

# Entwicklung einer virtuellen Versuchsumgebung zur experimentellen Untersuchung von Raumorientierung und visueller Aufmerksamkeit

Sascha Jünger, Frederik Beuth, Fred H. Hamker

Technische Universität Chemnitz, Künstliche Intelligenz  
Strasse der Nationen 62, D-09111 Chemnitz, Deutschland  
sascha.juengel@s2010.tu-chemnitz.de

**Zusammenfassung.** Was für Menschen kein Problem ist, stellt für Roboter eine Herausforderung dar: Den visuellen Raum in kürzester Zeit zu erfassen und relevante Informationen daraus abzuleiten. Ein visuelles System erhält so viele sensorische Daten, das eine clevere Fokussierung auf das Wesentliche nötig ist. Diese Fragestellungen werden im Europäischen Projekt Spatial Cognition erforscht, um diesen noch nicht vollständig verstandenen Prozess der Aufmerksamkeit und Raumwahrnehmung in Zukunft besser verstehen zu können. Die hier vorliegende Arbeit beschreibt die Erstellung einer virtuellen Szene mit vier verschiedenen Einzelszenarien für dieses EU Projekt, um verschiedene Aspekte der Raumwahrnehmung erforschen zu können. Der Schwerpunkt liegt dabei auf der Objektlokalisierung, dem Aktualisieren räumlicher Kontextinformationen, dem aufmerksamkeitsbasierten Beobachten mehrerer Objekte und dem Erinnern an eine Position im Raum. Ein virtueller Agent (Kind) führt diese Aufgaben aus, wobei er über die Fähigkeiten verfügt Objekte anzuschauen und zu greifen. Weiterhin wird ihm, unter anderem durch eine implementierte Pfadsuche, die Möglichkeit gegeben sich in seiner virtuellen Umgebung (Kinderzimmer) zu bewegen. Bis in dem Forschungsprojekt Spatial Cognition entwickelte Gehirnmodelle den Agenten steuern, sind die Handlungsabläufe des Agenten vorgegeben, um die unterschiedlichen Szenarien durchführen zu können. Insgesamt liegt nun eine stabile, virtuelle Versuchsumgebung mit dynamischen Schnittstellen vor, die für alle nachfolgenden Forschungen in diesem Bereich verwendet werden kann.

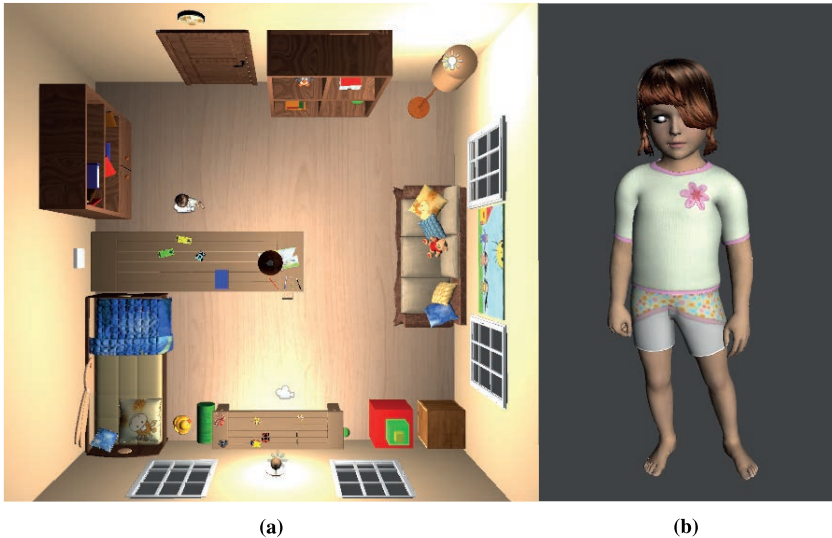
**Stichworte:** Visuelle Aufmerksamkeit · Raumorientierung · Virtuelle Realität

## 1 Einleitung

Für Menschen ist es einfach ein Zielobjekt in einer unbekanntem und unstrukturierten Umgebung zu greifen [Ant14]. Es erfordert allerdings die Implementierung komplexer Fähigkeiten, wie die Identifikation des Ziels im Sichtbereich und das präzise Steuern von Motoren um einen Roboter zu entwerfen, der die selbe Aufgabe ausführen kann. Technologien wie Videoüberwachung und visuell gesteuerte Roboter benötigen flexible visuelle Systeme um die Szene entsprechend der aktuellen Aufgabe zu interpretieren [Ham05]. Es ist naheliegend, dass Aufmerksamkeit eine wichtige Rolle beim Verstehen einer Szene spielt, da sie relevante Informationen priorisiert. Die zugrunde liegenden Prozesse dieser wurden jedoch noch nicht vollständig entdeckt. Was wir allerdings wissen ist, dass wir durch Aufmerksamkeit in der Lage sind, Objekte oder Regionen im visuellen Raum zu überwachen und Informationen aus diesen zu ziehen [CA05]. Klassische Theorien über Aufmerksamkeit nahmen an, dass es ein einzelner ausgewählter Bereich fokussiert wird. Aber viele alltägliche Tätigkeiten wie beispielsweise das Aufpassen auf Kinder im Schwimmbad benötigen Aufmerksamkeit an mehreren Stellen. Und in der Tat haben auch Versuche im Labor gezeigt, dass wir die Fähigkeit haben, vier oder mehr Zielobjekte gleichzeitig zu verfolgen.

