TOWARDS UNDERSTANDING INTUITIVE PHYSICS WITH LANGUAGE AND VISION

DİL VE GÖRMEYİ KULLANARAK SEZGİSEL FİZİĞİ ANLAMAYA ÇALIŞMAK

TAYFUN ATEŞ

ASSOC. PROF. DR. MEHMET ERKUT ERDEM

Supervisor

ASSOC. PROF. DR. İBRAHİM AYKUT ERDEM

Co-Supervisor

Submitted to

Graduate School of Science and Engineering of Hacettepe University

as a Partial Fulfillment to the Requirements

for the Award of the Degree of Master of Science

in Computer Engineering

January 2021

To my lovely boy. . .

ABSTRACT

TOWARDS UNDERSTANDING INTUITIVE PHYSICS WITH LANGUAGE AND VISION

Tayfun Ateş

Master of Science, Computer Engineering Department

Supervisor: Assoc. Prof. Dr. Mehmet Erkut Erdem

Co-Supervisor: Assoc. Prof. Dr. İbrahim Aykut Erdem

January 2021, 64 pages

Visual question answering (VQA) is one of the difficult tasks in multimodal machine reasoning. VQA requires machines to provide correct answers to questions about an image or a video. Here, the machine should perceive the scene and infer true judgements on the relationships between different entities. Recent benchmarks on VQA have been mostly proposed for static images and they only question spatial reasoning capabilities of artificial models. In other words, it is not a requirement for the machines to learn the physical properties of objects and understand different physical relationships among them. Hence, it is not possible to evaluate whether the models have intuitive physics or causal and temporal reasoning capabilities using these datasets. This thesis proposes a new benchmark, CRAFT, which is designed to evaluate these capabilities of artificial intelligence models. In particular, it comprises of 38K video and question pairs that are automatically generated from 3K videos of dynamic scenes. These scenes are synthetically created using a physics engine by considering ten different two-dimensional scene layouts containing variable number of dynamic objects. While generating the questions in CRAFT, we consider five different categories, two of those (*descriptive* and *counterfactual*) have been investigated in earlier works. However, in our work,

we have introduced three new question categories (*cause*, *enable*, and *prevent*) which are proposed inspired by the representations of causal relationships in cognitive science. A special attention has been given to data generation process to focus on creating questions which are easy to solve by humans, but difficult for machines. In order to support this claim, CRAFT questions are asked to both artificial models and 12 adult participants. Our experimental results demonstrate that although the tasks seem intuitive for human participants, there is a large gap between them and the most successful artificial model.

Keywords: Deep Learning, Computer Vision, Natural Language Processing, Cognitive Science, Visual Question Answering, Intuitive Physics, Causal and Temporal Reasoning

ÖZET

DİL VE GÖRMEYİ KULLANARAK SEZGİSEL FİZİĞİ ANLAMAYA ÇALIŞMAK

Tayfun Ates

Yüksek Lisans, Bilgisayar Mühendisliği

Danışman: Doç. Dr. Mehmet Erkut Erdem

Yardımcı Danışman: Doç. Dr. İbrahim Aykut Erdem

Ocak 2021, 64 sayfa

Çok kipli yapay muhakeme görevlerinin en zorlularından biri de görsel soru cevaplamadır. Bu problemde makinenin verilen bir görüntü ya da video hakkında sorulan soruya doğru cevap vermesi beklenmektedir. Soruya doğru cevap verebilmesi için, makinenin sahneyi iyi anlaması, sahnedeki varlıklar ve varlıklar arası ilişkiler hakkında doğru yargılara varması gerekmektedir. Görsel soru cevaplama üzerine çıkmış yapay veri kümeleri genellikle sabit görüntüler üzerinedir ve sadece modellerin uzamsal muhakeme yeteneklerini ölçmektedir. Bu tarz sabit sahnelerde ise makine, soruya doğru cevap vermek için sahnedeki varlıkların fiziksel özelliklerini öğrenmek zorunda değildir. Öte yandan, bu veri kümelerini kullanarak modellerin sezgisel fizik ya da zamansal ve nedensel muhakeme yeteneklerinin olup olmadığını ölçmek mümkün değildir. Bu tez kapsamında, bu yetenekleri de ölçebilmek adına; soruları ve görselleri, yapay ve otomatik yollarla elde edilmiş, CRAFT adında yeni bir veri kümesi oluşturulmuştur. CRAFT içindeki yaklaşık 38000 adet soru ve video çifti, yine yaklaşık olarak 3000 adet hareketli sahne videolarından oluşturulmuştur. Bu videolar, farklı sayıda hareketli varlık içeren on farklı iki-boyutlu sahne düzeninden sentetik bir biçimde oluşturulmuştur. CRAFT soruları hazırlanırken ise daha önce de çalışılmış iki adet

soru kategorisinin (betimsel ve *karşıolgusal*) yanında bilişsel bilimlerdeki nedensel ilişkilerin temsillerinden de ilham alınarak daha önce çalışılmamış yeni soru kategorileri (*sebebiyet*, *kolaylaştırma* ve *engelleme*) de eklenerek toplam beş adet soru kategorisi yaratılmıştır. Bu video ve soru çiftleri hazırlanırken özellikle insanlar için kolay ama makineler için zor olmasına dikkat edilmiştir. Bu iddiayı savunmak için de CRAFT soruları hem seçilmiş yapay modellere hem de 12 yetişkin katılımcıya sorulmuştur. Deneysel sonuçlarda ise videolarla ilgili soruların insanlar tarafından kolayca cevaplanabilmesi ve yapay modellerin benzer sezgisel fizik yeteneğine kolayca erişememesi gözlemlenmiştir.

Anahtar Kelimeler: Derin Öğrenme, Bilgisayarla Görme, Doğal Dil İşleme, Bilişsel Bilim, Görsel Soru Cevaplama, Sezgisel Fizik, Nedensel ve Zamansal Sorgulama

ACKNOWLEDGEMENTS

First and foremost, I would like to present my gratitude to my supervisors Assoc. Prof. Dr. Mehmet Erkut ERDEM and Assoc. Prof. Dr. İbrahim Aykut ERDEM for all the guidance and knowledge that they have provided throughout this degree.

I would like to thank Prof. Dr. Deniz YÜRET for providing valuable feedbacks especially regarding the dataset statistics emphasizing the importance of creating an unbiased dataset.

I also would like to thank Assoc. Prof. Dr. Tilbe GÖKSUN for pointing out the literature on force dynamics that we use constructing our dataset.

I would like to thank the remaining committee members, Prof. Dr. Pınar DUYGULU ŞAHİN and Assoc. Prof. Dr. Nazlı İKİZLER CİNBİŞ for reviewing my thesis and providing valuable comments.

I also would like to thank my colleagues Muhammed Şamil ATEŞOĞLU, Çağatay YİĞİT, İlker KESEN, and Mert KOBAŞ.

Moreover, I would like to thank my friends Dr. Savaş ÖZKAN, Deniz Can ÇALIŞKAN, Burak ERCAN, and Menekşe KUYU for providing answers to my series of exhaustive questions.

Furthermore, I deeply thank my parents, Ayhan and Fatma Nağlan, my big brother Tuğrul Kağan for their endless support for the completion of this thesis.

Lastly, I would like to thank Gamze ALTUN for her supports during my studies.

GENİŞLETİLMİŞ ÖZET

Son yıllarda yapay öğrenme alanındaki gelişmeler sayesinde bir görüntü içinde hangi varlıkların olduğunu [1–3], o varlıkları çevreleyen kutuların hangileri olduğunu [4–6] ya da bu varlıkların tam olarak sınırlarının hangi pikseller olduğunu söyleyebilen modellere sahibiz [7–9]. Daha da ötesinde, bir görüntü [10, 11] ya da video [12, 13] içindeki hareketleri sınıflandırabilen modeller de var. Çok kısa bir süredir, veriyi anlamanın yanında, verinin kendisini üretebilen modellerde de büyük gelişmeler oldu. Örneğin, daha önce hiç varolmamış insan yüzleri yaratabilen modeller geliştirildi [14]. Bu gelişmeler elbette bizi çok mutlu etse de, henüz yapay zekânın insan zekâsına yaklaşamadığı noktalar da mevcut. Bunlardan biri insanların sağduyuya dayalı akıl yürütme yeteneğidir (*commonsense reasoning*). Bu yeteneğimiz günlük hayatta karşılaştığımız olay ve durumlarla ilgili diğer insanlarla ortak bir şekilde vardığımız yargılardan oluşmaktadır. Örneğin, hepimiz bir yumurtayı biraz yüksekten bıraktığımızda, yere düşünce kırılıp dağılacağını tahmin edip kafamızda canlandırabiliyoruz. Sezgisel fizik ise bu yeteneğimizin bir alt yeteneği olarak düşünülebilir. Örnek vermek gerekirse, hacim sahibi iki varlığın birbirine çarpacağını gördüğümüzde, daha ağır olarak hayal ettiğimiz varlığın diğerine göre bu çarpışmadan daha az etkileneceğini gözümüzde canlandırabiliyoruz. İnsanoğlunun bu yeteneği, yapay öğrenme ve bilişsel bilim alanındaki araştırmacılara, yapay yollarla üretilmiş bir zekânın benzer kabiliyetleri olabilir mi sorusunu sorduruyor. Bu tezin amaçlarından biri de yapay zekâlara gözlemledikleri fiziksel olaylarla ilgili zamansal ve nedensel muhakeme yeteneği kazandırılmasına yardımcı olmaktır.

İnsanların gerçekleşmekte olan olaylarla ilgili zamansal ve nedensel muhakeme yetenekleri dışında, gerçekleşecek olan olaylarla ilgili karşıolgusal değerlendirme yetenekleri de bulunmaktadır. Bir sahnede gerçekleşecek olay ile ilgili sahnede basit bir değişikliğe gidilecek ise (yeni bir varlığın eklenmesi, bir varlığın çıkarılması, dış bir kuvvet uygulanması vs.), bu değişikliğin sonuçlarını olay gerçekleşmeden kafamızda resmedebilme ve yeni olası olaylarla ilgili yargıya varma yeteneğine sahibiz. Bahsedilmesi gereken önemli bir nokta

ise, robotların bu tarz fiziksel muhakeme yeteneklerini geliştirmek onları bulundakları fiziksel ortamda gerçekleştirdikleri hareketin sorumluluğunu almasında yardımcı olacaktır. Bir hareketi gerçekleştirmeden önce, olası sonuçlar ile ilgili tahminde bulunmaya başlayacaklardır. Yapay öğrenme alanındaki güncel çalışmalardan güzel bir örnek olarak Jenga oynayan robotu söyleyebiliriz [15].

Sezgisel fiziğin yapay yollarla öğrenilmesi yeni bir araştırma problemi olmasına rağmen, bazı problemler araştırmacılar tarafından çalışılmaya başlandı. Örneğin, bu çalışmalardan birinde yazarlar 3-boyutlu küplerden oluşan bir yığının sabit mi yoksa yıkılmak üzerine mi olduğunu anlamaya çalıştı [16]. Bir diğer çalışma ise eğer yıkılmak üzere ise bu küplerin nereye düşeceğini tahmin etmeye odaklandı [17]. Bazıları ise probleme tersten bakıp planlama algoritması kullanarak küplerden sabit bir yığın oluşturma üzerine çalıştılar [18]. Tek bir görüntüye bakarak bir ya da daha fazla kuvvete maruz kalacağı gözüken bir objenin hareketinin nasıl olacağını tahmin etmeye çalışan yapan modeller de mevcut [16]. Son olarak ise, temsil edilen fiziksel varlığın sadece katı bir varlık olmayabileceğini düşünen ve sıvıların temsilini yapay öğrenme ile sağlamaya çalışan yöntemler de mevcut [19]. Bunlar bütün çalışmaları kapsamasa bile, bu tez hakkında fikir vermesi açısından başarılı örnekler olarak sayılabilirler.

Bu tezin temel amacı; yapay öğrenme modellerinin bir sahnede yer alan hareketli varlıkları ve fiziksel ilişkileri anlamasına ve sahne ile ilgili muhakeme yeteneği kazanmasına yardımcı olmaktır. Bunu yapmak için, tez kapsamında literatüre CRAFT isminde yeni bir görsel soru cevaplama (visual question answering) veri kümesi sunulmuştur. Bu veri kümesi yaklaşık 3000 adet videodan oluşturulan yaklaşık 38000 adet soru-video çifti içermektedir. Videolar 10 farklı 2-boyutlu sahneden otomatik ve yapay yollarla (Box2D [20] motoru kullanılarak) üretilmiştir. Soru üretimi için 65 adet şablon soru hazırlanmıştır. 65 adet şablon 5 farklı kategoride sunulmuştur (*betimsel*, *karşıolgusal*, *sebebiyet*, *kolaylaştırma* ve *engelleme*). Bu şablonların içi videodaki sahnenin içeriğine göre otomatik olarak doldurulmuş ve bu şekilde soru-video çiftleri yaratılmıştır. Veri kümesi ile birlikte referans doğru cevapların hazırlanması için video yaratılırken objeler ve olaylarla ilgili bilgiler kaydedilmiştir. Doğru cevabı otomatik bulmak için gereken tüm bilgiler veri kümesi ile birlikte paylaşılmıştır.

Sadece uzaysal muhakeme yetenekleri sorgulatan güncel görsel soru cevaplama çalışmalarının aksine [21], CRAFT sorularına doğru cevap verebilmek için modellerin kuvvetli zamansal ve nedensel muhakeme yeteneklerinin olması gerekmektedir. Ayrıca, sorulara yüksek oranda doğru cevaplar verebilmek için modellerin karışık fiziksel olaylar için de temsil yeteneğinin kuvvetli olması gerekmektedir. Bu tezin çoğunluğunu veri kümesinin nasıl oluşturulduğuna ayrılmıştır. Ayrıca, bu tez içinde güncel yapay modellerin CRAFT üzerinde nasıl çalıştığını görmek için yapılan deneyler hakkında bilgiler de mevcuttur. Yaptığımız bu deneyler, veri kümesindeki cevap sıklıklarını inceleyen kolay modelleri içerdiği gibi, daha karışık derin öğrenmeye dayalı modelleri de içermektedir. CRAFT'ı oluştururken temel amacımız insanlar için kolay; ama yapay zekâlar için zor bir veri kümesi oluşturmaktı. Bu amacımızı gerçekleştirdiğimize yönelik iddiamızı desteklemek için 12 adet yetişkin katılımcı ile bir deney daha gerçekleştirdik. CRAFT içerisinden rastgele seçtiğimiz soruları katılımcılara sorduk. Tezin sonunda, katılımcıların gösterdiği performans ile eğittimiz temel modeller arasında en başarılı olan modelin performansı arasında geniş bir fark olduğunu raporladık. Bu fark bize bu tarz yeteneklerin yapay yollarla kazanılması için daha çok çalışılması gerektiğini göstermiştir.

Sonuç olarak, CRAFT adlı veri kümemizi ve bu veri kümesini yaratırken kullandığımız araçları literatüre sunarak, bu tezin, yapay zekâların insan zekâsına yaklaşması yolunda ufak bir basamak olacağına inanmaktayız. CRAFT üzerinde eğitilecek daha başarımlı modellerin ortaya çıkmasını ve CRAFT'ın değinmediği yerlere değinen yeni veri kümelerinin yaratılmasını hedeflemekteyiz.

CONTENTS

Page

TABLES

FIGURES

ABBREVIATIONS

AI	**A**rtificial **I**ntelligence
QA	**Q**uestion **A**nswering
VQA	**V**isual **Q**uestion **A**nswering
CRAFT	**C**ausal **R**easoning **A**bout **F**orces and In**T**eractions
CLEVR	**C**ompositional **L**anguage and **E**lementary **V**isual **R**easoning
ResNet	**Res**idual **Net**work
LSTM	**L**ong **S**hort-**t**erm **M**emory
RNN	**R**ecurrent **N**eural **N**etwork
CNN	**C**onvolutional **N**eural **N**etwork
GNN	**G**raph **N**eural **N**etwork
TIWIQ	**T**able-top **I**nteraction **V**isual **W**hat **I**f **Q**uestions
CLEVRER	**CO**ll**L**ision **E**vents for **V**ideo **RE**presentation and **R**easoning
PROPNET	**PROP**agation **NET**work
MFA	**M**ost **F**requent **A**nswer
AT-MFA	**A**nswer **T**ype-based **M**ost **F**requent **A**nswer
MAC	**M**emory, **A**ttention, and **C**omposition
MAC-V	**M**emory, **A**ttention, and **C**omposition for **V**ideo
G-SWM	**G**enerative **S**tructured **W**orld **M**odels

1. INTRODUCTION

Imagine yourself playing bowling with some of your friends and the turn is yours. Your aim is to knock the pins ahead of you, possibly with a strike. Regardless of your experience in the game, you can estimate whether it will be a good hit or not just after you bowl using your intuitions about the environment. You may consider the direction, speed or spin of the ball and current positions of the pins to do the estimation. Your estimations are all based on approximate predictions. You do not try to use Newtonian physics exactly in your estimations in this very short period of time. The ability of humans to understand and make approximate predictions about the physical environments consisting of different objects that are in steady state or in motion is known as intuitive physics [29]. Humans gain the collection of these type abilities starting from their birth. Cognitive scientists extensively studied which factors affect infants' or adults' ability of physical reasoning [30–33]. Some of these abilities are also studied for chicks (Gallus gallus) as well [34].

