Published by :

MICHAEL JAMES HERON (https://orcid.org/0000-0003-3393-0733)

Dr. Michael James Heron is a Senior Lecturer in Interaction Design and Software Engineering, working at Chalmers University of Technology. His primary research interests are in accessibility, games, and accessibility in games. He is also the editor of Meeple Like Us, probably the most comprehensive data-set of board game accessibility available on the Internet.

INTERNATIONAL JOURNAL OF GAMES AND SOCIAL IMPACT, Vol. 1 Issue no. 2
pp. 98-118
DOI: 10.24140/ijgsi.v1.n2.05
ijgsi.ulusofona.pt

COMPUTER SUPPORTED ACCESSIBLE DEXTERITY-BASED BOARD GAMES

MICHAEL JAMES HERON
Chalmers University of Technology / University of Gothenburg (Sweden)

Abstract

Board game accessibility analyses conducted by the Meeple Centred Design project have identified a number of problem areas where there exist no accessible intersection of disability and game design paradigm. While there exist fun and innovative board games for most players, making use of most game mechanisms, there exists no identified dexterity game accessible to those with physical impairments. Indeed, it is hard to imagine what form a game of that nature might take. In this paper, we outline a technological solution making use of computer vision, digital representation, and accessible game design. In its proof of concept form, it serves as a way of making the game Crokinole playable by mixed-needs groups. Future work will generalise this solution to work for a wide array of flicking and pushing based dexterity games, along with investigating how existing digital accessibility support tools can be leveraged to expand the demographic that can benefit from this approach.

Keywords *board games, tabletop games, crokinole, accessibility, computer vision, meeple like us*

Acknowledgements

This work was funded by a grant from Stiftelsen Promobila, awarded to the paper author, for work specifically focused on exploring accessibility in physical dexterity games for young adults.

Introduction

For several years now, the author of this paper has been engaged in a systematic study regarding the accessibility of modern, hobbyist board games. This work has primarily been published through a public-facing research blog called Meeple Like Us (Heron et al., 2018a, 2018b). The results of this ongoing analysis represent the first serious exploration of the accessibility landscape of board games. Meeple Like Us is a map to previously unexplored territory but also notable for the extent to which its cartography is transparently annotated. It is made up of deep, detailed 'accessibility teardowns' on over two hundred and thirty games. This work has been widely referenced in the discourse surrounding board games, and has been highlighted in venues as diverse as the BBC; the Guardian; Forbes Magazine; New York Magazine; Monocle and many more places. It represents then an invaluable source of primary analysis of the accessibility of board games. It is also a body of work that represents considerable critical analysis of games. Each game that is evaluated comes with a comprehensive critique. Each game is given a rating, and that rating represents the site's view as to the quality of the game independent of its accessibility profile.

As part of this work, several exploratory tools were written to help users connect with accessible games, as well as make informed decisions on whether such games will be enjoyable for their playing context. The site provides a recommender that permits users to set an accessibility profile along with a minimum game rating. This permits visitord to choose from a curated selection of games that will likely meet their specific usability needs. Figure 1 shows an example, where someone with colour blindness and complex physical accessibility needs is looking for recommendations of games that the site regards, at minimum, to be 'good'.

If Meeple Like Us represents a map to hitherto unknown territory, it is important that it can be used to identify locations of rich, vibrant interest. There are good games available for many combinations of accessibility needs, and even more if one wishes to discount the subjective and evaluative opinions of the author. The full teardown for each game outlines in detail the reasoning behind their grading so as to allow readers to make an informed decision. The metaphorical map can point out areas of gaming that can be productively explored, but it is up to the individual to make the final call on applicability. These games cover a range of different styles, and incorporate a range of different mechanics. Someone who has needs for a cognitively accessible game will have a wide range of options. Someone who needs games that are maximally accessible with regards to memory may find joy in anything from Rhino Hero to Telestrations to Ice Cool to Funemployed. Some house rules and compensation strategies may be required, but the site highlights the availability of gameplay experiences that stresses physicality (Rhino Hero / Ice Cool); or ridiculous social improvisation (Funemployed); or comic drawing (Telestrations). There is a wealth of different experiences available a buffet table of fun from which to select mouth-watering morsels.