Im Rahmen des EU-Projektes Spatial Cognition soll die Orientierung im Raum und das Extrahieren relevanter Informationen aus der Umgebung erforscht werden [spa]. Ziel ist es, herauszufinden, wie das Aktualisieren räumlicher Kontextinformationen funktioniert und durch welche Mechanismen der Überblick über wichtige Umgebungsdetails beibehalten und Informationen aus diesen gewonnen werden. Hierbei hilft vor allem der Versuch diese Prozesse zu simulieren und innerhalb eines Gehirnmodells so weit nachzustellen wie sie bisher erschlossen sind. Ein solches virtuelles Gehirnmodell benötigt dafür jedoch eine virtuelle Versuchsumgebung, in der verschiedene, die visuelle Wahrnehmung untersuchende Experimente durchgeführt werden können. Eine solche Umgebung wurde im Rahmen dieser Arbeit erstellt. Als Basis diente die Unity-Engine, eine flexible Entwicklungsplattform zur Erstellung von 2D und 3D Spielen und anderen interaktiven Anwendungen. Später soll der an der TU Chemnitz entwickelte Neuro-Simulator ANNarchy (Artificial Neural Networks architect) verwendet werden um das Gehirn des Agenten zu simulieren.

Die Grundlagen des bereits vor dieser Arbeit existenten Frameworks zur Kopplung von VR und Agent werden im folgenden Kapitel beschrieben. Kapitel 3 beschäftigt sich mit der Erweiterung dieses Frameworks um diverse neue Funktionen, die Sensorik und Aktorik betreffen. Im nachfolgenden, 4. Kapitel werden verschiedene Problemstellungen der räumlich-visuellen Wahrnehmung erläutert und für jede von diesen wird ein Szenario innerhalb der virtuellen Versuchsumgebung bereitgestellt, um weitere Erkenntnisse auf diesem Gebiet gewinnen zu können. Da aktuell die Kontrolle über den Agenten, der diese Szenarien ausführt, noch nicht von einem Gehirnmodell übernommen wird, werden viele Abläufe vereinfacht, automatisiert oder nachgebildet um die Funktionalität der erstellten Versuchsumgebung zu demonstrieren. Diese Arbeit hat nicht direkt zum Ziel einen tieferen Einblick in die Prozesse der visuellen Wahrnehmung zu erhalten, sondern stellt viel mehr ein grundlegendes Framework zur Verfügung, um dieses Verständnis in zukünftigen Forschungsvorhaben erhalten zu können.



**Abbildung 1:** (a) Die Virtuelle Umgebung, die für die Szenarien genutzt wird. (b) Der virtuelle Agent Felice.

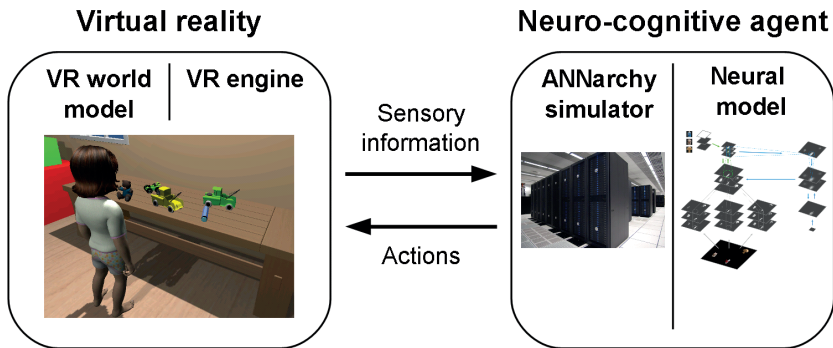
## 2 Simulation eines neuro-kognitiven Agenten in der VR

Die Grundlage der in dieser Arbeit erstellten Szenarien bildet ein bereits in der Vergangenheit an der TU Chemnitz entwickeltes Framework. Der Zugang zum Spatial Cogniton Projekt wird in Zukunft auf der zugehörigen Homepage bereitgestellt (<https://www.tu-chemnitz.de/informatik/KI/projects/agents-vr>). Da der Lernprozess visueller Stimuli hauptsächlich während der ersten Lebensjahre auftritt, wurde Felice einem Kind nachempfunden. Dieses kann sich, unter Berücksichtigung von Kollisionen, in der Szene durch Laufen und Drehen bewegen, sowie Objekte greifen und anschauen. Die virtuelle Umgebung wurde passend zum Agenten gewählt und stellt ein Kinderzimmer da, welches neben anderen Einrichtungsgegenständen zwei Tische beinhaltet, auf denen sich verschiedene Objekte befinden, die in den einzelnen Szenarien von Bedeutung sind (Abb. 1).