Recent advances in machine learning systems have enabled computers to understand what is the object in a specified image [1–3], which rectangle best wraps that object [4–6], what is its exact boundaries [7–9]. Some of these systems tried to understand what is happening in a single image [10, 11] or in a video [12, 13]. More recent systems started to generate new collections of data (such as human faces [14] or art samples [35]) by incorporating real existing data. Although, these artificial systems have been amazing us for decades, there are areas in which artificial systems are far from performing as humans do. One of these areas includes the humans' capability reasoning about physical actions of the objects by sensing the environment. This is a new recent research direction for which cognitive and machine learning scientists are working jointly to bring similar capabilities to the artificially intelligent robots so that they generate similar intuitions and understand their environment more. One crucial point that is worth mentioning here is that improving physical reasoning capabilities can make them responsible for their actions in their physical environments. They can gain abilities to consider counterfactual actions without actually performing the actions. They can

estimate what will happen if they perform a specific action. One of the recent example in this research is the robot playing the game Jenga [15].

The research on understanding intuitive physics has started very recently, hence there are only a few studies that focus on this task. For instance, some current methods of intuitive physics in artificial intelligence tried to estimate whether a scene of objects are in stable configuration or not [16] while the other tried to predict where the objects fall after a simulation if the configuration is not stable [17]. Other methods tried to estimate a motion trajectory of a query object under different forces in an image [16]. Some other tried to build a stack configuration of the objects from scratch through a planning algorithm [18]. Some other researchers have recently extended the notion of the objects by considering them as a collection of particles to represent the fluids and deformable objects [19]. These are not only but some examples of teaching physics to deep neural networks.

1.1. Scope of the Thesis

The main aim of this thesis is to help machine learning models to understand and reason about physical relationships between dynamic objects in a scene. We propose a new visual question answering (VQA) task that requires understanding complex physical reasoning to be able to score high. VQA tasks require machine learning models to answer a question or questions about a visual which may be video or image according to the definition of the task (Figure 1.1.). Models are needed to be trained with multimodal datasets which contain visual and textual information to correctly answer the question. The performance of the model then can be calculated by the ratio between the number of correctly answered questions and all questions.

By making use of the experience gained for visual question answering tasks, a new virtual dataset named CRAFT (Causal Reasoning About Forces and inTeractions), is created [36]. The dataset contains virtually generated 2-dimensional videos as well as questions regarding those videos. The most prominent properties of CRAFT dataset are that it contains visuals which include complex physical interactions between objects and tasks which question

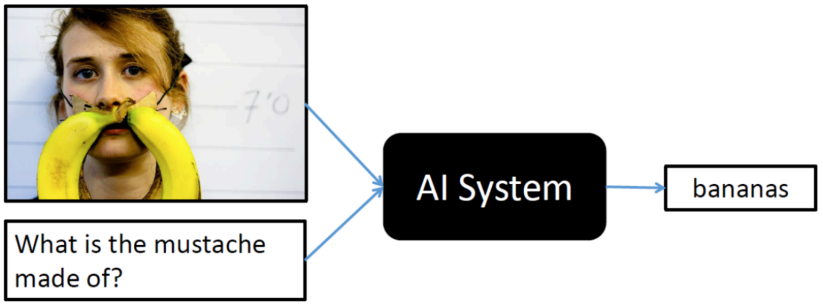


Figure 1.1. Visual Question Answering (image taken from [22]).

strong reasoning capabilities. For example, understanding the relations and detecting the object or objects which are causing, enabling or preventing certain events from happening are some of these capabilities. Moreover, understanding what would have happened if a slight change in the visual is created is also one of them. Most of this thesis is dedicated to our efforts in creating this dataset. Furthermore, some neural baseline and human study results are also provided to demonstrate that there is still more way to go for the current AI systems. We believe that the dataset will lead to the generation of more novel systems on the path of approaching human intelligence for physical reasoning.

1.2. Contribution

The contributions of this thesis can be listed as follows:

1. A new benchmark with a new virtual dataset (CRAFT) is proposed to the literature to improve causal and temporal judgements of the machine learning systems.

2. Simulators for creating new CRAFT visuals and questions are provided to extend the dataset further.

Implementations of the video simulator, the question generator, and baseline models will be publicly available to the community.

1.3. Organization

The rest of this thesis is organized as follows:

In Chapter 2, we provide detailed background information to fully grasp the rest of the thesis. Then, in Chapter 3, we briefly review the related works from the literature. In Chapter 4, we provide details of the proposed CRAFT dataset. We continue with the baseline models and experiments conducted on CRAFT with their results in Chapter 5. Finally, in Chapter 6, we conclude this thesis by providing some remarks about shortcomings of our baselines and some possible future research directions.

2. BACKGROUND

In this section, some background information about the related concepts required to understand this thesis comprehensively will be provided. Firstly, we present the formal definitions of intuitive physics, physical reasoning and common sense and some of the research controversies among cognitive science researchers. Advances in cognitive science and understanding the researchers' way of thinking can be very useful for developing machine learning models which do not only recognize patterns. Secondly, we provide some information about visual question answering studies in general. Then, we introduce the physics simulator that is used to create CRAFT. Moreover, we give some background information in automatic question generation before ending this section with details of the backbone models used in our baselines.

2.1. Intuitive Physics in Cognitive Science

Common sense can be considered as humans' collection of capabilities to perceive, understand and judge about everyday situations. These senses are also shared by all humans. The definition of common sense is also philosophically historical originating from the works of Aristotle. These senses are not required to contain physical activities belonging some objects. If we hear that doorbell of our house is ringing, we directly understand that someone has just arrived at our home. Moreover, when we raise our hands in a restaurant, the waiter understands that we need something. Intuitive physics, on the other hand, is considered as how people perceives how the physical world changing its state by objects' dynamism describing similar common sense beliefs [37]. If we represent the chain of events in a dynamic environment as a causal graph, the ability to find the reason events of some other events is titled as physical reasoning.

One of the main controversies among cognitive science researchers is to decide whether innate ideas have impact on infants' physical reasoning. While some are claiming that they are important, some claim that these intuitions can only be gained by experimenting. These

ideas include cohesion and continuity, which state objects are bounded and connected entities and objects exists and move continuously in time and space, respectively. Other than these ideas, the notion of variables is also important. Humans are able to identify different variables such as height and color of objects to predict the outcome of an event at different ages of their lives. However, the authors in [31] showed that when these variables are induced to the infants using proper mediums, they are started to predict the outcomes correctly regardless of the type of the event and age of theirs. This is somehow correlated to what we are doing in artificial intelligence systems. The data or the feature may not be significant until you provide it to the model using a more proper medium.

2.2. Visual Question Answering

Visual question answering (VQA) is a subdiscipline of question answering (QA) which is building artificial systems that are trying to give answers to the questions given in natural language. As well as containing natural language questions, VQA tasks consist of visuals for which the questions are generated. The performance of a model is calculated by the number of the correctly answered questions. The visuals can be both images or videos according to the definition of the task.

The datasets generated for visual question answering can be divided into two sub-categories according to how the visuals are created or collected. The first set of works created questions for real world images or videos [22, 38–41]. The other set of works used a computer simulator to generate virtual scenes for the questions [21, 42, 43]. From this perspective, this work belongs to the second category since we are also using Box2D simulator to generate our visuals.

Current machine learning models are very successful learning patterns in the datasets. This makes generating a VQA dataset more difficult because a bias existing in the dataset may lead the model to cheat answering the question without even considering the features obtained from the visual. For example, if the question asks the color of a car in the video or the image, the model may learn to give answer as *"red"* if most of the cars in the samples are

"red". Similarly, model may look at the visual and can answer without even looking at the question's natural language representation [44]. Detailed information about how we deal with certain biases in CRAFT will also be provided in Chapter 4.

Image visual question answering tasks are not able to emphasise dynamic state changes of the objects which is required to improve intuitive physics capabilities of the models. Although some of the video question answering works question temporal reasoning capabilities of the models [45–47], they do not require physical reasoning capabilities to answer the questions. On the opposite end, works developing models that can learn some sort of physical reasoning capabilities do not integrate visual question answering in the learning process considering mostly single physical events. In this sense, our aim is to enable learning physical reasoning capabilities through a VQA task.

2.3. Physics Simulation

There are plenty of 2-dimensional simulator alternatives on the internet to be used for research purposes freely. From them, we have decided to go with Box2D [20]. The library is implemented fully in C++ and includes continuous collision detection algorithms and provide begin, end, pre-solve, post-solve contact callbacks from them we have detected our events in our physical environments. Box2D's engine also takes physical quantities like mass, friction and restitution into account in its calculations. It include complex joint types such as evolute, prismatic, distance, pulley, gear and mouse joint. Although it also provides a graphics engine as well as a physics engine, we developed a custom and flexible graphics engine to visualize our simulations. Finally, it has a nice test application for game developers and researchers to start coding their own simulations easily (Figure 2.1.).

2.4. Question Generation

As our simulations, our questions and their corresponding answers about the simulations are generated automatically. There are studies which generate datasets whose questions are annotated by humans [22, 38]. Although this is an option, human annotations make expansion

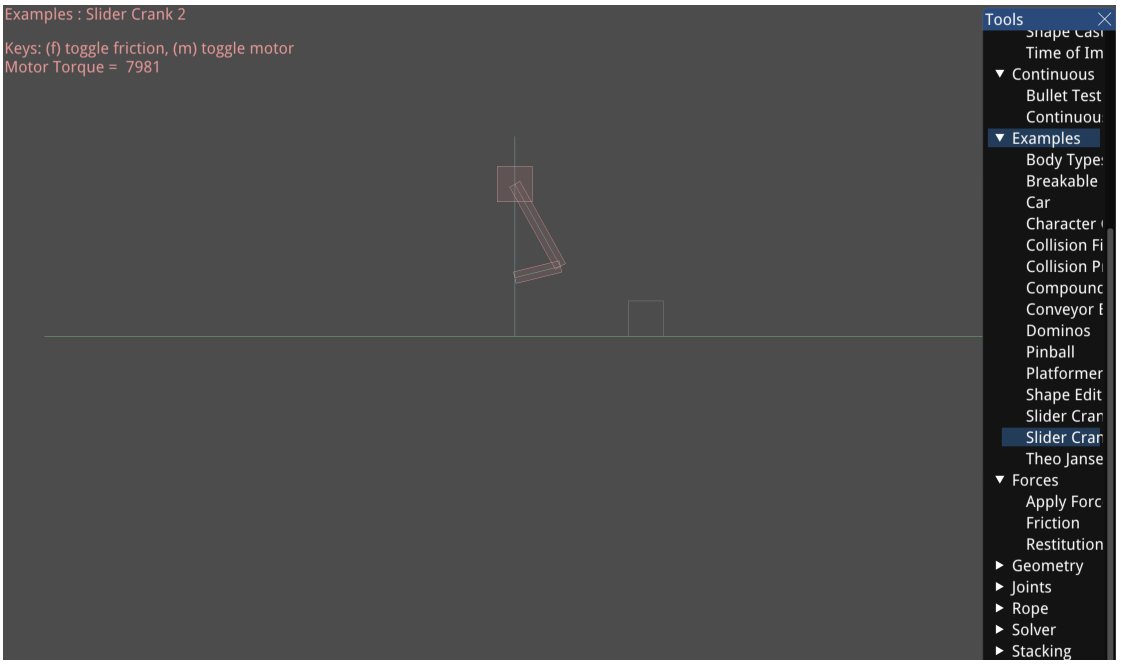


Figure 2.1. Box2D Test Application Interface.

of the dataset very difficult and expensive. Furthermore, there are synthetic question generators which depend on the descriptions provided with the visuals [39]. Main idea in these studies is to convert descriptions, such as the ones in MS-COCO dataset [48], into question and answer pairs. For example, if the description about the visual is *"A man is riding a horse"*, then the question and the answer become *"What is the man riding?"* and *"A horse"*, respectively. One advantage of the method is that if the descriptions are human-like, then the questions become human-like preserving variability in the language. On the other hand, one disadvantage of the method is that it depends hardly on the descriptions. This makes impossible to create a set of questions about a visual containing lots of objects as in our work.

Recent advances in automatic question generation (CLEVR, [21]) has enabled dataset generators to create synthetic questions having some sort of formal structures. CLEVR images are constructed by some set of objects with different attributes. Objects are placed in a scene creating spatial relationships. A scene represents an image for which some questions are generated. CLEVR represents each question with a functional program consisting of several

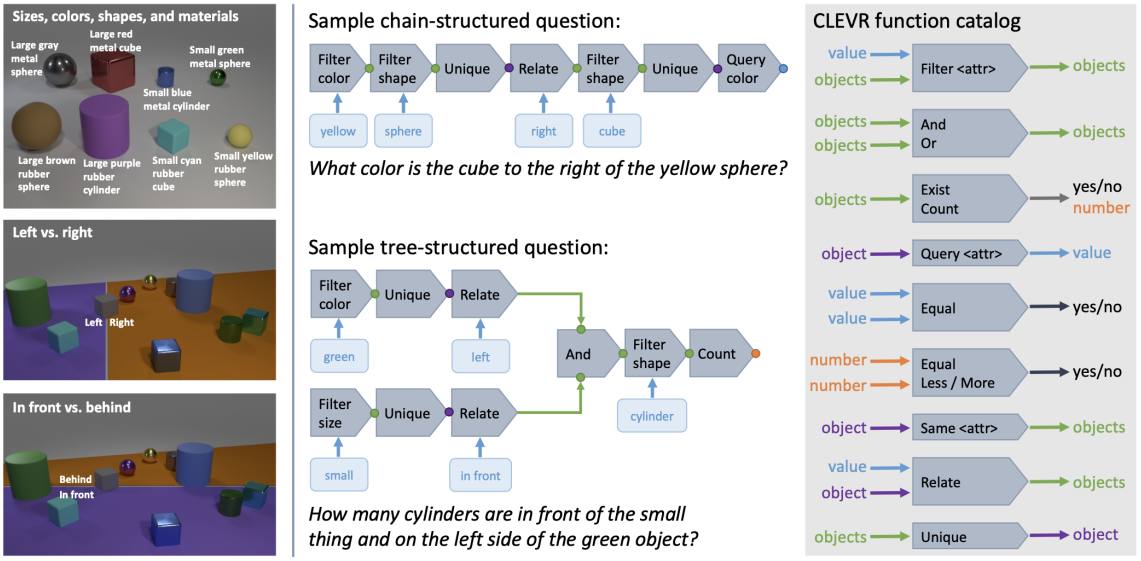


Figure 2.2. CLEVR World. **Left:** Object properties and representation of spatial relationships. **Middle:** Functional program examples. **Right:** Filter examples. (image taken from [21])

filters attached to each other as shown in Figure 2.2. Starting from a formal scene representation, each filter is applied to the output of the previous filter(s) to obtain a single answer by the last filter of the program. A functional program and a question pair correspond to a specific task in CLEVR and this task can be varied by the use of side inputs such as shape, material, size and color. These functional programs inherit the reasoning abilities required to answer a single question. For example, a program of a question, which has *"in front of"* preposition, contains a **Relate** filter stating that spatial reasoning is required to answer this question. Our approach in CRAFT is very similar to CLEVR's approach for which we extend this work to question temporal and causal reasoning capabilities of the machine learning models. Instead of static scenes, our scene are dynamic consisting of complex physical interactions between objects. Details of our dataset generation method can be found in Chapter 4.

2.5. Deep Learning Backbones

Since CRAFT is a new visual question answering dataset, it provides data in both visual domain and textual domain. Here we provide information about some core models which are

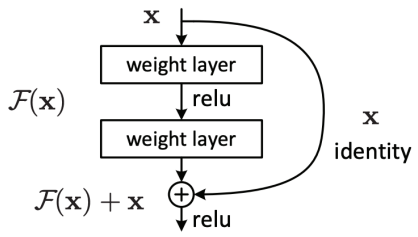


Figure 2.3. A block skipping 2 layers in ResNet (image taken from [23])

crucial for understanding baseline models which are tested on CRAFT. Detailed information about these neural baselines will be given in Chapter 4.

Residual Network [23]: Residual Network, aka. ResNet is an artificial neural network which is mainly used for feature extraction to represent images. This work proposes utilizing skip connections which leads model to bypass two or more layers. A simple building block in a ResNet uses input feature map as a residual adding it to the output that it provides as in Figure 2.3. Using this block in ResNets is shown to be a successful way for avoiding vanishing gradient problem of deep neural networks. This problem is mostly encountered when the model is too deep so that the gradients extracted vanish before reaching some or all layers preventing weight updates for them. Avoiding such a problem enables obtaining deeper representations of the visuals. Some of our baselines utilize PyTorch implementation of ResNet-18 which is pretrained on ImageNet 2012 dataset [49].