However, there exist combinations of gameplay experience and disability that are not well supported. Within the metaphor of our map, we can think of these as featureless wastelands. One of these problem areas concerns dexterity games

This is still a beta tool – please send any bugreports to me at [redacted]

Disability Criteria	Minimum Value
Colour Blindness	B (recommended)
Visual Accessibility	NA (not applicable)
Fluid Intelligence	NA (not applicable)
Memory	NA (not applicable)
Physical Accessibility	A (strongly recommended)
Emotional Accessibility	NA (not applicable)
Socioeconomic Accessibility	NA (not applicable)
Communication	NA (not applicable)
Minimum [redacted] Rating	3.5 (good)

Find Games Reset

Suggested Games For Specific Disability Profiles

These games may be suitable for your criteria. Links in the first column are affiliate links - we earn a small commision if you buy a game using one of those.

Name	Review	Teardown	BGG Rank
Arboretum	4	teardown	298
Blank	3.5	teardown	7968
Coup	3.5	teardown	575
Funemployed	4	teardown	2371
High Society	3.5	teardown	546
Imperial Settlers	4.5	teardown	274
Jaipur	4.5	teardown	159
No Thanks	4	teardown	474
Once Upon a Time	4.5	teardown	1924

Figure 1
The Meeple Like Us Board Game Recommender

for those with physical accessibility needs. Ice Cool and Rhino Hero, as discussed above, fall into this category. Ice Cool is a flicking game, stressing fine (Judge & Stirling, 2003; Perez-Marmol et al., 2016) and gross (Morris & Bartlett, 2004; Wang, 2004) motor control with regards to accuracy and positioning. Rhino Hero is a stacking game, and stresses the ability to place things on top of other things without upsetting a fragile, collaboratively built structure. Neither of these games are suitable for someone with a physical disability. That these specific games are unsuitable need not be considered a great problem in itself, unless there are no alternative options that fill that usability use-case. This specific combination of desired gameplay experience and disability profile currently has no game that can be recommended. Sustained analysis of the accessibility landscape suggests that such a game does not exist, and perhaps **cannot** exist.

This paper outlines a solution to this problem, by shifting the inaccessibility into an area where compensatory strategies are possible. It provides a concrete example of the way in which a completely inaccessible flicking game can be made accessible to those with physical disabilities. This is done through the use of computer vision algorithms, some simple 3D modelling, and a set of software tools implemented through Unity. It effectively converts the problematic requirements of precision, power and positioning into a single click of a button, a key, or an external switch device. In this paper, the game chosen for this work is called Crokinole, however the technique can be effectively ported into other game contexts.

The Physical Expectation of Dexterity Board Games

The approach of Meeple Like Us, and the wider Meeple Centred Design project in which it resides, is one that focuses on an analysis of what a game asks of its players. We make no comment on what impairments may inhibit players—that is not for us to decide. What we seek to do is provide an informed commentary on what skills, abilities, knowledges and competencies are expected from the design and implementation of the game under discussions. Conclusions as to whether a game is appropriate for any individual player are left up to the reader, who will be in a better position to assess the data within their own personal context. The map of Meeple Like Us can only ever say 'Here (May)Be Dragons'.

With this in mind, we can consider the set of dexterity games that the site has covered as a data-set that outlines in detail the physical expectations of play. The following games were assessed as part of this work–most have a public teardown on the site that can be examined to investigate the specific nuances of their accessibility profiles. The others have been privately analysed, with teardowns existing only in draft form. The public teardown, where appropriate, is provided for each game:

Cube Quest; Rhino Hero; Rhino Hero Super Battle; Terror in Meeple City; Junk Art; Meeple Circus; Jenga; Ice Cool; Crokinole; Ice Cool 2; Pitchcar; Shuffleboard; Sonara

These games largely make use two key mechanisms, which may be combined or experienced in isolation. There are stacking games (Rhino Hero; Rhino Hero Super Battle; Jenga; Junk Art; Meeple Circus) and flicking games (Cube Quest; Terror in Meeple City; Ice Cool; Ice Cool 2; Pitchcar; Crokinole; Shuffleboard; Sonara).

Stacking games

The main gameplay mechanism within a stacking game is centred around placing pieces–often awkwardly shaped and inconveniently weighted–on to other pieces. In some cases, this is to create a catastrophe point where all the effort that has gone into a collaborative construction collapses in a moment of joyful catharsis (Rhino Hero; Rhino Hero Super Battle; and Jenga being examples where the fun payload (funload) comes from when it all falls apart). These games we will term 'inevitable collapse' games. In other cases, a sense of accomplishment is based on scoring acquired through skilful manipulation of pieces. In this we might consider Junk Art and Meeple Circus as ideal examples, where players gain score based on how spectacularly they execute upon the arrangement of game components. In the scope of this paper, we will call this 'demonstrated skill' stacking.