Die VR läuft unter Windows und stellt alle sensorischen Daten, wie stereoskopische Bilder und Kollisionen bereit. Kern der in der VR verwendeten Skripte bilden ein Agenten- (*AgentScript*) und ein Umgebungsskript (*BehaviourScript*). Das Agentenskript steuert die Bewegungen des Agenten und verwaltet die Eingangsdaten wie zum Beispiel Kamerabilder. Allerdings wird es nicht direkt an das Agenten-Objekt angehängt, sondern ein von ihm abgeleitetes Skript, das die im ursprünglichen Agentenskript enthaltenen virtuellen Methoden überschreiben kann und so die Möglichkeit hat, das Verhalten jedes Agenten individuell anzupassen. Das Umgebungsskript ist für die Kontrolle der virtuellen Umgebung

zuständig. Sein abgeleitetes Skript wird an ein Master-Objekt angehängt, enthält Referenzen auf die Agenten sowie deren Agentenskripts und initialisiert diese. Neben diesen beiden Haupt-Skripten wurden noch diverse andere Skripte (beispielsweise zum Blenden verschiedener Greifanimationen und der Berechnung von Kollisionen), sowie Unity-Prefabs und die vom Agenten genutzten Animationen durch frühere Projekte bereitgestellt.

Die Agentenseite läuft unter Linux und ist eine C++ Implementierung. Aktuell werden dem Agenten von hier aus lediglich Nachrichten über das Netzwerk zugesendet um in Test-Szenarios die Funktionalität der VR zu testen. Später soll der an der TU Chemnitz entwickelte Neuro-Simulator ANNarchy (Artificial Neural Networks architect) verwendet werden um das Gehirn des Agenten zu simulieren [ann]. Dessen Kern-Bibliothek wurde in C++ geschrieben und kann entweder durch openMP oder CUDA parallelisiert werden. Er stellt außerdem ein Python-Interface für Netzwerke bereit.



**Abbildung 2:** Das Framework um neuro-kognitive Agenten zu simulieren, besteht aus einer virtuellen Realität, in der sich der Agent befindet und dessen Gehirn.

Um die Kommunikation zwischen der VR- und Agentenseite (Abb. 2) zu ermöglichen, werden eine Netzwerkverbindung (TCP/IP) sowie das von Google Inc. entwickelte Datenformat Protocol Buffers (kurz protobuf) verwendet [pro]. Dabei handelt es sich um einen erweiterbaren, sprachen- und plattformübergreifenden Mechanismus zur Serialisierung, ähnlich XML. Genutzt wurde die C++ Implementierung. Protobuf ermöglicht das Erstellen verschiedener Nachrichten, die entweder von VR- zu Agentenseite gesendet werden können oder umgekehrt. Um die Test-Szenarien dieser Arbeit durchführen zu können, wurden einige der bereits in der Vergangenheit implementierten Protobuf-Nachrichten verwendet: *MsgAgentEyeMovement* lässt Felice die Augen bewegen. Als Input werden jeweils die Drehung um die y-Achse (pan) für das linke und rechte Auge sowie die Drehung um die x-Achse (tilt) für beide Augen entgegen genommen. Diese Nachricht wird vor allem dazu verwendet, den Blick des Agenten wieder in die Ausgangsposition (gerade nach vorne) zu richten. Um etwas anzuschauen und zu fixieren, wird die Nachricht *MsgAgentEyeFixation* verwendet, die das Fixations-Ziel in Weltkoordinaten entgegen nimmt. Der Blick des Agenten bleibt dann auf diese Position fixiert (auch wenn dieser sich be-

wegt), bis *MsgAgentEyeMovement* gesendet wurde. An eine Position greifen kann Felice durch die Nachricht *MsgAgentGraspPos*, die Kamerakoordinaten entgegen nimmt und diese dann ins Weltkoordinatensystem abbildet. Ein gegriffenes Objekt kann durch *MsgAgentGraspRelease* wieder losgelassen werden. Um die globale Ausgangssituation eines Szenarios herzustellen wird die Nachricht *MsgEnvironmentReset* verwendet. *MsgTrialReset* ist zum Zurücksetzen von Teilaufgaben gedacht und wird in den Szenarien dieser Arbeit zur Manipulation der Umgebung (beispielsweise dem Ein-/Ausblenden von Objekten) genutzt.

### 3 Erweiterte Sensorik und Aktorik

Im Rahmen dieser Arbeit wurden Sensorik und Aktorik des Agenten erweitert um zusätzliche Informationen zu erhalten bzw. neue Aufgaben durchführen zu können. Weiterhin wurde kleinere Fehler korrigiert.