3D Residual Network [50]: After the huge success of 2D ResNets on image tasks, they are also integrated to solve some problems receiving video inputs by applying them to individual frames. On the other hand, using 3D convolutional layers in a residual setup have shown to be more successful than 2D convolutional layers in action recognition task preserving spatiotemporal content better. Some of our baselines also use PyTorch implementation of 3D ResNet (r3d-18) which is pretrained on Kinetics-400 dataset [13].

Long Short-term Memory Network [51]: Long Short-term Memory Network, aka. LSTM is an artificial neural network which is used to represent sequential data such as text or video. This network is a special type Recurrent Neural Network (RNN). In RNNs, representation of an individual item in the sequential data does not only depend on the item itself, but also

it depends on the previous items. This dependency is important for inputs, such as natural language sentences, for which the meaning of an item (word) should be updated by the meaning of a previous item. LSTMs are designed to overcome the problem of vanishing gradient problems in naive RNNs enabling extracting features from longer sequences. LSTMs avoid this problem by proposing the usage of gates which are input, forget, and output gates. These gates learn whether the information written to the cell state is important or not. Irrelevant information extracted can be removed from the state and it is not used to update the states of next items. All of our baselines utilize LSTMs to represent CRAFT questions.

3. RELATED WORK

This section is provided to give information about the research in cognitive science and artificial intelligence which have helped this thesis to be created. As in Chapter 2, we begin by investigating the related work in cognitive science.

3.1. Intuitive Physics in Cognitive Science

Chapter 2 has provided some background information about how cognitive scientists approach some of the interesting problems intuitive physics. Here, we extend those studies with some others which are very closely related what we are trying to achieve building CRAFT dataset in detail.

3.1.1. Causal Reasoning With Forces

Understanding how people perceive events or statements requiring causal reasoning is important. Causal reasoning may require understanding the intentions of possible multiples of affectors or patients which apply forces to each other in an environment. The affector may intend to help the patient to do its task, or it may try to prevent the patient from doing its task. In such scenarios, patient should have a task to be accomplished. This direction of causal reasoning is also at the core of the CRAFT. Our questions about the simulations inherit the task of the patient and also the intention of the affector. Therefore, understanding these is crucial solving some of the CRAFT tasks.

From cognitive science point of view, researchers have developed several theories how humans reason about the causal events that consist of affectors and patients. Some of these theories are mental model theory [52] and causal model theory [53] whose units of perceptions are abstract. Mental model theory states that logical operations are used in sub-relations of complex compositions to produce a single conclusion, whereas causal model theory is based on a Bayesian network where the relations are represented probabilistically. In [54], authors

propose a new theory which they title as force theory for which their first claim is that the units of perception need not to be abstract as in the previous theories and can resemblance to real world entities. They represent cause, help, and prevent relations by different configurations of the forces appeared in the affector and the patient. Therefore, composition of sub-relations to obtain the final (possibly more than one) result can be achieved by transferring or eliminating the forces existing in the relation. In their experiments, they created physical environments based on 3D simulators and compared participants' perception results with the outputs of three models mentioned. They observe that the force theory predicts as well as or better than the theories which are based on abstract units. They also provide experiment results evaluating abstract causation performances. They compared how close predictions of force theory to humans when compared to other theories when provided causal compositions. This time the forces are not real world forces but are approximate influences of affectors on again patients. They observe a similar pattern in their experiments evaluating abstract causation. All three models are very successful mimicking human performances with few mistakes. However, depending on iconic representations, force theory can better explain how these events might be perceived.

3.1.2. Game Engine as an Architecture for Intuitive Physics

As most of the virtual dataset generators, we adopt a game engine to run our simulation. If a machine learning model can simulate as with our simulator, it would be much easier for the model to answer the question. It would only require to extract the causal relationships of the objects whose positions, velocities, rotations etc. are extracted easily. Use of game engines in intuitive physics research in machine learning is also cognitively important since there are researches which demonstrate the similarities between the human mental process and a game engine [55]. This paper investigates the hypothesis that intuitive decisions about physics are made by the help of a mental engine which has similar characteristics with game physics engines especially for the young infants. This hypothesis claims that the data structures in our mind to represent the objects and the events, and the algorithms to simulate have similar characteristics with those provided by video game industry. One of the facts for

authors to support their hypothesis is that both mental and game engines are designed to approximate the complex scenes to a reasonable-looking and human-relevant scale. Other than the similarities of representing objects and events, bodies and shapes, static and dynamic objects; resolving the collisions between mental processings and physics engines, they both fail to identify exact physical situations in some physical illusions because of some simplified assumptions made by their processes.

3.2. Intuitive Physics in Artificial Intelligence

Besides the research conducted in cognitive science, there are studies which investigate similar types of notions in artificial intelligence. Firstly, we provide related work about learning architectures which are mainly developed to improve physical, temporal and causal reasoning capabilities of the machine learning models. Then, we focus on some of the datasets stating the differences between those and ours.

Naive convolutional neural network (CNN) architectures have proven their worth representing images or videos with smaller dimensions to be able to solve problems such as classification, detection, segmentation. Although they are also used in mostly feature representation for physical understanding of the models, recent advances get benefit from graph neural networks (GNN). Furthermore, GNNs, by their nature, are capable of representing objects (nodes) and complex relationships (edges) between these objects inside in a graph. As oppose to CNNs which can work on Euclidean space such as images, video or text, GNNs enable working on non-Euclidean space to ease problems like node classification, edge prediction or clustering [24]. Figure 3.1. illustrates the difference between Euclidean and non-Euclidean spaces.

Here, we provide some, but not all, architectures which are developed to understand physics in different environments. In [56], authors propose to compute interactions and effects between objects via a relational model, and uses this model to predict how the interactions and dynamics influence the objects. However, their model did not consider effects of a relation between two objects on other objects as can be observed for most of the real life scenarios.

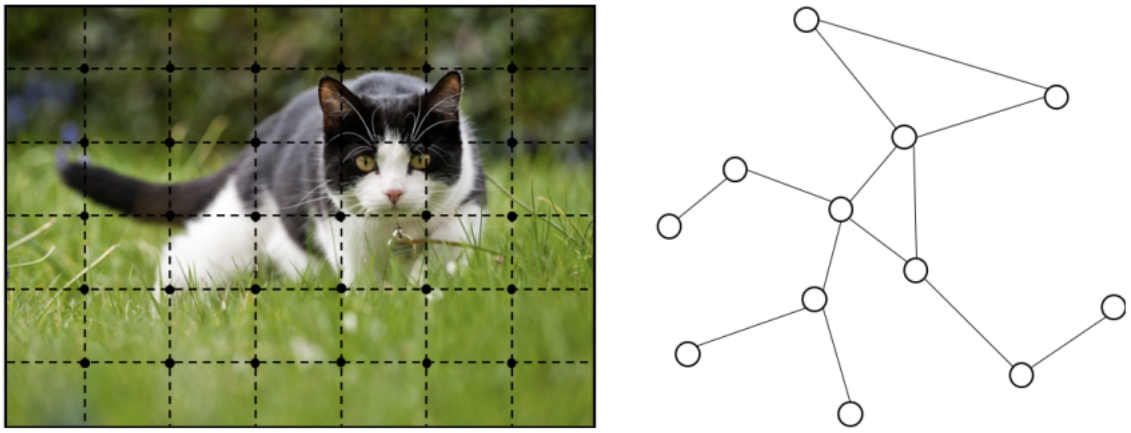


Figure 3.1. Difference between Euclidean and non-Euclidean spaces. **Left:** Image patches lying on a Euclidean space. **Right:** Nodes and edges on a non-Euclidean space. (image taken from [24])

Therefore, in [57], Li et al. propose Propagation Networks to propagate those effects of interactions in a single time step. When the number of objects gets higher, this propagation starts to slow down the execution of the network. A possible solution for this problem is somehow grouping the objects by adding hierarchy to objects in the scene. In [25], Mrowca et al. uses a hierarchical particle-based object representation for rigid and deformable bodies, and a hierarchical graph convolution to predict physical dynamics. The idea behind building this types of hierarchies is visualized in Figure 3.2. Furthermore, Ye et al. built an intuitive physics model with a focus on interpretability, where specific vectors in the model represent specific physical parameters like mass, friction and speed [58]. Lastly, some extended understanding physics with planning in order to build new configurations of object stacks as well as understanding physical events occurred [59].

3.2.1. PHYRE: A New Benchmark for Physical Reasoning

Very recently, in [26], Bakhtin et al. created the PHYRE benchmark dataset that consists of different types 2D-environments. Each environment inherits a task to be completed by an agent's smart action (*"make green ball touch the blue wall"*). An example can be seen in Figure 3.3. The agent must reason about the scene and predict the possible outcomes of its move to provide the best possible action completing the task. Authors have set three

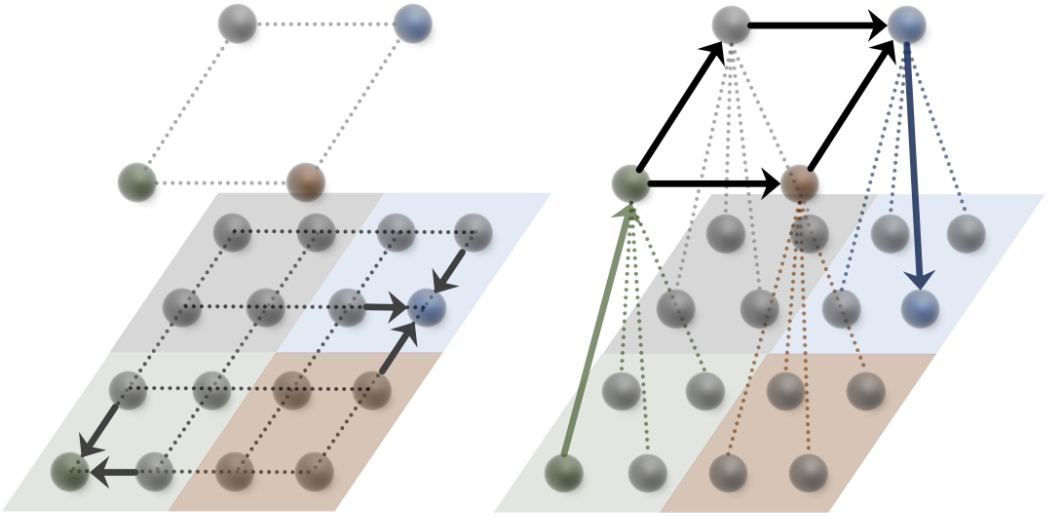


Figure 3.2. The idea behind using hierarchical representations when the number of objects gets high as in representing fluids. **Left:** Local propagation between objects of the same level. **Right:** Hierarchical propagation between representations of object groups in different levels. (image taken from[25]).

main goals for PHYRE benchmark. The models must focus on physical reasoning, the models must perform also well in the scenes that they do not see during training, models must find the best action with as few attempts as possible. To evaluate the second goal, authors have separated train and test task templates. Furthermore, to evaluate the last goal, authors propose an evaluation metric penalizing if the number of attempts get increased. PHYRE consists of two tiers. First tier requires finding a ball radius and position (3 dimensional) that would complete the task. The second tier requires finding ball pairs with their radii and positions (6 dimensional). Each tier consists of 25 templates and each template has many initial state configurations. Although, there are many similarities between PHYRE and CRAFT in constructing 2D environments, they are different for the way of aiming to enable the implementation of physical reasoning algorithms for which PHYRE does not consider a VQA task, as CRAFT does.

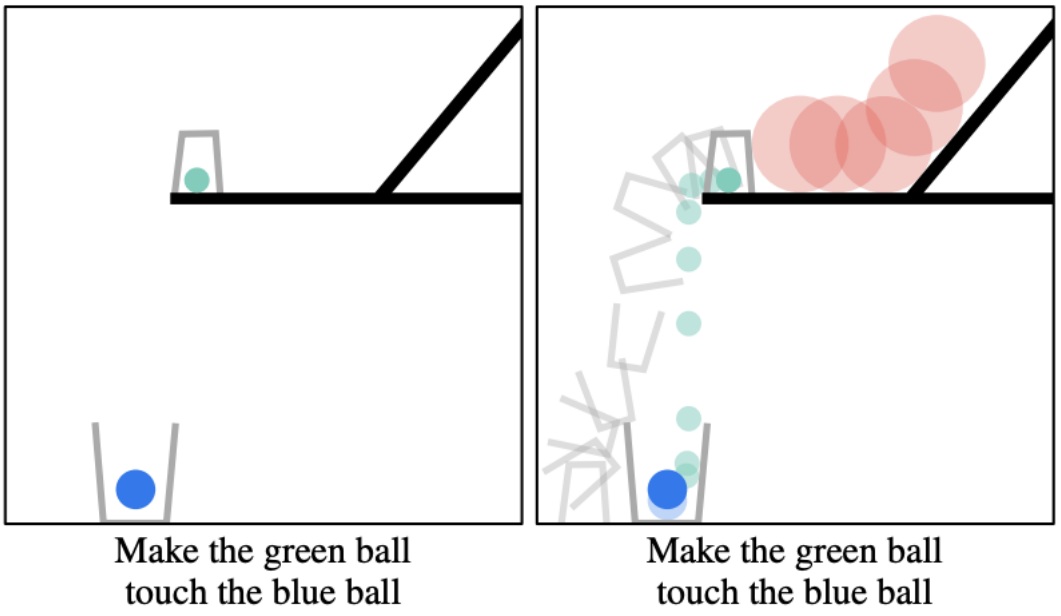


Figure 3.3. A sample task from PHYRE. **Left:** A task defined in PHYRE. **Right:** Solution for the task in left. (image taken from [26]).

3.2.2. Answering Visual What-If Questions: From Actions to Predicted Scene Descriptions

Current work for scene understanding and next frame prediction problems in robotics see the agents as passive observers and do not allow them to manipulate the environment. What-if question tasks, on the other hand, allow the scene to be manipulated by a hypothetical action. The main problem to be solved is to describe the outcome of an action on a table top scenario. To solve their problem, Wagner et al. created a dataset, Table-top Interactions Visual What-If Questions (TIWIQ), consisting of 3D scenes of realistically textured objects with interactions [60]. Scenes contained five of eight realistic looking objects such as brick, banana, softball. They utilize four different actions which are (1.) Push an object in a specific direction. (2.) Rotate an object clockwise or anti-clockwise. (3.) Remove an object from the scene. (4.) Drop an object on another object. They gather annotations on top of simulation rendering videos and ask their model and human baseline to output as similar as possible with the annotations. Their prediction tool includes a hybrid question answering model for which they integrate a physics engine. The engine gets input from learning-based components to

simulate and the result of the simulation is then tried to be expressed as a natural language output.

3.2.3. CLEVRER: CoLlision Events for Video REpresentation and Reasoning

Yi et al. proposes a new dataset, CoLlision Events for Video REpresentation and Reasoning (CLEVRER), for evaluating reasoning performance of the models trained for video understanding [43]. Besides recognizing visual features inside the video, this dataset challenges models to understand the dynamics between objects and events and answer to the questions which require causal analysis to recognize the events and their reasons. The dataset contains four different question types; descriptive, explanatory, predictive and counterfactual whose samples are provided below. There are three events which are enter, exit and collision from them collision is the reason for extracting a causal graph of the events. The dataset objects, materials, colors are very similar to CLEVR dataset except the fact that authors use a physics engine to simulate events and a render engine to create videos instead of outputting single images. A couple of baseline models are trained and evaluated on CLEVRER. While the overall performance of baseline models are quite low, authors observed that using object segmentation maps as explicit object representations increases performances. The second observation that the authors provided is that the dynamics modeling is required as well as explicit object representations in models to be able to be successful in such datasets requiring investigation of the causal structure between the events. Therefore, they also train a model utilizing Propagation Networks (PropNet) and show that the results are better compared to baselines.

TIWIQ and CLEVRER are two recent VQA datasets questioning intuitive physics capabilities of machine learning models. Compared to CRAFT, both lack of visual variations. CRAFT contains different types of simulations enlarging the variety in the visual domain as well. This variety also makes minimizing the dataset biases difficult because of the multiplicity in the number of the domains (textual and visual). Although CLEVRER integrates different types of tasks such as descriptive, predictive, explanatory, and counterfactual; they

do not directly question cause, enable, and prevent relationships, which are the backbones of causal reasoning.

4. CRAFT DATASET

CRAFT is built to evaluate temporal and causal reasoning capabilities of existing algorithms containing videos of 2D simulations and related questions. It has 37768 question and video pairs in total that are created from 3000 videos. In order to avoid having same video in more than one split, CRAFT's train, validation, and test sets are created by splitting videos with ratios of 0.5, 0.3, and 0.2, respectively. Furthermore, our train, validation, and test sets contain 18806, 11430, and 7750 question and video pairs, respectively. We provide an example set of questions from CRAFT in Figure 4.1. In this chapter, we mention how we generate visual scenes, which types of objects and events exist in our visuals and questions, how we represent our simulations, how we generate questions and the corresponding tasks, and finally, how we minimize the biases that may occur in visual question answering datasets [1].

