recovery operations and 8% of blade-swapping procedures failed, which led to the failure of 20% of the active applications during these recovery epochs. A successful recovery does not guarantee to protect the application and system but only 0.02% of the active applications failed during the successful recovery period.

- Using real attack data and associated alerts as drivers, we have developed and evaluated *AttackTagger*, an adaptive learning-based IDS. The approach is based on probabilistic graphical models—specifically factor graphs—which integrates security alerts from multiple sources for accurate and preemptive detection. The method was validated using real data from attacks at NCSA.

WHY BLUE WATERS

Blue Waters is one of the few systems that can scale computations to tens or hundreds of thousands of cores on CPUs and GPUs. It also enables the study of failures in production petascale systems with its unique mix of XE6 and XK7 nodes. This capacity allows us to understand the performance–fault-tolerance continuum in HPC systems by enabling the investigation of application-level designs for mixed CPU and GPU node systems, and fault isolation in system components to mitigate failures at the application level. This allows us to design high-performance and resilient genomics pipelines that can make use of HPC systems.

NEXT GENERATION WORK

We ascertained the performance pathologies of several common kernels used in popular computational-genomics tools and built a scheduling algorithm that is able to dynamically decide task placement in a heterogeneous cluster. Also, we analyzed log data produced by Blue Waters to discover and quantify new failure modes that could not be efficiently handled by system-level recovery mechanisms. Our future work will bring together these observations to build a holistic runtime system that will jointly reason about performance and resiliency for coordinated placement and checkpointing decisions.

PUBLICATIONS AND DATA SETS

Banerjee, S. S., et al., Efficient and Scalable Workflows for Genomic Analysis. Proceedings of the *ACM International Workshop on Data-Intensive Distributed Computing (DIDC '16)* (ACM, New York, New York, June 1, 2016), pp. 27–36.

Jha, S., et al., Understanding Gemini Interconnect Failovers on Blue Waters. *Proceedings of the Cray User's Group (CUG)*, London, England, May 8–12, 2016.

Banerjee, S. S., et al., Sequences to Systems. Coordinated Science Laboratory Technical Report UILU-ENG-16-2201.

Di Martino, C., et al., LogDiver: A Tool for Measuring Resilience of Extreme-Scale Systems and Applications. Proceedings of the *5th Workshop on Fault Tolerance for HPC at eXtreme Scale (FTXS '15)* (ACM, New York, New York, June 15–19, 2015), pp. 11–18.

Cao, P., E. C. Badger, Z. T. Kalbarczyk, and R. K. Iyer, A Framework for Generation, Replay, and Analysis of Real-World Attack Variants. Proceedings of the *Symposium and Bootcamp on the Science of Security (HotSoS 2016)* (ACM, New York, New York, April 19–21, 2016), pp. 28–37.

LARGE-SCALE LEARNING FOR VIDEO UNDERSTANDING

Allocation: GLCPC/377 Knh
PI: Jia Deng[1]
Co-PI: Jason Corso[1]

[1]University of Michigan

EXECUTIVE SUMMARY

Video understanding, endowing computers with the ability to interpret videos as humans do, is one of the fundamental challenges of computer vision and artificial intelligence. Video data is ubiquitous, but our ability to perform automated analysis on such data is still primitive. In this project, we use Blue Waters to advance the research in video understanding, including recognizing human actions and activities, extracting high-level semantics from instructional videos, and accelerating deep neural network computation.

INTRODUCTION

Video understanding, endowing computers with the ability to interpret videos as humans do, is one of the fundamental challenges of computer vision and artificial intelligence. Video data is ubiquitous and is projected to account for 79% of all consumer Internet traffic in 2018 [1]. Yet our ability to perform automated analysis on such data is still primitive. We use Blue Waters to advance the research in video understanding.

One important problem is to understand human actions and activities. That is, given a visual input such as a video frame, generate a list of human action categories and their locations, for example, predicting that there is a person riding a horse at the lower left region of the given video frame. Automated recognition of human actions is key to the success of many important applications, such as human-computer interaction, robotics, and smart healthcare systems.

Another problem is to understand the high-level semantics of instructional videos with a focus on cooking activities. For a given cooking video, such as making a peanut butter and jelly sandwich, we seek to learn a cross-modal model of the temporal structure and constraints of the cooking process from both visual and audio content. The learned model will be able to generate a visual-textual summary of the process and will serve as a natural index for search and query across many such processes.

A third problem is in the area of robotic localization and mapping, which have recently become important in the context of autonomous driving. With the availability of more computing power, techniques like deep reinforcement learning [2] have become viable in many practical problems, including autonomous driving. An interesting question is how to improve deep reinforcement learning algorithms using games and simulations. We are interested in evaluating this technique on real-world problems such as learning the concept of objects and learning simple physics rules.

A fourth problem is how to make video understanding algorithms efficient. One aspect is how to accelerate the computation of deep neural networks (DNN) [3, 4], which are widely used for many subtasks for video understanding.