Eine wichtige Neuerung sind die Augenbewegungen zwischen zwei Fixationsphasen (Sakkaden). Diese wurden entsprechend der Beschreibungen in [BH08] und [VWVO88] implementiert.

$$A = \frac{1}{1 - e^{-\frac{R}{m_0}}} \quad (1)$$

*A*: Skalierungsfaktor  
*R*: Verschiebung auf der Retina  
*m*<sub>0</sub>: konst. (*m*<sub>0</sub> = 7)

$$V_{pk} = V_{max} * (1 - e^{-\frac{\phi}{n}}) + 50; \quad (2)$$

*V*<sub>pk</sub>: höchste Augengeschwindigkeit  
*V*<sub>max</sub>: Maximale Geschwindigkeit (*V*<sub>max</sub> = 750)  
 $\phi$ : Winkel der Augenbewegung ( $0 \leq \phi \leq 60$ ) [deg]  
*n*: konst. (*n* = 15)

$$Ecc = m_0 * \ln \frac{A * e^{\frac{V_{pk} * t}{m_0}}}{1 + A * e^{\frac{V_{pk} * t - R}{m_0}}} \quad (3)$$

*Ecc*: Momentane Augen-Exzentrizität (= S(t) in [VWVO88])

Mit *MsgEyePosition* und *MsgHeadMotion* wurden zwei Nachrichten implementiert, die mit einer Art Gleichgewichtssinn verglichen werden können und fortlaufend Informationen zur Agentenseite schicken können. Erstere sendet die Drehposition und Rotationsgeschwindigkeit der Augen des Agenten, Letztere dessen Bewegungsgeschwindigkeit und -beschleunigung sowie Rotationsgeschwindigkeit und -beschleunigung. *MsgAgentTurn* wird zur VR gesendet und nimmt eine Gradzahl entgegen. Sie gibt Felice die Anweisung, sich um die jeweilige Gradzahl zu drehen (im Uhrzeigersinn bei einem positiven

Wert und gegen den Uhrzeigersinn bei einem negativen Wert). Für die Implementierung dieser Rotation wurde eine Koroutine verwendet um eine kontinuierliche Drehbewegung zu erzeugen [cor]. Diese wird im ersten Fünftel linear beschleunigt, läuft dann mit Maximalgeschwindigkeit ab und bremst im letzten Fünftel wieder linear ab. Zwei weitere neue Nachrichten sind *MsgAgentMoveTo*, die eine Position im Weltkoordinatensystem entgegennimmt und den Agenten (falls möglich) dorthin laufen lässt, sowie *MsgAgentCancelMovement*, die sämtliche Bewegungen (Laufen und Drehen) abbricht.

Um den Weg des Agenten durch die virtuelle Umgebung entsprechend *MsgAgentMoveTo* berechnen zu können, wurde eine A\*-Suche implementiert. Dazu wurde das Kinderzimmer zuerst in ein  $n \times n$  Gitter aufgeteilt, um eine Karte des Raumes zu erhalten. Für jeden der Gitterpunkte, die in regelmäßigem Abstand etwa auf Kniehöhe des Agenten angelegt sind, wurde getestet, ob er innerhalb der Boundingbox irgendeines Objektes in der Szene liegt. Falls ja, befindet sich an diesem Punkt offenbar ein Objekt und er wird als blockiert behandelt. Um eventuellen durch die Engine bedingten Kollisionen vorzubeugen, die entstehen könnten, wenn sich Felice sehr nahe an einem Gegenstand vorbei bewegt, wurden auch alle Nachbarn der blockierten Gitterpunkte als blockiert gekennzeichnet. Damit entstehen später nur Pfade, die ein wenig Abstand zu Hindernissen lassen. Gitterpunkte, die weder innerhalb eine Boundingbox liegen, noch mit einem Gitterpunkt benachbart sind, der das tut, sind frei und vom Agenten begehbar. Die A\*-Suche nimmt nun das erstellte Gitter und behandelt jeden seiner Punkte als Knoten. Start- und Zielpunkt werden dem Algorithmus als Raumpunkte (3D-Vektoren) übergeben und auf den am nächsten liegenden Punkt des Gitters abgebildet. Vom Startpunkt ausgehend werden nun jeweils die Nachbarpunkte betrachtet und anhand ihres Abstandes zum diesem und der Manhattan-Distanz zum Zielpunkt gewichtet. Dieses Verfahren wird nun für den vielversprechendsten der so entdeckten Punkte fortgesetzt (wobei als Distanz zum nächsten Punkt jeweils der gesamte bisherige Pfad gesehen wird), bis man am Zielpunkt angelangt ist.

Der Agent kann sich auf 2 verschiedene Weisen entlang des Pfades bewegen. Die erste orientiert sich direkt am Raumgitter: Mittels Koroutinen dreht sich Felice immer zuerst in Richtung des nächsten Pfadknotens und läuft dann so lange in diese Richtung bis sie diese ändern muss. Ihr erstes Ziel ist dabei der zweite Knoten des berechneten Pfades, da der erste in etwa ihrer Position auf dem Gitter entspricht. Drehung und Vorwärtsbewegung laufen dabei analog zu den Nachrichten *MsgAgentTurn* und *MsgAgentMove* ab. Bei der zweiten Art der Bewegung läuft Felice entlang einer Bezier-Kurve durch das Kinderzimmer, deren Kontrollpunkte durch die Gitterpunkte des Pfades gegeben sind. Die aktuelle Position des Agenten auf der Kurve wird dabei mit Hilfe des Algorithmus von de Casteljau approximiert. Dieser basiert auf einer rekursiven Vereinfachung des Kontrollpolygons, bei der nach und nach die Menge der Kontrollpunkte reduziert wird, bis nur noch ein Punkt übrig ist, der sich dann auf der Bezier-Kurve befindet. Durch eine inkrementelle Erhöhung des Eingabewertes (bereits zurückgelegte Strecke im Bezug auf die Gesamtstrecke) erhält man eine Bewegung entlang der Kurve.