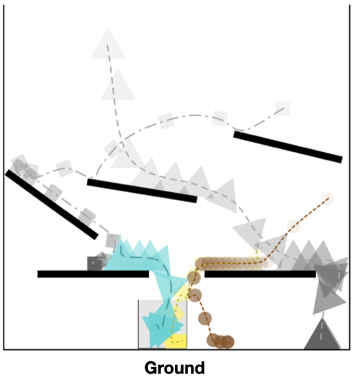
4.1. Video Generation

We use Box2D [20] to create our virtual scenes. There are 10 different scenes from them we extract 10 seconds videos whose resolutions are 256 by 256 pixels. Besides generating original simulation video, CRAFT scripts also generate variation videos by removing each object of the same video from the scene. These variation videos help question generation script to provide answer for certain types of questions.

4.2. Objects

There are static scene elements and dynamic objects in our scenes. Each scene includes variable number of and different type of these elements and objects. There are 6 static scene elements, namely *(ramp, platform, basket, left wall, right wall, ground)*. These elements are all drawn in *black* color in the video sequences. Their attributes such as position or orientation are decided at the beginning of a simulation and then they are fixed throughout the

[1] It should be noted that Çağatay YİĞİT and Muhammed Şamil ATEŞOĞLU contributed to this work by helping us to increase the number of scene layouts and to construct the data splits, respectively.



Descriptive
Q: "How many objects **fall to** the **ground**?" A: "2"
Q: "After **entering** the **basket**, does the **small yellow square** collide with other objects?" A: "True"

Counterfactual
Q: "How many objects **fall to** the **ground**, if the **small yellow box** is removed?" A: "1"
Q: "Does the **small gray box enter** the **basket**, if any other single one of the objects is removed?" A: "True"

Cause
Q: "Does the **small brown sphere cause** the **tiny yellow box** to **enter** the **basket**?" A: "True"
Q: "Does the **small gray block cause** the **big cyan triangle** to collide with the **basket**?" A: "True"

Enable
Q: "Does the **small brown sphere enable** the **small yellow box** to **enter** the **basket**?" A: "False"
Q: "How many objects does the **small** gray **block enable** to **enter** the **basket**?" A: "0"

Prevent
Q: "Does the **small yellow square prevent** the **tiny brown circle** from **entering** the **basket**?" A: "True"
Q: "How many objects does the **large cyan triangle prevent** from **entering** the **basket**?" A: "1"

Figure 4.1. Example CRAFT questions for a specific scene. There are 65 different tasks divided into 5 distinct categories for 10 different scenes. Besides having tasks questioning descriptive properties possible needing temporal reasoning, CRAFT proposes challenges including more complex tasks requiring single or multiple counterfactual analysis or understanding object intentions for deep causal reasoning.

video sequence. The values of these attributes are assigned randomly from sets of different intervals which are predefined for each type of scene as in Figure 4.2. The set of the dynamic objects contains 3 shapes *(cube, triangle, circle)*, 2 sizes *(small, large)*, and 8 colors *(gray, red, blue, green, brown, purple, cyan, yellow)*. Attributes of dynamic objects, on the other hand, are in continuous change throughout the sequence due to the gravity or the interactions that they are subject to, until they rest.

4.3. Events

To represent the dynamism in the simulations formally, we extract different types of events from our simulations. They are *Start*, *End*, *Collision*, *Touch Start*, *Touch End*, and *Basket End Up*. *Start* and *End* events represent the start and the end of the simulations, respectively. Although we only question *Collision* events in our tasks, we want from algorithms to differentiate a collision from a touching event. Therefore, *Touch Start*, *Touch End* are also extracted by our simulations. Finally, *Basket End Up* event is triggered if the object enters the one and only basket in our scenes. All events inside a simulation are represented in a causal graph for the question generator to extract causal relationships easily. Causal graph is a directed graph where events are represented as nodes. Each edge represent a cause relation where the source event is considered as the cause of target event because of the shared

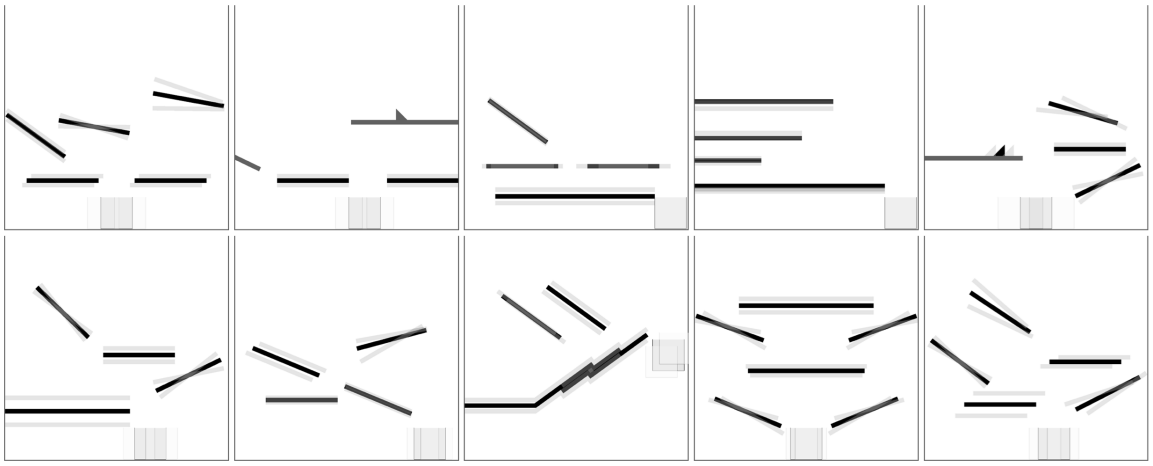


Figure 4.2. Random configurations of static scene element properties for each scene. The opaque regions show the mean value for that element, whereas the overlayed regions show the extreme values.

objects between them. Utilization of causal graphs is important since they help finding automatic answers to the questions requiring causal reasoning.

4.4. Simulation Representation

A video simulation sample is represented by three different structures, which are *scene representation at the start*, *scene representation at the end*, and *causal graph of events*. Scene representations hold information regarding attributes of static scene elements and dynamic objects such as color, position, shape, velocity, etc. at the start and at the end of a simulation. These structures are described as collections of attributes of objects rather than object graphs as in CLEVR by [21] since CRAFT does not question spatial relationships of different objects. On the other hand, the last structure holds causal relationships between events of a simulation. These structures about a simulation are enough to find the correct answer to a CRAFT question related to the simulation.

4.5. Tasks

CRAFT has 65 different question types under 5 different categories which are *Descriptive, Counterfactual, Enable, Cause, Prevent* which are listed in Table 4.1. and Table 4.2. Although the names for the last three categories are chosen from the representations of causal relationships in cognitive science since they directly require understanding these, the tasks in all categories require some degree of causal reasoning capabilities. *Descriptive* tasks mainly require extracting the attributes of objects and some of them, especially those involving counting, need some temporal analysis as well. CRAFT extends the work CLEVRER by [43] with different types of events and multiple environments. *Counterfactual* tasks require understanding what would happen if one of the objects was removed from the scene. Exclusive to CRAFT, some *Counterfactual* tasks (*"Does the small gray circle enter the basket, if any other single one of the objects is removed?"*) require multiple counterfactual simulations to be answered. As an extension to *Counterfactual* tasks, *Enable, Cause, Prevent* tasks require grasping what is happening inside both the original video and the counterfactual video. In other words, models must infer whether an object is causing or enabling an event or preventing it by comparing the input video and the counterfactual video that should be simulated somehow. Dynamic objects in one of the *Enable*, *Cause*, and *Prevent* tasks can be associated with two entities, the affector and the patient similar to causal reasoning research in cognitive science. CRAFT questions for these tasks explicitly specify the affector and the patient objects. Although there is a single affector for each tasks, some questions require multiple patient objects to be considered.

In order to have a better understanding of the differences between *Enable*, *Cause*, and *Prevent* questions, one should understand the "intention" of the objects. We identify the intention in a simulation by examining the initial linear velocity of the corresponding object. If the magnitude of the velocity is greater than zero, then the object is intended to do the task specified in the question text, such as entering the basket or colliding with the ground. If the magnitude of the velocity is zero, then the object is not intended to do the task, even if there is an external force, such as gravity, upon it at the start of the simulation. Therefore, an affector

can only enable a patient to do the task if the patient is intended to do it but fails without the affector. Similarly, an affector can only cause a patient to do the task if the patient is not intended to do it. Moreover, an affector can only prevent a patient from doing the task if the patient is intended to do it and succeeds without the affector.

4.6. Question Generation

CRAFT questions are represented with functional programs as in CLEVR. Functional programs enable finding the answers to questions automatically by parsing the inputs provided for the scenes or simulations. These programs consist of functional modules which are similar to the functions of a programming language having set of inputs and outputs. Input and output types for CRAFT's functional modules are listed in Table 4.3.

CRAFT includes input functional modules which provide initial information about our scenes. These input modules do not receive any inputs. The list of CRAFT's input modules is provided in Table 4.4.

Each program ends with a module which has the output type corresponding to the on of the answer types (e.g. color, shape, boolean) used in CRAFT. These are called output functional modules and listed in Table 4.5.

A functional program can be considered as a directed acyclic graph where the nodes are all functional modules residing from input modules to single output module. Other than input and output modules there are three different types of functional modules. The first type of modules are object filter functional modules which are used to extract static or dynamic object attributes. The list of these modules is provided in Table 4.6. Since our scenes are not static as in CLEVR, we have to extract information about the simulation from events to find answers to questions requiring temporal and causal reasoning. Therefore, the second type of modules are event filter functional modules which are listed in Tables 4.7. and 4.8. Finally, the last type of modules are auxiliary functional modules which consist of some helper modules such as set operations. They are listed in Table 4.9.

Table 4.1. CRAFT's descriptive tasks. ***Z***, ***C***, and ***S*** correspond to templates for Size, Color, and Shape attributes, respectively.

Descriptive Tasks
*"What color is the object that the **Z C S** first collides with?"*
*"What shape is the object that the **Z C S** first collides with?"*
*"What color is the object that the **Z C S** last collides with?"*
*"What shape is the object that the **Z C S** last collides with?"*
*"How many **S**s are moving when the video ends?"*
*"How many **C** objects are moving when the video ends?"*
*"How many **Z** objects are moving when the video ends?"*
"How many objects are moving when the video ends?"
*"How many **S**s enter the basket?"*
*"How many **C** objects enter the basket?"*
*"How many **Z** objects enter the basket?"*
"How many objects enter the basket?"
*"How many **S**s fall to the ground?"*
*"How many **C** objects fall to the ground?"*
*"How many **Z** objects fall to the ground?"*
"How many objects fall to the ground?"
*"How many **S**s collide with the basket?"*
*"How many **C** objects collide with the basket?"*
*"How many **Z** objects collide with the basket?"*
"How many objects collide with the basket?"
*"How many objects enter the basket after the **Z C S** enters the basket?"*
*"How many objects enter the basket before the **Z C S** enters the basket?"*
*"How many objects fall to the ground after the **Z C S** falls to the ground?"*
*"How many objects fall to the ground before the **Z C S** falls to the ground?"*
*"How many objects collide with the basket after the **Z C S** collide with the basket?"*
*"How many objects collide with the basket before the **Z C S** collide with the basket?"*
*"After entering the basket, does the **Z C S** collide with other objects?"*
*"Before entering the basket, does the **Z C S** collide with other objects?"*
*"After falling to the ground, does the **Z C S** collide with other objects?"*
*"Before falling to the ground, does the **Z C S** collide with other objects?"*
*"After colliding with the basket, does the **Z C S** collide with other objects?"*
*"Before colliding with the basket, does the **Z C S** collide with other objects?"*
*"Are there any collisions between objects after the **Z C S** enters the basket?"*
*"Are there any collisions between objects before the **Z C S** enters the basket?"*
*"Are there any collisions between objects after the **Z C S** falls to the ground?"*
*"Are there any collisions between objects before the **Z C S** falls to the ground?"*
*"Are there any collisions between objects after the **Z C S** collides with the basket?"*
*"Are there any collisions between objects before the **Z C S** collides with the basket?"*

Table 4.2. CRAFT's other task categories. ***Z-Z2***, ***C-C2***, and ***S-S2*** pairs correspond to templates for Size, Color, and Shape attributes of two different objects, respectively.

Counterfactual Tasks
*"Does the **Z2 C2 S2** enter the basket, if the **Z C S** is removed?"*
*"Does the **Z2 C2 S2** fall to the ground, if the **Z C S** is removed?"*
*"Does the **Z2 C2 S2** collide with the basket, if the **Z C S** is removed?"*
*"How many objects enter the basket, if the **Z C S** is removed?"*
*"How many objects fall to the ground, if the **Z C S** is removed?"*
*"How many objects collide with the basket, if the **Z C S** is removed?"*
*"Does the **Z C S** enter the basket, if any other single one of the objects is removed?"*
*"Does the **Z C S** fall to the ground, if any other single one of the objects is removed?"*
*"Does the **Z C S** collide with the basket, if any other single one of the objects is removed?"*
Enable Tasks
*"Does the **Z C S** enable the **Z2 C2 S2** to fall to the ground?"*
*"Does the **Z C S** enable the **Z2 C2 S2** to enter the basket?"*
*"Does the **Z C S** enable the **Z2 C2 S2** to collide with the basket?"*
*"How many objects does the **Z C S** enable to fall to the ground?"*
*"How many objects does the **Z C S** enable to enter the basket?"*
*"How many objects does the **Z C S** enable to collide with the basket?"*
Cause Tasks
*"Does the **Z C S** cause the **Z2 C2 S2** to fall to the ground?"*
*"Does the **Z C S** cause the **Z2 C2 S2** to enter the basket?"*
*"Does the **Z C S** cause the **Z2 C2 S2** to collide with the basket?"*
*"How many objects does the **Z C S** cause to fall to the ground?"*
*"How many objects does the **Z C S** cause to enter the basket?"*
*"How many objects does the **Z C S** cause to collide with the basket?"*
Prevent Tasks
*"Does the **Z C S** prevent the **Z2 C2 S2** from falling to the ground?"*
*"Does the **Z C S** prevent the **Z2 C2 S2** from entering the basket?"*
*"Does the **Z C S** prevent the **Z2 C2 S2** from colliding with the basket?"*
*"How many objects does the **Z C S** prevent from falling to the ground?"*
*"How many objects does the **Z C S** prevent from entering the basket?"*
*"How many objects does the **Z C S** prevent from colliding with the basket?"*

Representing a question with a functional program of modules provides also the information for the ability that is required to give an answer to a question. An algorithm should be able to apply filters which are similar to these modules to be able to perform operations of basic reasoning, such as counting objects satisfying a condition, understanding whether an event

Table 4.3. Input and output types of functional modules in CRAFT.

Type	Description
Object	A dictionary holding static and dynamic attributes of an object
ObjectSet	A list of unique objects
ObjectSetList	A list of *ObjectSet*
Event	A dictionary holding information of a specific event
EventSet	A list of unique events
EventSetList	A list of *EventSet*
Size	A tag indicating the size of an object
Color	A tag indicating the color of an object
Shape	A tag indicating the shape of an object
Integer	Standard integer type
Bool	Standard boolean type
BoolList	A list of *Bool*

Table 4.4. Input functional modules in CRAFT.

Module	Description	Input Types	Output Type
`SceneAtStart`	Returns the attributes of all objects at the start of the simulation	*None*	*ObjectSet*
`SceneAtEnd`	Returns the atttributes of all objects at the end of the simulation	*None*	*ObjectSet*
`StartSceneStep`	Returns `0`	*None*	*Integer*
`EndSceneStep`	Returns `-1`	*None*	*Integer*
`Events`	Returns all of the events happening between the start and the end of the simulation	*None*	*EventSet*

is causing another event etc., as people do. CRAFT includes tasks questioning complex temporal and causal reasoning capabilities which require understanding physics of the event, as well as simple visual reasoning capabilities. Creating combinations of different modules assures more complex questions to be generated leading different types of reasoning abilities to be questioned with a single question. Furthermore, integration of the functional modules

Table 4.5. Output functional modules in CRAFT.

Module	Description	Input Types	Output Type
`QueryColor`	Returns the color of the input object	*Object*	*Color*
`QueryShape`	Returns the shape of the input object	*Object*	*Shape*
`Count`	Returns the size of the input list	*ObjectSet*	*Integer*
`Exist`	Returns true if the input list is not empty	*ObjectSet* / *EventSet*	*Bool*
`AnyFalse`	Returns true if there is at least one false in a boolean list	*BoolList*	*Bool*
`AnyTrue`	Returns true if there is at least one true in a boolean list	*BoolList*	*Bool*

can ease doing detailed analysis about the performances of the algorithms to see whether they are capable of certain types of reasoning. It would be very difficult to do such detailed analyses if only natural language questions are provided.