Within games operating within an inevitable collapse paradigm, accessibility issues are lessened because it is the fall that is the thing. These games often make use of a shared game state where each player inevitably creates complications for the next. There is no succeeding in these games, merely delaying the catastrophe. It is in this that we find the fun–it is not to achieve, but to engineer the circumstances under which someone else will **fail** to achieve. Each pull of the Jenga tower, or each card added to the Rhino Hero edifice, involves us navigating increasingly difficult physical puzzles while adding in complexities of our own. When the tower falls, we are rewarded with the spectacle of the collapse, and the comedy that comes with it. However, these games all exhibit a 'minimum viable product' which they must reach for the catharsis to be experienced. It's not fun if the Jenga tower is knocked over in the first pull. It's not fun if the Rhino Hero building collapses on the second card. Each successful interaction with the game builds anticipation. If there's not enough

Figure 2
Rhino Hero Super Battle

anticipation built, the experience is a disappointment. Thus, even in games where catastrophe is the goal, accessibility is still a vital and problematic factor.

The skills stressed in these games are varied–visual acuity is important in being able to precisely gauge distance, especially in games where the goal is in staving off a collapse. Stacking games represent a problematic design for physical accessibility **and** visual accessibility. Inevitable collapse games tend to involve constructions getting bigger, sprawling more, and becoming ever more precarious. See Figure 2 for an example of how Rhino Hero Super Battle looks as the game goes on.

Early stages of the game involve primarily fine-grained motor control and the ability to judge and control for depth. They involve assessing structure solidity, and then manoeuvring pieces in and out of a dangerously unstable construction. Rhino Hero Super Battle adds to this by requiring players to move weighted pieces up and down the building, or hang monkeys onto particular floors, without upsetting the whole.

As the game goes on, gross motor control becomes more important. It becomes necessary to stand up and move around the shared structure. It becomes important to assess things from multiple angles, and even contort the arm and hand to reach around inconvenient obstacles. A single jerk of the limbs, or spasm, or even sneeze, can bring the entire thing down. As such, these games tend to progress in terms of their emphasis on skill sets–fine-grained and gross motor control–until the point that it all ends in laughter.

The 'demonstrated skill' category of games also exhibit some of this feature–it is enjoyable to watch someone else defeated by their own hubris, especially when–as in something like Meeple Circus–things are being done to a time limit or are being carried out under group scrutiny.

However for these kind of games the fun comes from accomplishing a difficult task that may involve balancing several competing requirements. In Meeple Circus, some pieces only score if they're under other pieces, or on top of others still. Some gain points based on being balanced in conjunction with perversely unbalancable partner pieces. The extent to which a player can master these often tricky combinations is where the satisfaction lies. These games tend to emphasize fine-grained control to the exclusion of gross motor control, as it is the accomplishing of intricate feats of arrangement in which the fun may be found.

Flicking and Pushing games

Flicking and Pushing games in many ways offer a far simpler expectation in terms of mechanisms. The expectation is that you make use of the appropriate force, applied in the appropriate direction, to move a piece in a way that accomplishes a wider game goal. Cue sports, such as Snooker, Pool and Billiards could be argued to fall into this general category–the primary distinction being the use of an intermediate tool rather than direct manipulation (Pan et al., 2021). Games may involve flicking pieces (Cube Quest; Ice Cool; Ice Cool 2; Crokinole; Pitchcar) or pushing pieces (Shuffleboard). What changes in all of these different incarnations of the game form is the specific accessibility profile required of players. A game which involves pushing makes more use of joint movement at the wrist with correspondingly less on micropositioning at the fingers (Longmuir et al., 2013). A flicking game may put restrictions on what counts as a flick–for example, using a thumb to build up pressure before releasing it is often prohibited. In general they require the ability to use one finger to strike a piece (often oddly weighted and proportioned) at a direction and force compatible with having it trace a predicted line, or arc, to a target or target zone. Traditionally this stresses the ability to assess and action a plan in the Z axis, rotated around some reliably accessible strike point. Skilled players may also take into account the X and Y axes–making use of angled shots; top and bottom spin or its equivalent; or perhaps even making use of the ability to have a piece leave the physical confines of the board. Trick shots in Ice Cool for example allow a player to leap over the walls of the board, or trace a curve through doors. Pitchcar by comparison explicitly punishes players that have a piece leave the track of the race. Most games acknowledge real world complexity by allowing repositioning of pieces when certain criteria are met.