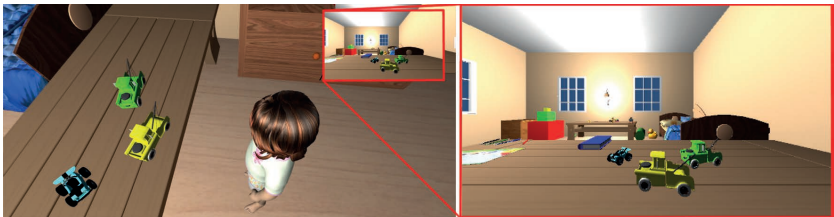
In jedem Szenario wurde die beobachtende Kamera so positioniert, dass sowohl Felice als auch die Objekte, mit denen sie interagiert, in der Szenenansicht zu sehen sind. In der ursprünglichen Szenenansicht waren im oberen rechten Bereich die Kamerabilder vom linken und rechten Auge des Agenten nebeneinander angeordnet. Um mehr von der ei-

gentlichen Szene sehen zu können, wurden diese beiden aufeinander geschoben (Bild der linken Augen-Kamera obenauf). Da die Bilder der Augen beinahe identisch sind, hat das keinen Nachteil für den Beobachtenden. Um die Auswertung der verschiedenen Spatial Cognition Szenarios zu erleichtern, wird bei jedem Start eine Textdatei erstellt, die von allen Objekten mit denen Felice interagiert die Positionen sowohl im Gitter- als auch im Weltkoordinatensystem enthält.

## 4 Spatial Cognition

### 4.1 Object Localization

Objekterkennung ist die Aufgabe ein gegebenes Zielobjekt in einer Szene zu lokalisieren und zu erkennen. Diese Aufgabe wird täglich vom Menschen ausgeführt, beispielsweise wenn man die Fernbedienung vom Wohnzimmertisch aufheben möchte. Psychologen untersuchen dies am Beispiel der gesteuerten visuellen Suche (engl. "guided visual search"), in der ein gegebenes Ziel zwischen verschiedenen Distraktoren (ablenkenden Objekten) gesucht werden muss. Wenn das Ziel sich von den Distraktoren in einer Eigenschaft klar unterscheidet, ist es leicht erkennbar (Pop-Out Effekt). Je ähnlicher das Ziel den Distraktoren ist, umso langsamer ist die Suche.

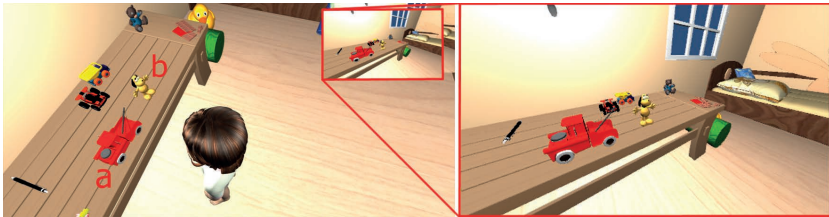


**Abbildung 3:** Felice steht vor dem Tisch auf dem sich die 3 Spielzeugautos befinden, von denen sie 2 lokalisieren soll.

Das Object Localization Szenario besteht aus zwei Objekten, die angeschaut werden sollen und einem Distraktor. Bei allen dreien handelt es sich um Spielzeugautos, die sich farblich unterscheiden, wobei der Distraktor zusätzlich in seiner Form abweicht. Zu Beginn wird Felice vor einem Tisch platziert und schaut geradeaus über diesen hinweg (Abb. 3). Auf dem Tisch befinden sich direkt vor ihr die drei Spielzeugautos (die zu erkennenden in gelb und grün und der Distraktor in hellblau). An dieser Stelle wird in Zukunft ein Gehirnmodell auf Agentenseite die Objekterkennung durchführen und die beiden zu erkennenden Objekte lokalisieren und anschauen. In der Test-Implementierung dieses Szenarios wird diese Lokalisierung als erfolgreich angesehen und die beiden Objekte werden nacheinander vom Agenten betrachtet.