Besides providing the list of modules used in programs, it is also important to observe them in action, i.e. helping giving answer to a question. Therefore, here, we provide example functional programs for some of the sample questions provided in Figure 4.1. which are used to extract the correct answers using our simulation environment. Figures 4.3. to 4.7. provide functional program samples that are designed for CRAFT descriptive, counterfactual, cause, enable, and prevent questions, respectively.

4.7. Variations in Natural Language

Language is one of the important key elements in human brain development. It also has an important role for allowing us to understand, speak, write causal relations between entities. For example, while reading a text, we are not just grasping the meaning of some ordered

Table 4.6. Object filter functional modules in CRAFT.

Module	**Description**	**Input Types**	**Output Type**
`FilterColor`	Returns the list of objects which have a color same with the input color	*(ObjectSet, Color)*	*ObjectSet*
`FilterShape`	Returns the list ofobjects which have a shape same with the input shape	*(ObjectSet, Shape)*	*ObjectSet*
`FilterSize`	Returns the list of objects which have a size same with the input size	*(ObjectSet, Size)*	*ObjectSet*
`FilterDynamic`	Returns the list of dynamic objects from an object set	*ObjectSet*	*ObjectSet*
`FilterMoving`	Returns the list of objects that are in motion at the step specified	*(ObjectSet, Integer)*	*ObjectSet*
`FilterStationary`	Returns the list of objects that are stationary at the step specified	*(ObjectSet, Integer)*	*ObjectSet*

words, their individual or collective representation. We make inferences, judgements connecting the ideas, events and states regarding the text we are reading [61]. These capabilities are developed starting from birth by the help of language development and improved with the variety. All of these statements are valuable not only for human intelligence but also artificially created intelligence.

It is crucial to enrich language variety for creating datasets consisting of natural language components created to empower the artificial models. In order to improve language variety, CRAFT data generation scripts for questions, first allow multiple paraphrased versions of the same text to be generated to represent the same task. For a question sample, a paraphrased version of the corresponding task is chosen randomly filling the object templates. Below, we

Table 4.7. Event filter functional modules in CRAFT.

Module	Description	Input Types	Output Type
`FilterEvents`	Returns the list of events about a specific object from an event set	*(EventSet, Object)*	*EventSet*
`FilterCollision`	Returns the list of collision events from an event set	*EventSet*	*EventSet*
`FilterCollisionWithDynamics`	Returns the list of collision events involving dynamic objects	*EventSet*	*EventSet*
`FilterCollideGround`	Returns the list of collision events involving the ground	*EventSet*	*EventSet*
`FilterCollideGroundList`	Returns the list of collision event sets involving the ground	*EventSetList*	*EventSetList*
`FilterCollideBasket`	Returns the list of collision events involving the basket	*EventSet*	*EventSet*
`FilterCollideBasketList`	Returns the list of collision event sets involving the basket	*EventSetList*	*EventSetList*
`FilterEnterBasket`	Returns the In Basket events	*EventSet*	*EventSet*
`FilterEnterBasketList`	Returns the list of In Basket event sets	*EventSetList*	*EventSetList*

share three paraphrased question texts belonging to the same *Enable* task consisting of an affector and a patient templates.

- *Does the **Z C S** enable the **Z2 C2 S2** to fall to the ground?*
- *Does the **Z C S** enable the collision between the **Z2 C2 S2** and the ground?*
- *There is a **Z C S**, does it enable the **Z2 C2 S2** to fall to the ground?*

Table 4.8. Event filter functional modules in CRAFT (continued).

Module	Description	Input Types	Output Type
`FilterBefore`	Returns the events from the input list that happens before input event	*(EventSet, Event)*	*EventSet*
`FilterAfter`	Returns the events from the input list that happened after input event	*(EventSet, Event)*	*EventSet*
`FilterFirst`	Returns the first event	*EventSet*	*Event*
`FilterLast`	Returns the last event	*EventSet*	*Event*
`EventPartner`	Returns the object interacting with the input object through the specified event	*(Event, Object)*	*Object*
`FilterObjectsFromEvents`	Returns the objects from the specified events	*EventSet*	*ObjectSet*
`FilterObjectsFromEventsList`	Returns the list of object sets from a list of event sets	*EventSetList*	*ObjectSetList*
`GetCounterfactEvents`	Returns the event list if a specific object is removed from the scene	*Object*	*EventSet*
`GetCounterfactEventsList`	Returns the counterfactual event list for all objects in an object set	*ObjectSet*	*EventSetList*

Secondly, CRAFT enables synonyms of certain words to be integrated. We choose a base word and create its synonyms inside the CRAFT context. Similar to question paraphrases, the base word is replaced by a synonym randomly at run-time. All synonyms including the base word have equal chance to be in the question text. This replacement is handled by word suffixes and verb conjugations by preserving English grammar. Below, we share the synonyms for the base words used for CRAFT dataset.

Table 4.9. Auxiliary functional modules in CRAFT.

Module	Description	Input Types	Output Type
`Unique`	Returns the single object from the input list, if the list has multiple elements returns `INVALID`	*ObjectSet*	*Object*
`Intersect`	Applies the set intersection operation	*(ObjectSet, ObjectSet)*	*ObjectSet*
`IntersectList`	Intersects an object set with multiple object sets	*(ObjectSetList, ObjectSet)*	*ObjectSetList*
`Difference`	Applies the set difference operation	*(ObjectSet, ObjectSet)*	*ObjectSet*
`ExistList`	Applies the Exist operation to each item in the input list returning a boolean list	*ObjectSetList* / *EventSetList*	*BoolList*
`AsList`	Returns an object set containing a single element specified by the input object	*Object*	*ObjectSet*

- **Nouns and Adjectives:** ***thing:*** *object;* ***sphere:*** *ball;* ***cube:*** *block;* ***small:*** *tiny;* ***ground:*** *bottom;* ***basket:*** *container, bucket*
- **Verbs:** ***prevent:*** *keep, hold, block, hinder;* ***enable:*** *permit, allow;* ***cause:*** *stimulate, lead, trigger;* ***enter:*** *go into, get into, end up in, fall into;* ***fall to:*** *hit, collide with*

4.8. Bias Minimization

Neural networks are very successful at recognizing the patterns in the data provided. Initial visual question answering studies suffered a lot from this fact. Initial models solving VQA tasks are found to have cheated by not actually understanding the visuals or questions, but by taking advantage of nonuniform distributions of the data [44]. Creating a VQA dataset

Question: *"How many objects fall to the ground?"*

```
Count (
        FilterDynamic (
                FilterObjectsFromEvents (
                        FilterCollideGround (
                                Events ()
                        )
                )
        )
)
```

Question: *"After entering the basket, does the small yellow square collide with other objects?"*

```
Var QueryObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Yellow"), "Cube" )
Var SmallYellowCubeEvents = FilterEvents ( Events(), QueryObject )
Exist (
        FilterAfter (
                FilterCollisionWithDynamics ( SmallYellowCubeEvents ),
                        FilterFirst (
                                FilterEnterBasket ( SmallYellowCubeEvents )
                        )
                )
        )
)
```

Figure 4.3. Example programs for *descriptive* questions.

Question: *"How many objects fall to the ground, if the small yellow box is removed?"*

```
Var QueryObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Yellow"), "Cube" )
Count (
        FilterObjectsFromEvents (
                FilterCollideGround (
                        GetCounterfactEvents ( QueryObject )
                )
        )
)
```

Question: *"Does the small gray box enter the basket, if any other single one of the objects is removed?"*

```
Var QueryObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Gray"), "Cube" )
Var OtherDynamicObjects = Difference ( FilterDynamic ( SceneAtStart() ), AsList ( QueryObject ) )
AnyTrue (
        ExistList (
                IntersectList (
                        FilterObjectsFromEventsList (
                                FilterEnterBasketList (
                                        GetCounterfactEventsList ( OtherDynamicObjects )
                                )
                        ),
                        AsList (
                                QueryObject
                        )
                )
        )
)
```

Figure 4.4. Example programs for *counterfactual* questions.

whose answers are uniformly distributed, is one of the major difficulties when considering handwritten questions or real world visuals. On the other hand, the problem becomes much easier for datasets which deal with simulated data since they have the full control of generation processes as in CLEVR [21] and CLEVRER [43].

Differently from CLEVR and CLEVRER, our dataset consists of video simulations from

Question: *"Does the small brown sphere cause the tiny yellow box to enter the basket?"*

```
Var AffectorObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Brown"), \Circle" )
Var PatientObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Yellow"), "Cube" )
Exist (
        FilterStationary (
                Intersect (
                        Difference (
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                Events()
                                        )
                                ),
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                GetCounterfactEvents (
                                                        AffectorObject
                                                )
                                        )
                                )
                        ),
                        AsList ( PatientObject )
                ),
                StartSceneStep()
        )
)
```

Figure 4.5. Example program for *cause* questions.

Question: *"How many objects does the small gray block enable to enter the basket?"*

```
Var AffectorObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Gray"), "Cube" )
Count (
        FilterMoving (
                Difference (
                        Difference (
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                Events()
                                        )
                                ),
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                GetCounterfactEvents (
                                                        AffectorObject
                                                )
                                        )
                                )
                        ),
                        AsList ( AffectorObject )
                ),
                StartSceneStep()
        )
)
```

Figure 4.6. Example program for *enable* questions.

10 different environments. Since it increases the variety in the visual domain, obtaining a uniform dataset which minimizes the biases is a more difficult process. Besides having 10 different scenes, CRAFT has 65 different tasks. CRAFT bias minimization scripts take these 650 pairs and prune video-question tuples according to the least observed answer according to the possible values inside the answer set (true or false for boolean questions). Our aim is to make it difficult for algorithms to reach high performances by simply recognizing the simulation identifier without understanding the question text or the task identifier without extracting any meaning from the visuals. CRAFT enforces models to extract simulation dynamics inside the videos. While generating pairs with uniform answer distributions, our

Question: *"Does the small yellow square prevent the tiny brown circle from entering the basket?"*

```
Var AffectorObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Yellow"), "Cube" )
Var PatientObject = FilterShape ( FilterColor ( FilterSize ( SceneAtStart(), "Small" ) , "Brown"), "Circle" )
Exist (
        FilterMoving (
                Intersect (
                        Difference (
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                GetCounterfactEvents (
                                                        AffectorObject
                                                )
                                        )
                                ),
                                FilterObjectsFromEvents (
                                        FilterEnterBasket (
                                                Events()
                                        )
                                )
                        ),
                        AsList ( PatientObject )
                ),
                StartSceneStep()
        )
)
```

Figure 4.7. Example program for *prevent* questions.

scripts also try to preserve overall dataset distribution as uniform as possible. This can be depicted in Figure 4.8. which shows answer distributions of each task category and answer type for overall dataset.

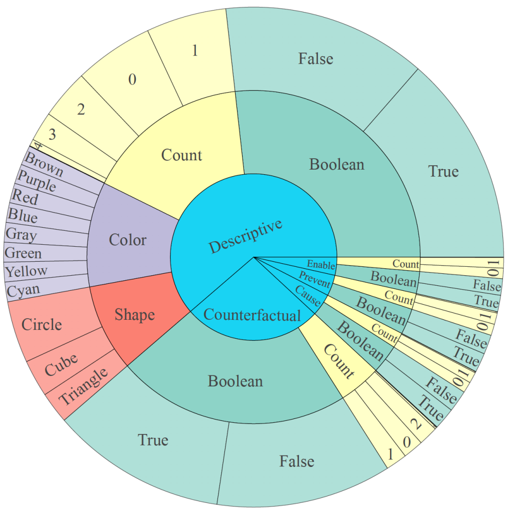


Figure 4.8. Statistics of the questions in CRAFT dataset. Innermost layer represents the distribution of the questions for different task categories. Middle layer illustrates the distribution of the answer types for each task category. Outermost layer represents the distribution of answers for each answer type.

5. EXPERIMENTAL ANALYSIS

In this chapter, we provide a detailed analysis on the experiments conducted on CRAFT dataset. We first introduce the baseline models, and then discuss their results on our dataset.

5.1. Baselines

In this section, we introduce the baseline models for CRAFT [2]. In addition to these artificial baselines, we also conducted a small human study [3] in order to test the claim that CRAFT is designed to be simple for humans, yet challenging for computers. Below, we share the details of all created baselines.

MFA: This is a simple model which finds the most frequent answer (MFA) in the training split of the dataset, and then outputs this answer for all the questions it encounters.

AT-MFA: This is similar to MFA except the fact that it is provided which type of answer is required to correctly solve the question. It then outputs the most frequent answer in the training set using this information. For example, it outputs only a single numeric answer when it encounters a counting question.

LSTM: This is a simple image-blind neural baseline that is trained using CRAFT questions. Background information about LSTM model is provided in section Deep Learning Backbones. It encodes the question by using 256 hidden units and initializing word-vector embeddings randomly. Final question representation is constructed obtaining the last hidden state of the network by processing each individual word sequentially.

LSTM-CNN: This model integrates both visual and textual cues in the training set. It uses previous LSTM model to represent the question. It encodes some of the selected frame(s)

[2] İlker KESEN implemented MFA, AT-MFA, LSTM, and LSTM-CNN baselines and provided the results for the experiments conducted on CRAFT using these baselines.

[3] Mert KOBAŞ prepared web interface for CRAFT human study and provided valuable feedbacks for generated scenes and simulations.

using the output of fourth convolutional layer in ResNet-18 model which is trained on ImageNet 2012 dataset without freezing it. The model then concatenates visual and textual features to represent the video and the question pair and provides it to a linear layer followed by a *tanh* activation function. Furthermore, a dropout with a probability of 0.2 is used for both visual and textual representations.

Adam optimizer [62] is used for the last two simple neural baselines with a learning rate of 0.0001. These two models are also trained for 30 epochs, and the best epoch is selected by comparing the validation scores of each epoch.

MAC: Memory, Attention, and Composition (MAC) network [27] which is designed to facilitate explicit and expressive machine reasoning. Originally, it is created to solve a single task, CLEVR. The model separates memory and control to be able to perform single universal reasoning operation. When it was first proposed, it set new state-of-the-art results for all CLEVR tasks.

Figure 5.1. demonstrates the overview of the MAC network. It has an input unit which extracts representations from image (knowledge base) and the question. Moreover, it has an output unit which predicts the final answer to the question using question representation and the final memory state calculated by MAC recurrent network. MAC recurrent network consists of **p** MAC cells and it iteratively calculates memory and control states. A MAC cell is a recurrent cell which is designed to capture the notion of an atomic universal reasoning. It consist of control, read, and write units. The control unit specifies the reasoning operation by attending the some part of the question and updating the control state. The read unit retrieves information from knowledge base (image) that is required to perform a specific reasoning operation. The write unit updates the memory state (intermediate result) for the cell by integrating the information retrieved from the read unit with previous memory state guided by current control state.

Since a single image is used to feed the MAC network, a CRAFT video is represented by Resnet-18 features of the first or the last frame similar to LSTM-CNN baseline. We trained

MAC using Adam optimizer with a learning rate of 0.0003 for 100 epochs with a batch size of 32.

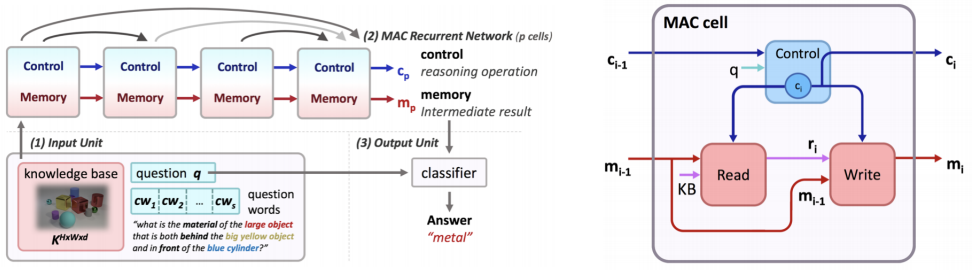


Figure 5.1. **Left:** Memory, Attention, and Composition model overview. **Right:** Single MAC cell architecture. (image taken from [27]).

MAC-V: This model is a video extension of MAC network. Original MAC model represents images by applying two convolutional layers with d output channels to the result of *conv4* features from ResNet101 [23]. Instead of extending the representation in the spatial domain, MAC-V uses corresponding dimension in order to represent videos with a 3D ResNet [50]. Other than this difference in the representation of visual input, MAC and MAC-V models are identical. Input videos to MAC-V are sampled at 1 frame per second. We trained it for 40 epochs using a batch size of 24. During training, Adam optimizer with a learning rate of 0.0003 is used.

G-SWM: This model is actually an unsupervised learning algorithm for object-centric state representation [28]. Originally, it is designed for future state simulation for environments consisting of multiple dynamic objects. It is one of the recent generative models which consider temporal imagination. The model assumes that each frame in a video can be modeled using two different latent variables which are for *objects* and *contexts*. *Contexts* variable represents everything which is non-object related.