Much as with the inevitable collapse style of dexterity game, there is also a considerable degree of pressure put on gross motor skills. The game board may be large (Crokinole or Shuffleboard), or it may sprawl in complex ways (Pitchcar) or have physical walls within the wider constraints (Ice Cool). Even in games where that isn't true (such as Cube Quest), it is rare that one can get a good angle on a piece from the same sitting position in which one might start play. One has only to view the video of Ronny Sulivan's first experience with playing Snooker in VR to see how important it is to freely engage physically with a game board. The extent of repositioning required is often underestimated when one considers the game mechanisms out of context.

A Solution for Crokinole

Where game design cannot suffice, technological solutions can offer a way to bridge need and feasibility. Previous work by Noble and Crabb (Noble & Crabb, 2016) has for example explored addressing issues of cognitive accessibility through projection of supporting information directly onto a board game layout, while Bhaduri et al. (Bhaduri et al., 2017) has explored 3D printing to explore new modalities of interaction. Taking a similar approach to technologically-mediated accessibility, a solution was implemented for Crokinole that focuses the complexity of the physical interactions with dexterity games into a digitized single-click paradigm. This is widely considered appropriate for even severe physical accessibility needs (Terrill et al., 2019; Wasterfors & Hansson, 2017). This is actualized through the following outline procedure:

1. A web-cam placed above the game board captures real time footage of a game in progress.
2. Computer vision algorithms identify game pieces against the board, define a circle around their physical location, and calculate an origin point for each.
3. Each of the pieces identified by the computer vision algorithm are sent via asocket to a server running inside an external piece of software. The X and Y position, along with identified player ownership, are transmitted individually for each piece.
4. This software constructs a representation of the game pieces within a digital game world.
5. The game software normalizes co-ordinates and sizes based on physical dimensions drawn in from a configuration file.
6. When it is a disabled player's turn to act, they click within the digital game world once to indicate direction, once again to indicate force, and then the physics engine of the game software creates a simulation of what happens.
7. The game software then halts until the game state captured by the overhead area matches that of the digital version, at which point play continues normally.

The software solution here is unidirectional–it allows for the state of a physical game to be captured in a digital form, but changes in the digital form must be replicated manually on the board. Possibilities for automating this exist, such as through magnets, robot arms, and even drones. These fell well outside the scope of this proof-of-concept but represent interesting options for future work.

Flicking games were picked as being within the achievable scope of a solution. The uniquely contextual aspects of stacking games, and the often intractable complexity of the physics, created a circumstance where any solution means basically building a bespoke digital implementation of one specific game. This issue is compounded by the fact that stacking games offer no clean opportunity for implementing 'replicability'–it is not possibly to alter the real game state to match the digital representation in the way that the approach outlined in this paper permits.

For this proof of concept, Crokinole was chosen as an ideal candidate for implementation (Clement, 2019). Crokinole is a game, likely of Canadian origin, in which discs are flicked in turn by players with the intention of landing them cleanly in the centre hole. When one or more opposing pieces are on the table, the active player must first strike one of these as part of their flick. Pieces that leave the board or fail to make legitimate contact with an opponent's piece are removed along with any of their own pieces that were struck in a turn. Each player has twelve discs that they will flick during the course of the game, and the end score is based on which scoring quadrant each disc remaining on the board landed within.

Crokinole was chosen for several reasons. The first is that the board is a clean circle with well differentiated regions. The pieces are discs, and thus easily processed by OpenCV's