## 4.2 Body Displacement

Im täglichen Leben sind wir uns der Objekte in unserem Umfeld nahezu perfekt bewusst [VPM07]. Auch wenn wir uns bewegen, scheinen wir keine Probleme damit zu haben die Übersicht zu behalten. Dieses scheinbar automatische Verhalten der Aktualisierung räumlicher Kontextinformationen funktioniert sogar in völliger Dunkelheit. Aber trotz umfassender Untersuchungen bleibt die Berechnungsgrundlage eines solchen Aktualisierens ein kontrovers diskutiertes Problem. Zentrale Fragen in den kognitiven Neurowissenschaften der räumlichen Erinnerung beziehen sich auf die Referenzkoordinatensysteme, die genutzt werden um Positionen zu repräsentieren. Beispielsweise ob diese egozentrisch (relativ zu Teilen des Körpers) oder allozentrisch (relativ zur externen Umgebung) sind, die Dauer über die verschiedene Repräsentationen beibehalten werden, wozu sie genutzt werden und wie sie miteinander interagieren. Im Szenario "Body Displacement"-oll versucht werden, den Antworten auf diese Fragen näher zu kommen. Dem virtuellen Agenten wird vor einer Translation seines gesamten Körpers kurzzeitig ein Zielobjekt eingeblendet. Nach der Verschiebung soll er sich an die Position des Zielobjektes erinnern und es anschauen. Die Fehler liefern dabei Hinweise ob die Körper-Translation in einem augenzentrierten und/oder körperzentrierten Koordinatensystem enkodiert ist.



**Abbildung 4:** Felice fixiert den Hund (b), während das rote Auto, dessen Position sich gemerkt werden soll (a), kurz eingeblendet wird.

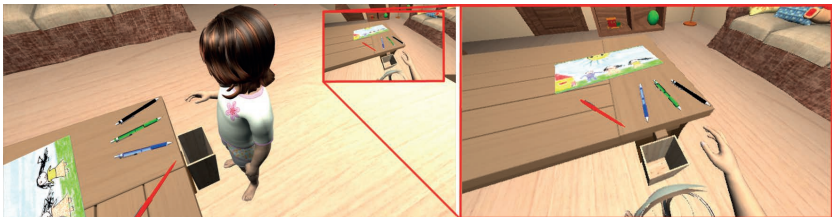
Auch hier wird Felice wieder vor einem Tisch platziert, auf dem sich verschiedene Objekte befinden. Ihr Blick geht gerade nach vorne. Die zwei wichtigen Objekte sind hierbei ein kleiner Spielzeughund sowie ein rotes Spielzeugauto. Letzteres wird allerdings nur kurz eingeblendet. Zu Beginn des Szenarios wird der Blick des Agenten auf einen Fixationspunkt (den Hund) zentriert. Im Anschluss wird kurzzeitig das Zielobjekt (rotes Auto) eingeblendet (Abb. 4), das der Agent in seinem peripheren Sichtfeld wahrnimmt, und Felice danach kontinuierlich vor dem Tisch nach rechts verschoben. In der Zeit in der das Zielobjekt sichtbar ist, soll sich dessen Position gemerkt werden. Dieses Merken und das spätere Erinnern an die Objekt-Position wird in Zukunft von einem Gehirnmodell realisiert. In der Test-Implementierung wird wieder davon ausgegangen, dass sich die Position korrekt gemerkt wurde. Die Seitwärts-Verschiebung des Agenten soll eine externe Krafteinwirkung simulieren durch die er zur Seite geschoben wird (beispielsweise einen Stoß durch eine andere Person). Der Blick von Felice ist während der gesamten Bewegung und auch danach weiterhin auf den Fixationspunkt gerichtet. Anschließend wird die erinnerte Position



des Zielobjekts von ihr angeschaut. Als letzter Schritt wird nun das rote Auto noch einmal eingeblendet, um die Korrektheit zu überprüfen.

### 4.3 Multiple Attention Pointers

Jedes Mal wenn die Augen sich bewegen, um potentiell relevante Objekte in den scharfen Bereich des Sehens zu bringen, muss das visuelle System mit starken Verschiebungen des Bildes auf der Retina umgehen können [Rol11]. Diese retinalen Bildverschiebungen sind nicht bewusst wahrnehmbar. Und noch wichtiger: Wir verlieren nicht die Objekte in der Szene, die von momentanen Interesse sind und eventuell das Ziel zukünftiger Augenbewegungen werden. Die Unfähigkeit Änderungen in unbeaufsichtigten Teilen der Szene zu bemerken, deutet darauf hin, dass Aufmerksamkeit das Verschiebungsproblem nur für eine kleine Anzahl an Positionen löst. Aufmerksamkeitszeiger spezifizieren die Position aktuell beachteter Objekte, ob Ziele folgender Sakkaden oder nicht, und ermöglichen damit sowohl das Planen von Aktionen mit ihnen, als auch die verbesserte Bearbeitung an diesen Positionen. Diese Zeiger auf Ziele von Sakkaden werden durch einen vorhersagbaren Prozess, kurz bevor das Auge anfängt, sich zu bewegen, aktualisiert. Dieser Prozess verschiebt die Aufmerksamkeit, im Gegensatz zur Sakkade, in die Richtung von Positionen, die den aktuellen Zielen weder distal noch proximal entsprechen. Damit wird, bevor die Augenbewegung dort ankommt, die Position vorausgesehen, die die Ziele auf der Retina haben werden, nachdem die Sakkade durchgeführt wurde.