G-SWM model has four different submodules which are discovery module, context module, propagation module, and rendering module, respectively. Except discovery module, which is used to detect new objects in the simulation, a summary for each submodule is provided in Figure 5.2. The main aim is to learn context and objects latent variables which also include some randomness to represent the uncertainty. Object representations are then used to

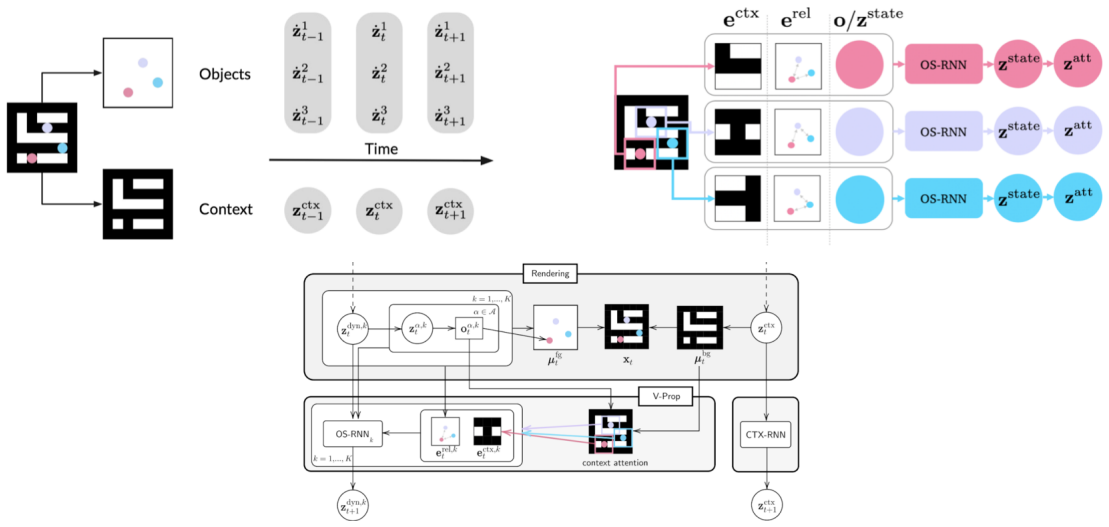


Figure 5.2. **Upper Left:** Objects and Contexts. **Upper Right:** Versatile Propagation. **Lower Middle:** Object-Centric Generation. (image taken from [28])

calculate the exact values of object properties such as *appearance*, *position*, *presence*, and *depth* to create foreground frame. Furthermore, background frame is encoded using context latent variable. These two are then combined to obtain a single frame by the rendering module. Moreover, object specific representations are estimated by using RNNs considering object-object and object-context relationships by propagation module. Finally, context module is also implemented with an RNN which assumes dynamic context for generality (CRAFT contexts are static).

G-SWM is given the initial 10 frames and requested to predict the remaining frames for 100 frame videos in the original experiments that authors conducted. Instead of simulating possible futures in an unsupervised setting, we adapt G-SWM to solve CRAFT classification task (VQA task) by providing whole CRAFT videos sampled at 5 frames per second. We also resize input CRAFT videos so that they have 64 by 64 pixel resolutions as in the original design of the model. This version of G-SWM concatenates *context* and *objects* features to obtain a single representation for a video. It then concatenates this representation with the LSTM question representaion similar to LSTM-CNN model to provide an answer to the question at hand. We trained G-SWM 100 epochs using a batch size of 24 with Adam optimizer and a learning rate of 0.0001.

Table 5.1. Performances of baselines mentioned in Section 5.1. on the validation and the test splits using average accuracy metric are reported. C, CF, D, E and P columns stand for *Cause*, *Counterfactual*, *Descriptive*, *Enable* and *Prevent* tasks, respectively.

Baseline	Input	Validation	Test					
			C	CF	D	E	P	All
MFA	Question	27.86	30.50	42.36	21.54	27.31	27.54	28.00
AT-MFA	Question	41.29	45.60	47.11	38.14	47.60	44.59	41.48
LSTM	Question	44.76	53.77	52.99	39.16	**52.77**	**55.08**	44.65
LSTM-CNN	Question + First Frame	47.68	44.34	50.34	46.32	52.03	53.77	47.83
LSTM-CNN	Question + Last Frame	**53.34**	50.63	**56.71**	**53.99**	52.40	51.48	**54.42**
MAC	Question + First Frame	35.69	39.31	45.49	31.9	29.52	32.13	35.81
MAC	Question + Last Frame	33.52	32.08	36.29	32.9	31.37	33.11	33.74
MAC-V	Question + Video	31.42	32.39	43.0	26.14	30.63	27.54	31.18
G-SWM	Question + Video	47.72	**55.35**	54.65	42.96	50.92	49.18	47.18
Human	Question + Video	–	66.67	74.07	93.01	59.09	90.9	85.89

Human: In order to support our thesis stating that CRAFT is designed to be easy for humans, but difficult for machines, we also conducted a small human study. In this study, we asked 522 randomly selected CRAFT questions to 12 adults whose native languages were Turkish. We divided these questions into 5 different parts and asked participants to choose one or more of them and complete the ones selected. As well as answering the questions, the participants were allowed to state that the question was not clear enough to understand. From 522 questions, responds to 489 questions were recorded.

5.2. Results

This section demonstrates the results obtained by the baselines mentioned in the previous section. Accuracy metric is used in the evaluations. During training, corresponding model parameters are selected from the epoch that achieves highest performance on the validation split. Table 5.1. shows the performances of each baseline on the validation and the test splits. Task-specific performances of each baseline on the test split is also given in Table 5.1. Input column is provided to state the type of inputs provided to the baseline models.

As can be depicted from the results in Table 5.1., there is a large gap ($> 30\%$) between human subjects and neural baselines. However, we should say that humans had difficulties while solving *Cause* and *Enable* questions. We suspect that one of the reasons why these are found to be the most difficult categories according to the human study results can be attributed to the fact that the difference between them (the definition of intention) is not clearly specified to human subjects before the experiment in order to realize a fair comparison between humans and neural baselines. We believe that if it had been specified, the gap between them would have been even much larger.

Performance differences between the validation split and the test split are so small ($< 2\%$) for all baselines that are used in our experiments, demonstrating that CRAFT's validation split is a good representative for its test split. Furthermore, our results show that integrating only the question features increases the performance by a small margin (3.17%) compared to AT-MFA model. This shows that in order to achieve higher scores when solving CRAFT, visual features must be integrated with questions. Although using first video frame features with textual features together decreases the performances for *Cause, Counterfactual, Enable, and Prevent* categories, it increases the overall performance because the amount of *Descriptive* questions is much larger compared to other categories (Figure 4.8.). We think that using the features of the first video frame improves the performance in *Descriptive* questions simply because it lets extracting information about different properties such as colors, sizes etc. from objects which are not specified in the text. On the other hand, the reason why using first frame features confuses the model in other categories is not very clear. Moreover, this decrease is not equivalently worrisome for the model that employs last video frame along with textual features. This is somewhat expected since there are CRAFT questions which require grasping final state of the corresponding scene. This is also the reason why the performance in *Descriptive* questions is increased more. Below, we share such example questions from CRAFT.

- *"How many **S**s enter the basket?"*
- *"How many **S**s fall to the ground?"*

Table 5.2. Performance comparisons of LSTM-CNN models using a simple CNN and Resnet-18. C, CF, D, E and P columns stand for *Cause*, *Counterfactual*, *Descriptive*, *Enable* and *Prevent* tasks, respectively.

Baseline	Input	Validation	Test					
			C	CF	D	E	P	All
LSTM-CNN (Simple CNN)	Question + First Frame	46.11	50.00	52.01	43.84	45.76	52.46	46.73
LSTM-CNN (Simple CNN)	Question + Last Frame	48.67	48.74	**57.20**	44.95	**53.87**	**53.77**	49.10
LSTM-CNN (Resnet-18)	Question + First Frame	47.68	44.34	50.34	46.32	52.03	**53.77**	47.83
LSTM-CNN (Resnet-18)	Question + Last Frame	**53.34**	**50.63**	56.71	**53.99**	52.40	51.48	**54.42**

While designing LSTM-CNN models, our aim was to specify a lower bound for the performances of other models. To be more simplistic, our initial attempts in these models used a simple 2-dimensional Convolution Neural Network (CNN) instead of ResNet-18. The performance comparison of using this simple CNN and Resnet-18 models is presented in Table 5.2. Despite the fact that there are different winners for different categories, the overall performance is increased when using Resnet-18 in our models. Therefore, we have selected models integrating Resnet-18 as our neural baselines in our comparisons in Table 5.1.

Another point that is worth mentioning here is that despite being complex models receiving multiple frames as inputs, MAC-V and G-SWM did not perform well on CRAFT questions. We believe that this is mostly because these models are not inherently created to solve CRAFT tasks. For instance, the original MAC model which achieves the state-of-the-art results on CLEVR, greatly takes advantage of spatial attentions to be able to obtain spatial reasoning capabilities. Converting this type of attention to temporal attention to represent a video as in MAC-V did not result in high accuracies. In order to compare extending features in the spatial domain and in the temporal domain, we also conducted experiments with MAC network using the first or the last frame features only. Although these models outperform MAC-V, the results are quite low compared to LSTM-CNN baselines. Furthermore, another reason why MAC-V fails might be that it is not an object-centric model similar to previous baselines. We believe that a model should be able to identify different objects and their relationships in order to be more successful when solving CRAFT questions. A model, such as MAC-V, can easily be transformed to an object centric model by integrating object mask representations as in [43]. This can be smoothly achieved by concatenating features arriving

from video frames and segmentation masks of all objects. We did not have the chance to experiment with such models since current version of CRAFT does not include object segmentation masks. Finally, MAC-V model sometimes fails to specify even the correct answer type for a question as can be seen from Figure 5.10.

Similarly, G-SWM is designed to learn temporally local relationships in order to predict future frames with a generative loss. Despite being an object specific model, using features from full video to train G-SWM for a classification task (CRAFT), did not also result in high performance. Since its original aim is to generate, it receives full frames instead of using strong frame representations (ResNet-18) as in the case for LSTM-CNN model. It would be very interesting to train a version of G-SWM receiving ResNet-18 features in order to see whether its performance on CRAFT questions increases or not. Furthermore, another important disadvantage for this version of G-SWM is that it resizes the input video resolution from 256 by 256 pixels to 64 by 64 pixels which could lead huge amount of information loss which cannot be neglected.

Our results on CRAFT demonstrate that special attention must be given when designing a model which has strong causal and temporal reasoning, and intuitive physics capabilities while having abilities to visualize counterfactual situations specified in the question.

Here, we provide some qualitative results provided by the experiments conducted on our baselines. Firstly, we compare the results obtained by our LSTM, LSTM-CNN (First Frame), and LSTM-CNN (Last Frame) models on some of CRAFT video question pairs in Figures 5.3. and 5.4., respectively. Furthermore, Figures from 5.5., 5.6., 5.7., 5.8., 5.9., 5.10., 5.11. to 5.12. show correct and wrong predictions produced by other baselines.

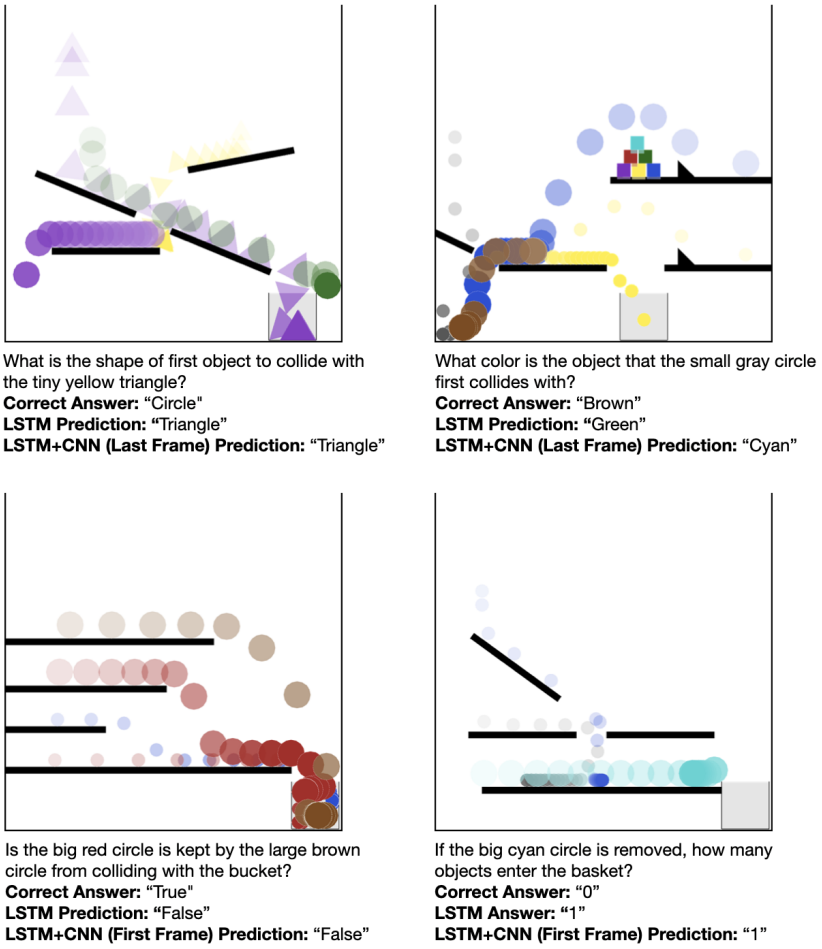


Figure 5.3. Example model predictions. **Upper:** The cases that LSTM-CNN (First Frame) can correctly find the answer, whereas LSTM and LSTM-CNN (Last Frame) cannot. **Lower:** The cases that LSTM-CNN (Last Frame) can correctly find the answer, whereas LSTM and LSTM-CNN (First Frame) cannot.

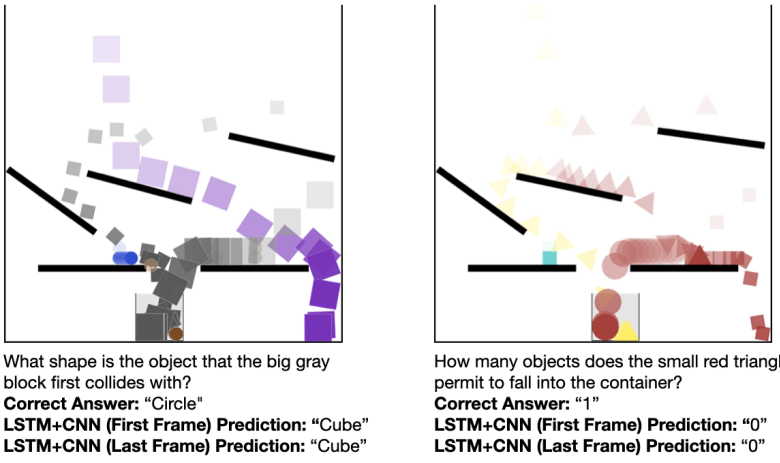
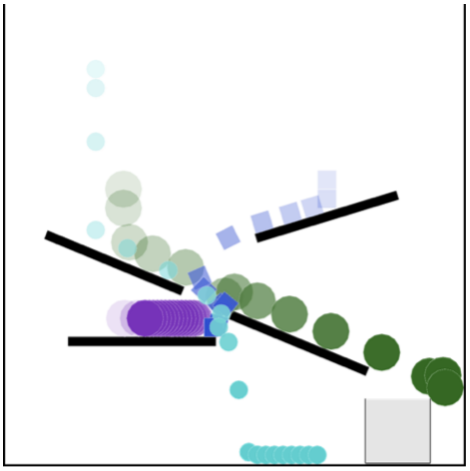
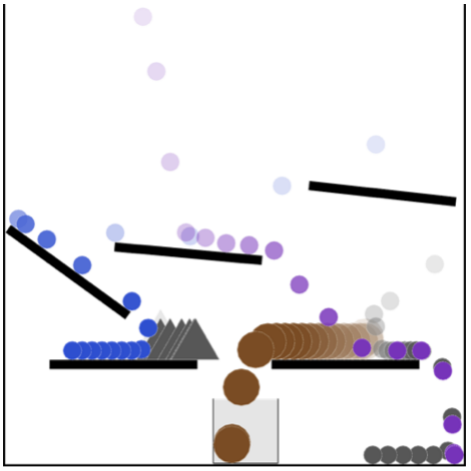


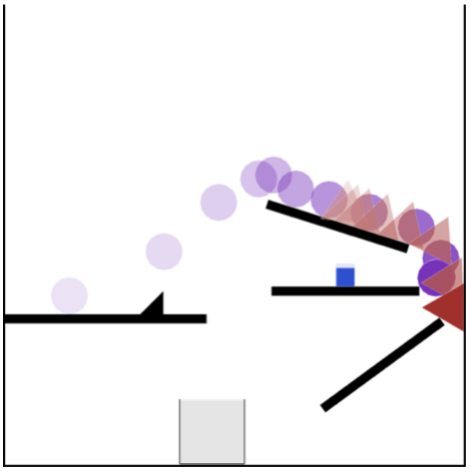
Figure 5.4. Example model predictions showing the cases that LSTM can correctly find the answer whereas LSTM-CNN (First Frame) and LSTM-CNN (Last Frame) cannot.



