additional images including the ones that failed in the most recent prediction. Because the model is already built, the model will only be learning from the new data and therefor the time commitment will be limited. Aside from addressing issues with false positives and false negatives, one should expect that retraining of models will be required as the subject matter evolves.

Conclusion

Throughout the past ten years, we have been tracking the efficiency of our logging staff to better understand and set expectations regarding how much visual metadata can be applied in a given day. A seasoned logger is about 25% efficient on their best day. That being the case, during an 8-hr shift, a logger can apply visual metadata to approximately 2 hr of content. In our smaller scale testing, we deployed our models to the same desktop that a logger uses and were able to analyze the 2 hr of footage in 51 min. During our testing, we opted to analyze every frame of video, which is well beyond what a human is capable of completing.

In addition, our machine learning models were able to detect objects with an accuracy of 97%, which is on par with the ImageNet expectations (mentioning previously). This accuracy would, therefore, put our models within the same, if not better, accuracy of our logging staff.

While these initial results seem promising, there are inherent limitations, or limitations as defined by the scope of our project. In our current use case, we are using machine learning to identify logos, car numbers, and key personalities.

The first issue arises with the accuracy of the model as it pertains to how it is visually skewed in the overall image. While we trained against a fairly large and diverse set of content, we did not offer images that accurately represented all angles in which a logo may appear on screen. We suspect that with additional training, we could teach the model to recognize the object, though we question if the additional data is needed on such a visually skewed object.

Next, we look at the key personalities within NASCAR. Again, we have a 95% accuracy in detection; however, when a key personality such as a driver is wearing a helmet, we lose the ability to see their face and therefore can no longer identify or track them. While we could add models to track helmets, we would have no way of acknowledging that the driver is actually wearing their own helmet and would, in turn, receive a false-positive response.

Machine learning, at this point, is not without limitations. While we see a great step forward in the potential for adding metadata to our vast archive, we are considering machine learning as a supplement to enhance the productivity of our human logging team. As our models continue to advance, so will our deployment and reliance on machine learning.

Acknowledgments

I would like to thank William Ihle and Christopher Wolford of the NASCAR Productions team who aided in testing. In addition, I would also like to thank Christian Ferrer-Garcia for code assistance, Joe Walker for editorial support, Kurt Jenkins for graphics, and the management team of NASCAR for their support.

References

1. J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence," Aug. 1955. [Online]. Available: http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html
2. A. L. Samuel, "Some Studies in Machine Learning Using The Game of Checkers," *IBM J. Res. Develop.*, *44*:206–226, 1959.
3. S. Herculano-Houzel, "The Human Brain in Number: A Linearly Scaled-Up Primate Brain," *Front. Hum. Neurosci.*, vol. 3, p. 1, Nov. 2009. [Online]. Available: https://ncbi.nlm.nih.gov/pmc/articles/PMC2776484
4. D. H. Hubel and T. N. Wiesel, "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex," *The Physiological Society*, Jan. 1962.
5. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and F. Li, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, 115(3):211–252, Dec. 2015.
6. Google, "Tensorflow GitHub." [Online]. Available: https://github.com/tensorflow/tensorflow
7. Tzutalin, "Github Repository—LabelImg," 2015. [Online]. Available: https://github.com/tzutalin/labelImg
8. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," Dec. 2015. [Online]. Available: https://arxiv.org/abs/1512.00567

About the Author

Christopher Witmayer is the director of broadcast, production, and new media technology for NASCAR, Charlotte, NC. Throughout his 17 years in media and entertainment, he has focused on streamlining/transforming content management and delivery solutions through new technology and workflow improvements. Prior to his current position, Witmayer served as the senior manager of new media technology for NASCAR, where he transitioned NASCAR Productions to NASCAR Plaza and Hall of Fame. He also designed the facility's technological infrastructure for production, editorial, and archival operations. Witmayer is a renowned speaker having delivered presentations to the Sports Video Group (SVG), National Association of Broadcasters (NAB), International Broadcast Convention (IBC), Microsoft Build, Microsoft Ignite, SMPTE, and The United States Library of Congress. His personal approach to his presentations brings simplicity to complex topics allowing everyone to learn and contribute to the conversation.

Presented at the SMPTE 2018 Annual Technical Conference & Exhibition, Los Angeles, CA, 22–25 October 2018.