METHODS & RESULTS

Our main approach is machine learning. For the problem of understanding human actions and activities, we investigated deep neural networks (DNN) [3, 4]. The recent development of DNNs has led to large improvements for object recognition [5]. But compared to objects, human actions are far more complex. We have found that naively applying DNN-based object recognition algorithms for human actions does not perform well. We thus investigated novel DNN architectures for recognizing human actions. We have developed a novel multi-stream architecture that can integrate cues from humans, objects, and scene context. Our approach has achieved **state-of-the-art performance** on a large-scale action recognition benchmark [6].

For the problem of extracting semantics from instructional videos, we collected a large cooking video dataset, YouCook2, by querying YouTube. The videos were then annotated with sentence descriptions, as well as timestamps denoting the particular cooking styles, such as grilling and frying. As a first step, we generated an intermediate video representation by grouping similar pixels in space and time in a video using our streaming hierarchical video segmentation method [7]. We further extracted the semantic entities, such as objects and actions, from videos using random field modeling [8]. Our methods achieve much better results than the previous state of the art in the field.

For the problem of deep reinforcement learning, we have set up a simulation in the robot simulator Gazebo with wooden blocks and a robotic arm with two degrees of freedom. For the problem of accelerating DNN computation, we investigated custom data representations and ran simulations on how they can be used to speed up DNN computation through new hardware design.

WHY BLUE WATERS

Running DNNs and processing video require intensive parallel computation on both CPUs and GPUs. In order to perform thorough experiments, we need access to a large number of CPUs and GPUs. This makes Blue Waters **essential** for our research. At the peak of our usage, we were able to use a large number of CPUs and GPUs concurrently, which allowed us to process a large amount of video and explore a large design space of models. In addition, we received timely help from the Blue Waters team, which was also **indispensable** for our project.

NEXT GENERATION WORK

We hope to use Blue Waters to continue advancing video understanding, including more accurate understanding of human actions, deeper semantics from instructional videos, and faster DNN computation.

ALGORITHMS FOR EXTREME SCALE SYSTEMS

Allocation: Illinois/100 Knh
PI: William Gropp[1]
Collaborator: Luke Olson[1]

[1]University of Illinois at Urbana-Champaign

EXECUTIVE SUMMARY

Continued performance enhancement of large-scale computer systems will come from greater parallelism at all levels. At the node level, this is seen in the increasing number of cores per processor and the use of large numbers of simpler computing elements in general purpose graphics processing units (GPGPUs). The largest systems must network tens of thousands of nodes together to achieve the performance required for the most challenging computations. Successful use of these systems requires new algorithms. Over the last year, we have shown the benefit of lightweight intranode balancing on scalability and performance. We continue to explore alternative formulations of conjugate gradient that eliminate some of the strict barrier synchronization and better use memory hierarchy, ways to reduce the impact of communication on the scalability of algebraic multigrid, and algorithmic approaches to resilience that exploit the multilevel representation in multigrid methodology.

INTRODUCTION

At extreme scale, even small inefficiencies can cascade to limit the overall efficiency of an application. New algorithms and programming approaches are needed to address barriers to performance. This work directly targets current barriers for effective use of extreme scale systems by applications. For example, Krylov methods such as conjugate gradient are used in many applications currently being run on Blue Waters (MILC code is one well-known example). Developing and demonstrating a more scalable version of this algorithm would immediately benefit those applications. In the longer term, the techniques that are developed will provide guidance for the development of highly scalable applications.

METHODS & RESULTS

Early results with alternative Krylov formulations have revealed several performance effects that can provide a factor of two or greater improvement in performance at scale. Current work has been limited by the fact that the non-blocking MPI_Allreduce on Blue Waters is functional but does not provide the expected (or perhaps hoped for) performance, particularly regarding the ability to overlap the Allreduce operation with other communication and computation. However, even with this limitation, we have seen a benefit in using non-blocking collective operations regarding a reduction in the sensitivity of the application to performance jitter and other irregularities.

WHY BLUE WATERS

Scalability research relies on the ability to run experiments at large scale, requiring tens of thousands of nodes and hundreds of thousands of processes and cores. Blue Waters provides one of the few available environments where such large-scale experiments can be run. In addition, only Blue Waters provides a highly capable I/O system, which we will use in developing improved approaches to extreme-scale I/O.

NEXT GENERATION WORK

We expect the next generation systems to rely on many more cores per node and to use different network topologies compared to Blue Waters. Additionally, there are likely opportunities to exploit new network capabilities and provide algorithms and programming systems adapted to the new memory, processor, and interconnect architectures.

PUBLICATIONS AND DATA SETS

Kale, V., Low-overhead scheduling for improving performance of scientific applications, Ph.D. Thesis, Spring 2015.

Eller, P., and W. Gropp, Exploiting nonblocking collective operations in conjugate gradient, *Copper Mountain Meeting,* 2015.

Bienz, A., Analyzing the performance of a sparse matrix vector multiply for extreme scale computers, *SC15*, Austin, TX, Nov. 2015.

Eller, P., Non-blocking preconditioned conjugate gradient methods for extreme-scale computing, *SC15*, Austin, TX, Nov. 2015.