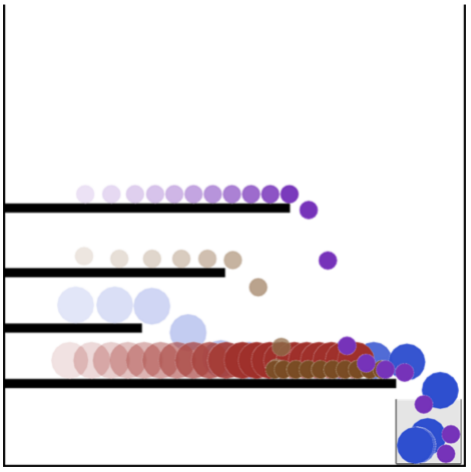
There is a big purple circle, does it lead the large green circle to collide with the container?
Correct Answer: "False"



If any other single one of the objects is removed, does the small purple circle get into the bucket?
Correct Answer: "False"

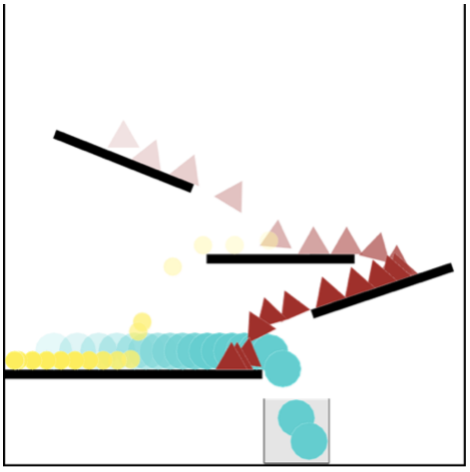


How many green objects hit the ground?
Correct Answer: "0"

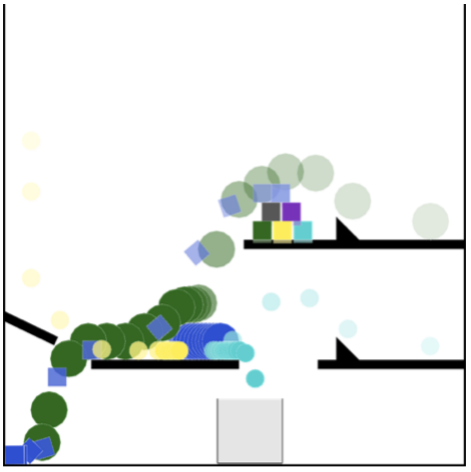


How many objects enter the basket?
Correct Answer: "2"

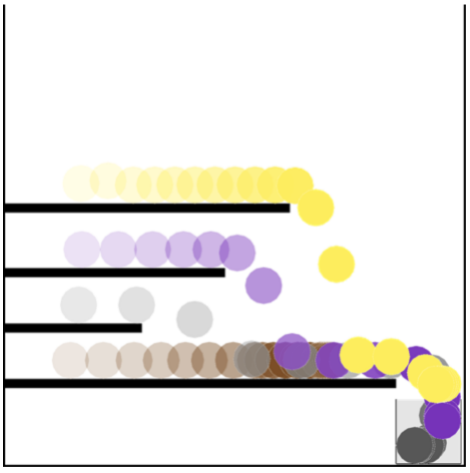
Figure 5.5. Example correct MAC (First Frame) predictions.



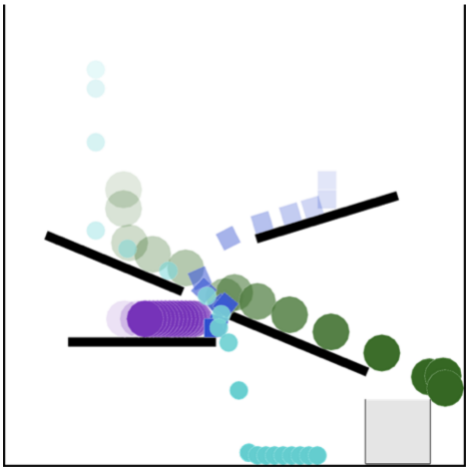
Does the big cyan circle go into the container, if any other single one of the objects is removed?
Correct Answer: "True"
Model Prediction: "False"



How many yellow objects end up in the basket?
Correct Answer: "0"
Model Prediction: "1"

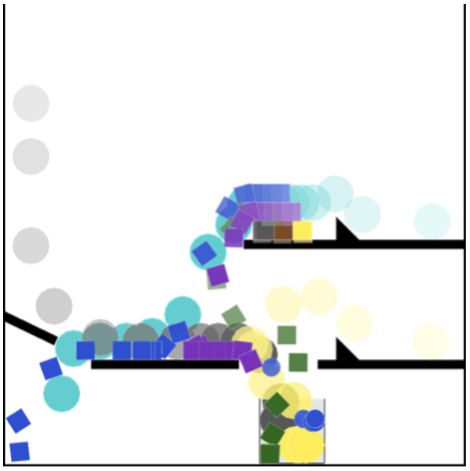


What color is the last object to collide with the large gray circle?
Correct Answer: "Purple"
Model Prediction: "Circle"

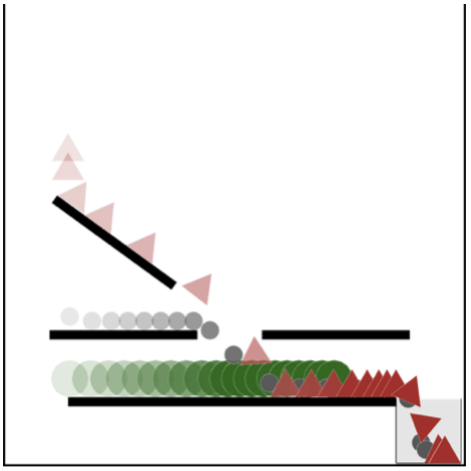


Are there any collisions between objects before the small cyan circle collides with the bottom?
Correct Answer: "True"
Model Prediction: "False"

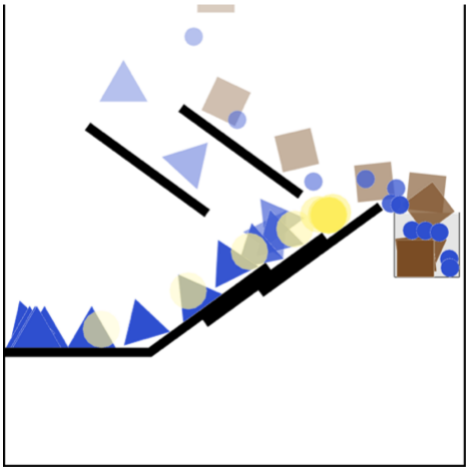
Figure 5.6. Example wrong MAC (First Frame) predictions.



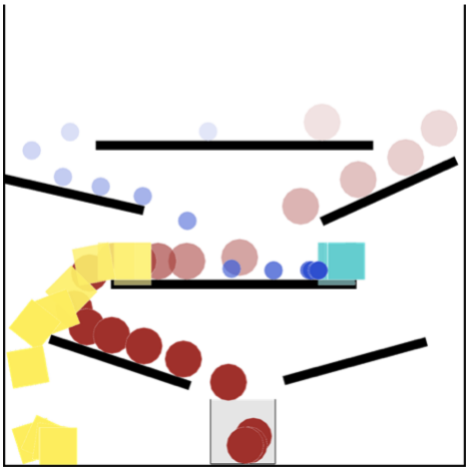
After colliding with the container, does the small blue circle collide with other objects?
Correct Answer: "False"



How many objects does the large green circle permit to collide with the bucket?
Correct Answer: "1"

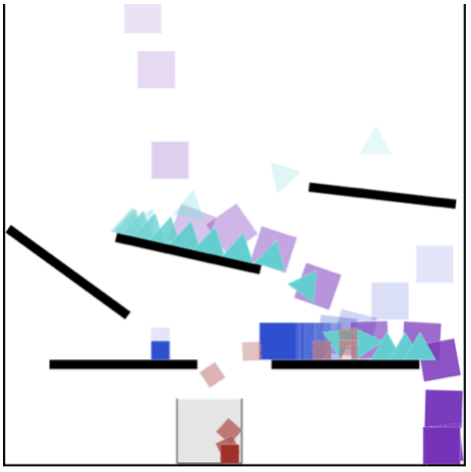


After ending up in the container, does the large brown block collide with other objects?
Correct Answer: "True"

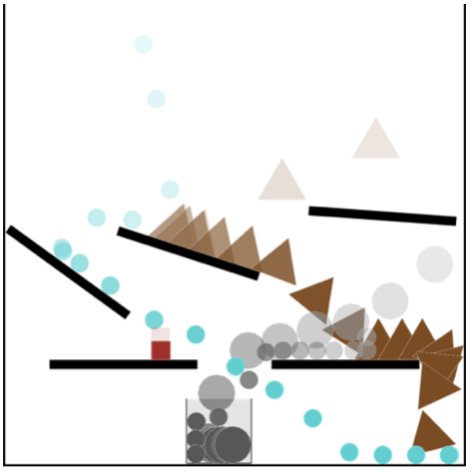


What shape is the object that the large red circle first collides with?
Correct Answer: "Cube"

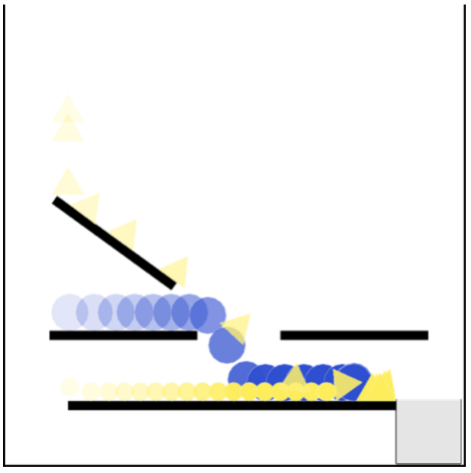
Figure 5.7. Example correct MAC (Last Frame) predictions.



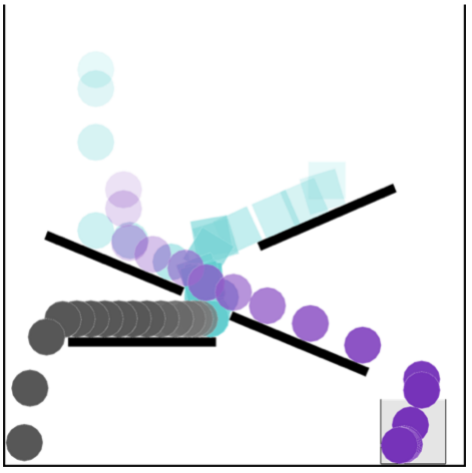
There is a big blue cube, does it lead the tiny red cube to enter the container?
Correct Answer: "True"
Model Prediction: "0"



What is the shape of object that the tiny cyan circle last collides with?
Correct Answer: "Triangle"
Model Prediction: "Circle"

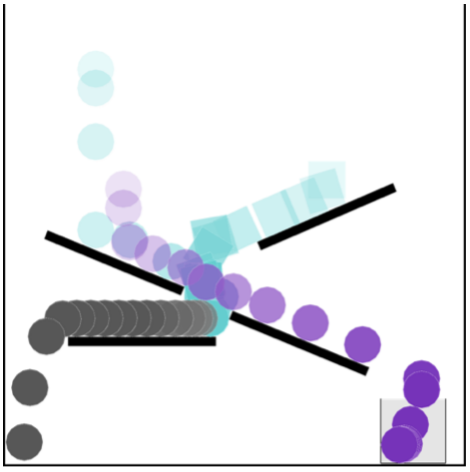


If the big blue circle is removed, how many objects end up in the bucket?
Correct Answer: "0"
Model Prediction: "1"

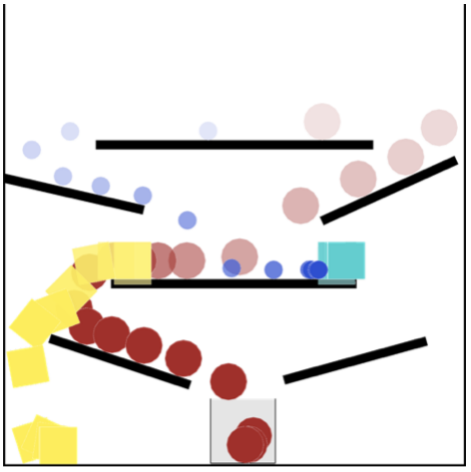


How many objects does the large purple circle allow to fall to the ground?
Correct Answer: "1"
Model Prediction: "0"

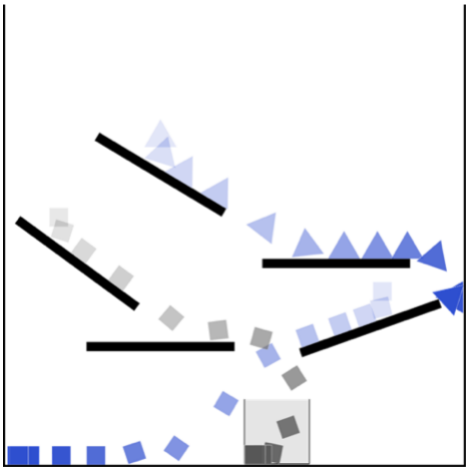
Figure 5.8. Example wrong MAC (Last Frame) predictions.



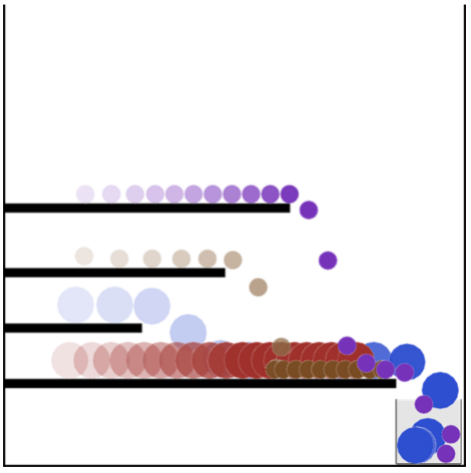
If the big cyan block is removed, does the large purple circle enter the basket?
Correct Answer: “True"



How many objects are kept by the small blue circle from entering the container?
Correct Answer: “0”

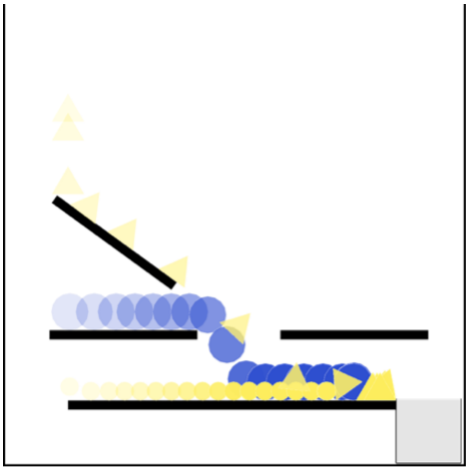


Are there any collisions between objects after the tiny gray block collides with the basket?
Correct Answer: “False”

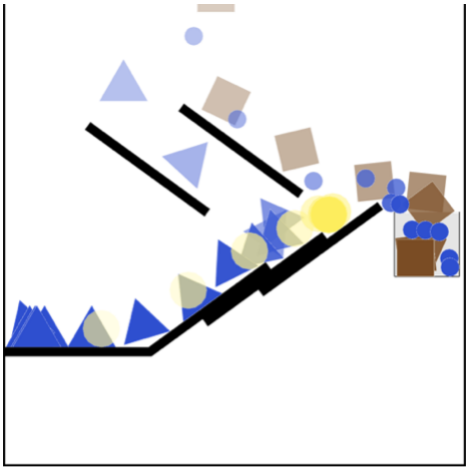


How many objects enter the basket?
Correct Answer: “2”

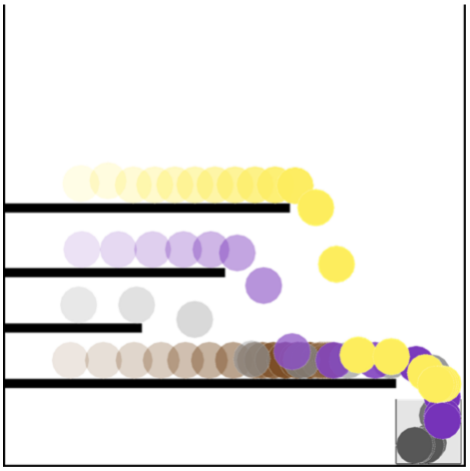
Figure 5.9. Example correct MAC-V predictions.