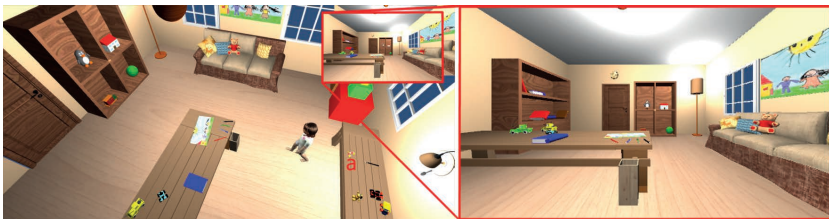
**Abbildung 5:** Felice startet die Greifbewegung in Richtung des blauen Stiftes.

Im Multiple Attention Pointers Szenario geht es darum mehrere Stifte zu greifen und in eine dafür vorgesehene Box zu legen. Währenddessen wird ein anderer Stift im Auge behalten“. Dieser wird dann unvorhergesehen entfernt und Felice soll darauf reagieren. Der Agent wird wie gewohnt in die Ausgangsposition gebracht. Diese ist vor einem Tisch auf dem sich vier verschiedenfarbige Stifte befinden (von links nach rechts: rot, blau, grün, schwarz). Zwischen Felice und dem Tisch auf dem Boden befindet sich eine Sammelbox, in die die gegriffenen Stifte später fallen gelassen werden. Der Blick des Agenten ist anfangs geradeaus über den Tisch hinweg. Nun wird zuerst der Blick auf den blauen Stift gerichtet und dieser dann gegriffen (Abb. 5). Damit die Stifte nach dem Greifen korrekt in der Hand des Agenten liegen, wurden entsprechende lokale Translationen und Rotationen in der Datei *felice\_grasp.csv* vermerkt. Nach Abschluss der Greifbewegung hält Felice

den Stift in ihrer Hand vor sich. Da in der VR-Implementierung bisher keine Animation existiert um gegriffene Objekte an einer gewünschten Stelle wieder abzulegen, besteht die einzige Möglichkeit, den Stift in ihrer Hand loszuwerden, darin, ihn fallen zu lassen. Das ist auch der Grund weshalb sich die Sammelbox auf dem Boden unter der Hand des Agenten befindet und nicht beispielsweise hinter den Stiften auf dem Tisch. Damit die Stifte in der Box liegen bleiben wurden sowohl sie als auch die Box mit entsprechenden Collidern ausgestattet. Nachdem der blaue Stift in die Box fallen gelassen wurde, geht es mit dem zweiten (grünen) Stift weiter. Auch dieser wird wieder, danach gegriffen und schließlich in die Sammelbox fallen gelassen. Als dritter und letzter ist nun der schwarze Stift an der Reihe. Der Agenten richtet erneut seinen Blick auf ihn. Bevor der schwarze Stift allerdings gegriffen werden kann, verschwindet im peripheren Sichtfeld der rote Stift. Dieses einfache Verschwinden soll darstellen, dass beispielsweise eine andere Person den Stift weg nimmt. Am Schluss des Szenarios betrachtet Felice die ehemalige Position des roten Stiftes.

#### 4.4 Vision and Memory

Wir sind oft mit der anspruchsvollen Aufgabe konfrontiert, zu entscheiden, wie wir uns im Angesicht unvollständiger sensorischer Informationen verhalten [BBB07]. Um solche Aufgaben zu lösen, müssen wir uns auf interne Repräsentationen von den Positionen der Objekte in ihrer Umwelt verlassen. Zu verstehen wie eine höhere kognitive Funktion wie die Erinnerung aus der Zusammenarbeit von Neuronen und Synapsen in unserem Gehirn entsteht ist eine der faszinierendsten Herausforderungen kognitiver Neurowissenschaften. Ein solches Verständnis würde als Brücke zwischen den neurophysiologischen und verhaltensbasierten Beschreibungsebenen der Neurowissenschaften dienen und es erlauben, die Lösung von Fragen von einem Beschreibungslevel durch Rückschlüsse auf ein anderes zu bestärken. Weiterhin könnte eine Theorie, die Zell- und Verhaltensebene verknüpft, zur Entwicklung von experimentellen Vorhersagen von einer Ebene auf eine andere führen und die Fähigkeit, Verhaltenssymptome ihrer zugrundeliegenden Symptomatik zuzuordnen, verbessern. Was das Entwickeln solch eines Verstehens der Erinnerung angeht, stellt die räumliche Erinnerung einen guten Startpunkt bereit, da man ähnliche Paradigmen bei Menschen und anderen Tieren benutzen kann.