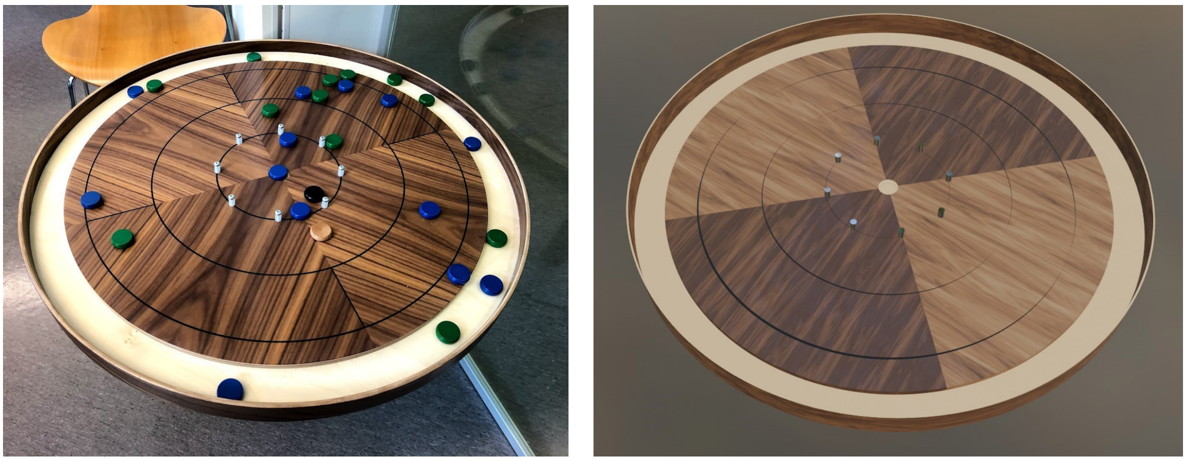

Figure 3
Real-life Crokinole Board (left) versus Blender representation (right)

built-in functions. There is no complex state that must be remembered between turns–the state of the board represents the state of the game. The model of flicking used within the game is pure, and the physics of play are handled by the board itself. Where a disc begins and ends is, in other words, a pure product of the natural physics of force applied to wood on polished wood. The natural colour of discs in Crokinole tends towards black and wood tones, so a replacement set was used to allow for more vibrant greens and blues.

OpenCV was chosen as the computer vision framework, offering as it does a set of industry standard software libraries that allow for rapid prototyping and developing of computer vision solutions. The board used in the Unity back-end was first done through creating simple cylinders and scaling them, and was later evolved to use a Crokinole board replica built in Blender, as shown in Figure 3.

As this represents the proof-of-concept work for a technical architecture, no user testing was conducted on this software prototype. The solution was evaluated only from a set of functional requirements, chief amongst these being that it should be possible for a player to play a game of Crokinole where one player was at the table, the other at a computer.

Implementation

OpenCV Implementation

The software architecture is broken up into two main applications. The first of these is a Python script that sits on the computer running the webcam. The other part is handled within Unity. The local image capture script is called Crokinole.py, and is responsible for reading video signals from the overhead webcam. The Unity system handles reconstructing a game engine version of the world as it is captured from video. In our study, Video was recorded from a Microsoft LifeCam connected up to a standard Windows 10 based desktop. Figure 4 shows the distribution of responsibilities between the two parts of the system. Test versions of the local image capture script were developed for an Android phone and an iPad, the utilisation of which would reduce the complexity of setup if expanded upon for home use.

The webcam was held in place with the use of a tablet arm, and positioned directly over the board. The central hole of a Crokinole board, being both circular and coloured neutrally, serves as a perfect anchor point for extracting a working area from the full video feed. The known size of the hole, and its centrality, is then usable for calculating scaling.

The size of a Crokinole board is in excess of that of many board games, and this necessitated the camera being held at some distance from the board. This in turn caused some issues with OpenCV accurately detecting each of the discs as a separate circle, especially in circumstances where discs were close in proximity to others.

The video image brought in from the webcam underwent several standard transforms to identify individual pieces. This was done in two passes–one for the green discs, and one for the blue. For each, a colour mask was set and applied to the raw frame to extract only those regions with the appropriate colour codes. This image was then converted into greyscale for ease of processing. Erosion and dilation (Singh, 2019) were applied to eliminate false positives from the image. Erosion, much like its name would suggest, erodes identified boundaries to more clearly differentiate white noise from the correctly identified shape, and helps differentiate circumstances where two or more discs are in very close proximity. Dilation does the opposite–erosion reduces the working area of identified shapes, and dilation can be used to expand a shape to something easier to work with. This allowed us to end up with a kind of cartoon exaggeration of our disc, but one that is more reliable as something properly identified as a game piece. Finally, a closing operation (Xie & Lu, 2013) was used to repair small holes in the identified shapes. A video of the process before the digital Crokinole board was implemented is available[16].

This process gives us a clean image upon which we can do more advanced shape identification. The first version of the code was based on Hough Circles (Leavers, 1992), the usual mechanism for identifying circles in an image. This resulted in a lot of false positives from a range of interference sources. Sometimes it was grain patterns in the wood of the board (which is decorative in the Crokinole board used as part of this work) and other times from external light sources (the board is very reflective). Early work with the algorithm focused on eliminating these false positives by filtering on size boundaries, but in the end the code became overly fit to one specific Crokinole board, in one specific lighting environment. Small circles below a certain radius were removed as a matter of the cleaning process using the ones() function of NumPy (Bisong, 2019), but artefacts remained. In addition, findCircles was more computationally expensive than a real-time interlink in a real-time game could sustain–there was obvious lag as the images were processed.