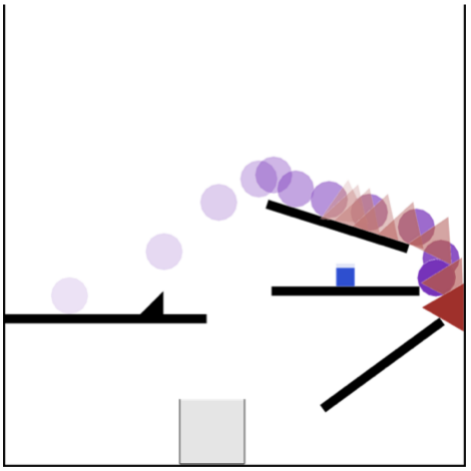
If the large blue circle is removed, does the large blue circle collide with the bucket?
Correct Answer: “False”
Model Prediction: “True”



How many objects enter the bucket?
Correct Answer: “2”
Model Prediction: “0”

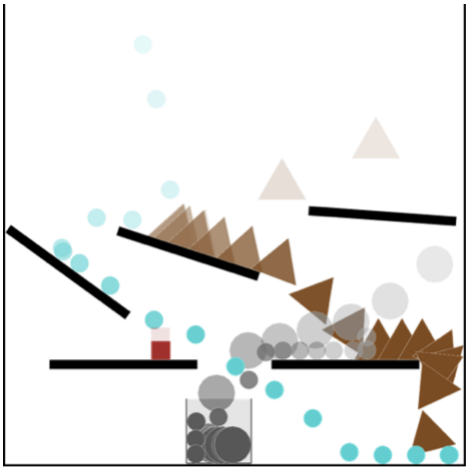


What is the color of first object to collide with the big yellow circle?
Correct Answer: “Brown”
Model Prediction: “True”

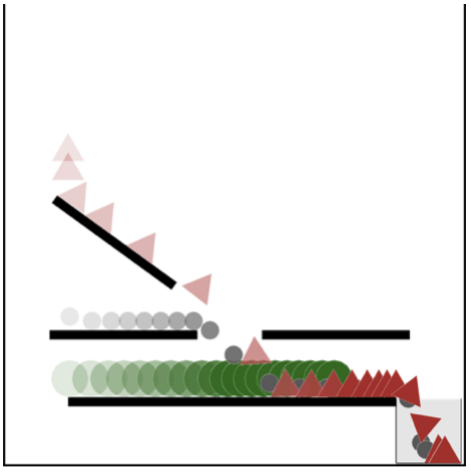


If any other single one of the objects is removed, does the big red triangle collide with the container?
Correct Answer: “False”
Model Prediction: “True”

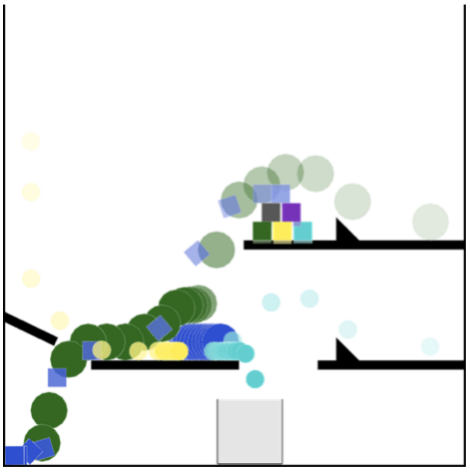
Figure 5.10. Example wrong MAC-V predictions.



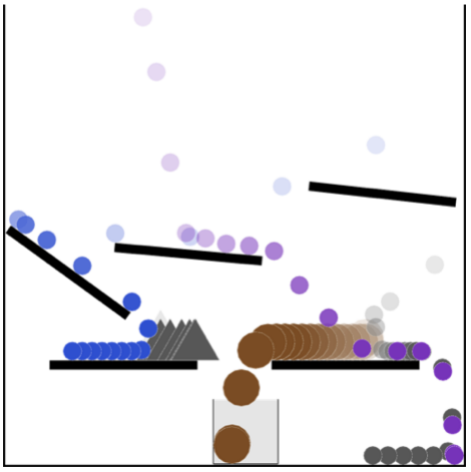
How many circles collide with the bucket?
Correct Answer: “3”



How many objects does the large green circle permit to collide with the bucket?
Correct Answer: “1”

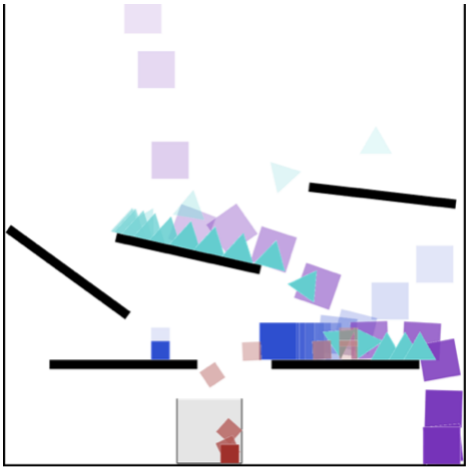


Does the large blue circle allow the tiny cyan circle to enter the bucket?
Correct Answer: “True”

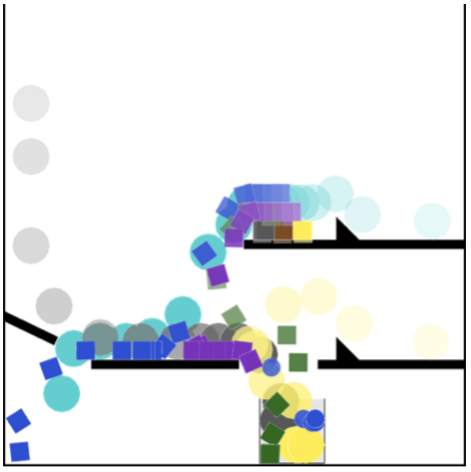


How many objects are held by the big brown circle from colliding with the bucket?
Correct Answer: “1”

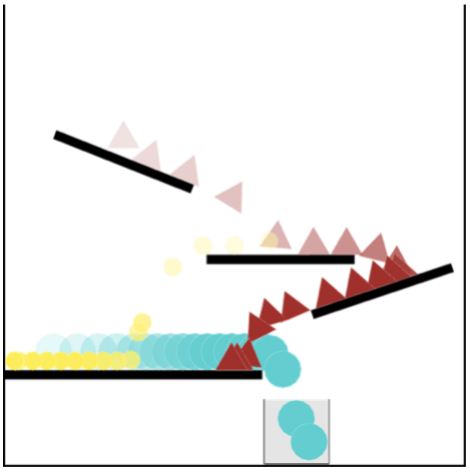
Figure 5.11. Example correct G-SWM predictions.



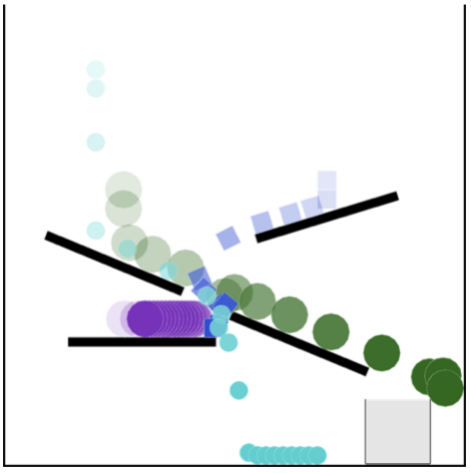
There is a big blue cube, does it lead the tiny red cube to enter the container?
Correct Answer: “True”
Model Prediction: “False”



How many objects collide with the bucket after the small blue circle collide with the bucket?
Correct Answer: “4”
Model Prediction: “1”



Are there any collisions between objects before the large cyan circle collides with the bucket?
Correct Answer: “True”
Model Prediction: “False”



How many objects get into the container, if the big green circle is removed?
Correct Answer: “0”
Model Prediction: “1”

Figure 5.12. Example wrong G-SWM predictions.

6. CONCLUSION

In this thesis, we have presented a new visual question answering dataset, CRAFT, which is designed to challenge causal and temporal reasoning with strong physics understanding capabilities of current machine learning algorithms. CRAFT task design is mostly inspired by the representations of different causal relationships in cognitive science. The tasks require visual understanding of the force dynamics between different entities and grasping natural language input to detect exact causal relationships to be extracted. We believe that studying these two modalities in the causal reasoning context is crucial for artificially intelligent agents as in human cognition research.

Besides providing detailed information about how this dataset is constructed, we also share information about the baselines that we use in the experiments along with the results they obtain. Our baselines consist of models searching frequent answers in the training set, simple neural models which investigate text or text+frame features, and more complex models (MAC-V and G-SWM) which are proposed to solve other tasks in the literature. As can be seen our results, MAC-V and G-SWM do not perform well on our CRAFT tasks. This may be due to the fact that these models are not originally created to solve CRAFT tasks. While MAC benefits from spatial attention a lot in CLEVR task, converting it to temporal attention to use video features as in MAC-V did not lead high performances. On the other hand, while the original G-SWM is a generative model using a short local region for a single frame, providing G-SWM a full video and training it for a classification task also did not lead high performances. These results of our baselines can be considered as a starting point for more novel architectures because of the fact that although the challenges seem intuitive for humans, they can be quite difficult for the machines. This is also demonstrated by the gap ($> 30\%$) between the humans and our most successful neural baseline. As a future work, both an extensive performance analysis of different artificial models and a more detailed human study may be conducted in order to fully understand the differences between human intelligence and artificial intelligence.

There are some extensions for CRAFT dataset that we would like to consider as a future work. Firstly, object segmentation masks for each video can be extracted by our simulator and shared for further use. Secondly, our programs of tasks depend only on the end results of the simulations to be able to provide correct answers to the questions. Our programs do not consider local temporal attempts of the objects whether they are trying to *cause*, *enable*, or *prevent* in a small time interval. Furthermore, there can be multiple patients in our *cause*, *enable*, and *prevent* tasks. Investigating tasks including multiple affectors would be interesting. Moreover, we consider events of entering the basket, but our objects are not able get out of the basket. Allowing such objects would increase the variety in the scenes and in the questions. In addition to these, material textures as in CLEVR can be used to emphasize that the objects in the scene have different densities. Not only density, but also other static properties, such as friction, can also integrated if visually possible.

We believe that by providing a new benchmark, we also propose a new research direction that will consist of different algorithms solving CRAFT, or different datasets which are extensions of it.

REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, **2017**.

[2] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164. **2017**.

[3] Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, **2017**.

[4] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788. **2016**.

[5] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597. **2018**.

[6] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, **2019**.

[7] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 280–287. **2014**.

[8] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation

methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–732. **2016**.

[9] Fahad Lateef and Yassine Ruichek. Survey on semantic segmentation using deep learning techniques. *Neurocomputing*, 338:321–348, **2019**.

[10] Gaurav Sharma, Frédéric Jurie, and Cordelia Schmid. Expanded parts model for human attribute and action recognition in still images. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–659. **2013**.

[11] Lu Liu, Robby T Tan, and Shaodi You. Loss guided activation for action recognition in still images. In *Asian Conference on Computer Vision*, pages 152–167. Springer, **2018**.

[12] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558. **2013**.

[13] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308. **2017**.

[14] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, **2017**.

[15] Nima Fazeli, Miquel Oller, Jiajun Wu, Zheng Wu, Joshua B Tenenbaum, and Alberto Rodriguez. See, feel, act: Hierarchical learning for complex manipulation skills with multisensory fusion. *Science Robotics*, 4(26), **2019**.

[16] Roozbeh Mottaghi, Hessam Bagherinezhad, Mohammad Rastegari, and Ali Farhadi. Newtonian scene understanding: Unfolding the dynamics of objects in static images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3521–3529. **2016**.

[17] Adam Lerer, Sam Gross, and Rob Fergus. Learning physical intuition of block towers by example. *arXiv preprint arXiv:1603.01312*, **2016**.

[18] Michael Janner, Sergey Levine, William T Freeman, Joshua B Tenenbaum, Chelsea Finn, and Jiajun Wu. Reasoning about physical interactions with object-oriented prediction and planning. *arXiv preprint arXiv:1812.10972*, **2018**.

[19] Yunzhu Li, Jiajun Wu, Russ Tedrake, Joshua B Tenenbaum, and Antonio Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv preprint arXiv:1810.01566*, **2018**.

[20] Erin Catto. Box2d v2.0.1 user manual. **2010**.

[21] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910. **2017**.

[22] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433. **2015**.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. **2016**.

[24] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, **2018**.

[25] Damian Mrowca, Chengxu Zhuang, Elias Wang, Nick Haber, Li F Fei-Fei, Josh Tenenbaum, and Daniel L Yamins. Flexible neural representation for physics

prediction. In *Advances in Neural Information Processing Systems*, pages 8799–8810. **2018**.

[26] Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. Phyre: A new benchmark for physical reasoning. In *Advances in Neural Information Processing Systems*, pages 5082–5093. **2019**.

[27] Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067*, **2018**.

[28] Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *International Conference on Machine Learning*, pages 6140–6149. PMLR, **2020**.

[29] James R Kubricht, Keith J Holyoak, and Hongjing Lu. Intuitive physics: Current research and controversies. *Trends in cognitive sciences*, 21(10):749–759, **2017**.

[30] Renee Baillargeon. Physical reasoning in infancy. *The cognitive neurosciences*, pages 181–204, **1995**.

[31] Renée Baillargeon. Innate ideas revisited: For a principle of persistence in infants' physical reasoning. *Perspectives on Psychological Science*, 3(1):2–13, **2008**.

[32] Ernő Téglás, Edward Vul, Vittorio Girotto, Michel Gonzalez, Joshua B Tenenbaum, and Luca L Bonatti. Pure reasoning in 12-month-old infants as probabilistic inference. *science*, 332(6033):1054–1059, **2011**.

[33] Peter W Battaglia, Jessica B Hamrick, and Joshua B Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, **2013**.

[34] Cinzia Chiandetti and Giorgio Vallortigara. Intuitive physical reasoning about occluded objects by inexperienced chicks. *Proceedings of the Royal Society B: Biological Sciences*, 278(1718):2621–2627, **2011**.

[35] Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms. *arXiv preprint arXiv:1706.07068*, **2017**.

[36] Tayfun Ates, Muhammed Samil Atesoglu, Cagatay Yigit, Ilker Kesen, Mert Kobas, Erkut Erdem, Aykut Erdem, Tilbe Goksun, and Deniz Yuret. Craft: A benchmark for causal reasoning about forces and interactions. *arXiv preprint arXiv:2012.04293*, **2020**.

[37] Dennis R Proffitt and Mary K Kaiser. Intuitive physics. *Encyclopedia of cognitive science*, **2006**.

[38] Mateusz Malinowski and Mario Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in neural information processing systems*, pages 1682–1690. **2014**.

[39] Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961. **2015**.

[40] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004. **2016**.

[41] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913. **2017**.

[42] Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Yin and yang: Balancing and answering binary visual questions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5014–5022. **2016**.

[43] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, **2019**.

[44] Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv:1606.07356*, **2016**.

[45] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. Tvqa: Localized, compositional video question answering. *arXiv preprint arXiv:1809.01696*, **2018**.

[46] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9127–9134. **2019**.

[47] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. Tvqa+: Spatio-temporal grounding for video question answering. *arXiv preprint arXiv:1904.11574*, **2019**.

[48] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, **2014**.

[49] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, **2009**.

[50] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459. **2018**.

[51] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, **1997**.

[52] Sangeet S Khemlani, Aron K Barbey, and Philip N Johnson-Laird. Causal reasoning with mental models. *Frontiers in human neuroscience*, 8:849, **2014**.

[53] Steven Sloman, Aron K Barbey, and Jared M Hotaling. A causal model theory of the meaning of cause, enable, and prevent. *Cognitive Science*, 33(1):21–50, **2009**.

[54] Phillip Wolff and Aron K Barbey. Causal reasoning with forces. *Frontiers in human neuroscience*, 9:1, **2015**.

[55] Tomer D Ullman, Elizabeth Spelke, Peter Battaglia, and Joshua B Tenenbaum. Mind games: Game engines as an architecture for intuitive physics. *Trends in cognitive sciences*, 21(9):649–665, **2017**.

[56] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. In *Advances in neural information processing systems*, pages 4502–4510. **2016**.

[57] Yunzhu Li, Jiajun Wu, Jun-Yan Zhu, Joshua B Tenenbaum, Antonio Torralba, and Russ Tedrake. Propagation networks for model-based control under partial observation. *arXiv preprint arXiv:1809.11169*, **2018**.

[58] Tian Ye, Xiaolong Wang, James Davidson, and Abhinav Gupta. Interpretable intuitive physics model. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 87–102. **2018**.

[59] Michael Janner, Sergey Levine, William T. Freeman, Joshua B. Tenenbaum, Chelsea Finn, and Jiajun Wu. Reasoning about physical interactions with object-oriented prediction and planning. In *International Conference on Learning Representations*. **2019**.

[60] Misha Wagner, Hector Basevi, Rakshith Shetty, Wenbin Li, Mateusz Malinowski, Mario Fritz, and Ales Leonardis. Answering visual what-if questions: From actions to predicted scene descriptions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0. **2018**.

[61] Yukie Horiba. The role of causal reasoning and language competence in narrative comprehension. *Studies in Second Language Acquisition*, pages 49–81, **1993**.

[62] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, **2014**.