**Abbildung 6:** Felice hat sich die Position der Gummibärchen (a) eingeprägt und begibt sich zum Zimmerausgang.

Das Vision and Memory Szenario besteht aus der Aufgabe ein Objekt anzuschauen, sich dessen Position im Raum zu merken, den Raum dann zu verlassen und schließlich den Raum erneut zu betreten und den Weg zum erinnerten Objekt wiederzufinden. Der Ablauf wurde zweigeteilt und die beiden Teile können auch unabhängig voneinander durchgeführt werden. Im ersten Teil wird Felice vor einen Tisch gestellt und blickt geradeaus an die Wand dahinter. Vor ihr auf dem Tisch befinden sich einige rote und gelbe Gummibärchen. Diese werden angeschaut. Dann wird der Blick des Agenten wieder in die Ausgangsposition versetzt und ihm wird der Befehl gegeben, durch das Zimmer in Richtung Tür zu laufen (Abb. 6). Ziel ist eine Position, die kurz vor der Tür liegt. Dabei steht der Tisch in der Mitte des Raumes zwischen Felice und dem Ziel, so dass sie einen Bogen laufen muss. Der zweite Teil startet vor der Tür mit Blickrichtung ins Zimmer. Hier soll sich nun in Zukunft ein Gehirnmodell an die Position der Gummibärchen erinnern bzw. diese suchen. In der Test-Implementierung wird angenommen, dass die Position bekannt ist und Felice läuft dorthin. Am Ende des Szenarios steht sie nun wieder vor den Gummibärchen und schaut sie erneut an.

## 5 Fazit und Ausblick

Das Ziel dieser Arbeit wurde erreicht: Es liegt nun eine stabil laufende, virtuelle Versuchsumgebung vor, die vier verschiedene Szenarien bereitstellt um Objekterkennung, Raumorientierung und visuelle Aufmerksamkeit auf einem oder mehreren Objekte untersuchen zu können. Die Implementierungen bieten die Möglichkeit, kleinere Änderungen ohne großen Aufwand vorzunehmen, den Ablauf anzupassen und die Szenarien auf andere Problemstellungen im visuellen Bereich zu erweitern. Die Farbe, Position und Anzahl der Objekte kann jederzeit verändert werden und auch das Nutzen anderer 3D-Modelle um die Form zu variieren stellt kein Problem dar. Natürlich gibt es noch viele Möglichkeiten, die virtuelle Umgebung in Zukunft je nach Bedarf auszubauen, wofür die momentanen Implementierungen eine hervorragende Grundlage bieten. Dem Agenten könnten noch mehr Interaktionsmöglichkeiten mit seiner Umgebung (und die zugehörigen Animationen) gegeben werden. Beispiele dafür sind unter anderem ein Seitwärtsschritt für das Body Displacement Szenario, eine Animation um gegriffene Objekte an einer bestimmten Position abzulegen und die Möglichkeit Türen zu öffnen und damit tatsächlich das Zimmer zu verlassen. Auch in grafischer Hinsicht ist noch eine Menge möglich (Raytracing, Motion Blur, bessere Materialien etc.). Dabei muss man allerdings abwägen, ob man eine maximal realistische Szenerie benötigt, oder ob es in Hinsicht auf die experimentellen Tests nicht sogar von Vorteil ist, die Darstellung möglichst einfach zu halten. Ebenfalls denkbar ist eine zukünftige Umsetzung in der Robotik, die sich an der VR orientiert.

## Literatur

[ann] <http://annarchy.readthedocs.org/en/latest/> 14.08.2015.

- [Ant14] M. et al. Antonelli. A Hierarchical System for a Distributed Representation of the Personal Space of a Humanoid Robot. *IEEE Transactions On Autonomous Mental Development*, 6(4):259–273, 2014.
- [BBB07] P. Byrne, S. Becker und N. Burgess. Remembering the Past and Imagining the Future: A Neural Model of Spatial Memory and Imagery. *Psychological Review*, 114(2):340–375, 2007.
- [BH08] F. Beuth und F. H. Hamker. The guidance of vision while learning categories. *Poster on the Bernstein Conference on Computational Neuroscience (BCCN)*, 2008.
- [CA05] P. Cavanagh und G. A. Alvarez. Tracking multiple targets with multifocal attention. *Trends in Cognitive Sciences*, 9(7):349–354, 2005.
- [cor] <http://docs.unity3d.com/ScriptReference/MonoBehaviour.StartCoroutine.html> 15.10.2015.
- [Ham05] F. H. Hamker. The emergence of attention by population-based inference and its role in distributed processing and cognitive control of vision. *Computer Vision and Image Understanding*, 100:64–106, 2005.
- [pro] <https://developers.google.com/protocol-buffers/docs/overview> 20.09.2015.
- [Rol11] M. et al. Rolfs. Predictive remapping of attention across eye movements. *Nature Neuroscience Advance Online Publication*, 2011.
- [spa] <http://www.spacecog.eu/project/index.html> 14.08.2015.
- [VPM07] S. Van Pelt und W. P. Medendorp. Gaze-Centered Updating of Remembered Visual Space During Active Whole-Body Translations. *J Neurophysiol* 97, Seite 1209–1220, 2007.
- [VWVO88] S. M. C. I. Van Wetter und A. J. Van Opstal. Experimental test of visuomotor updating models that explain perisaccadic mislocalization. *Journal of Vision*, 8(14):1–22, 1988.