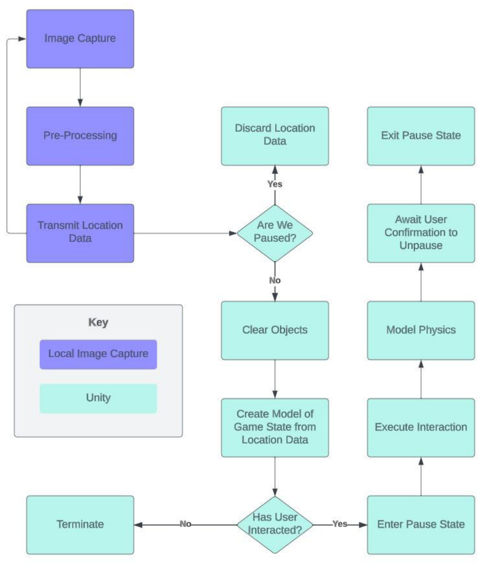

Figure 4
The Distribution of Responsibilities

Instead, the code was adapted to make use of topological structure analysis (Suzuki et al., 1985) as implemented through the standard OpenCV function findContours. Contours in OpenCV are detected curves that join a set of continuous points along a boundary. Points that share a colour or intensity are identified as being part of the same contour. These contours were then processed to find the minimum enclosing circle (Welzl, 1991) for each. The centre point of this provides an X and Y value as well as a calculated radius. It is this data that is then fed through the socket into the Unity backend. The socket sends data on a frame by frame basis, representing a continuous stream of locational data normalized around the identified centre hole of the board.

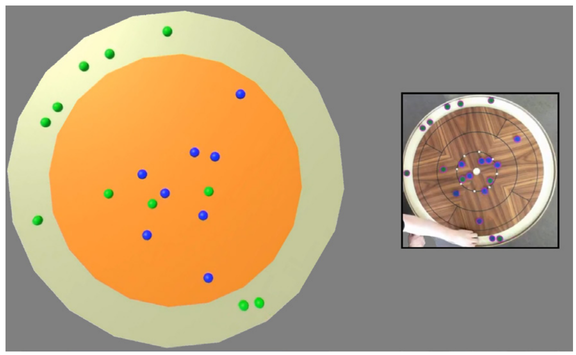

Figure 5
Early stage version of the process at work

The Unity Implementation

The dimensions of the Crokinole board used in this study were taken and used as the baseline for calculating the size of the pieces as well as the offset for the board from the camera. The model later constructed in Blender also conformed to these measured dimensions. Camera sway was calculated and compensated for through smoothing. This was done by building an average position of each part of the board and normalising this into a calculated canonical X,Y position. The OpenCV side of the code provided raw data. Unity normalized that into something that could be reliably used as the basis for constructing and destroying the game objects that represented each disc. This eliminated sway and the jitter of pieces on a frame by frame basis.

Critical to this approach is the recognition that there is no persistence of state between reads of the socket. The Unity part of the code will animate a disc being flicked across the board, but it does this only as a natural side-effect of how our eyes perceive animation. Unity destroys the whole game state and reconstructs it every time it receives a frame of information. No piece ever moves, or persists for longer than the length of a video frame. Correctly speaking, this Unity system does not actually identify any part of the game. Instead, it is completely stateless and simply creates, from scratch, the frame as it is described by the computer vision routines.

This approach though is effective, since it is only when a player at the computer chooses to interact with a piece that there is a need to persist anything. The only consequence is that there is a kind of 'mouse trails' effect that occurs as a piece moves, but this in itself is a useful accessibility feature (Heron et al., 2013) in that it gives a visual cue as to how a piece has changed position, in what direction, and at what speed.

When the player at the computer wishes to make a move, the usual update cycle is suspended. Object persistence is applied, and no further changes to the game state are permitted until the player has made a move. This persists until the real world game state from the webcam matches the state of the digital counterpart.

The other innovation in the Unity system is that pieces can be identified through clicks or cycling through them by pressing tab. When a piece is selected and the player wishes to 'flick', a laser beam appears in the centre of the disc, showing a broad beam of light

indicating the direction in which the piece should go. The player can press a key, click a button, or tap an external switch to set the direction of travel. The speed at which this light moves is completely configurable, and can be scaled to the comfort level of the player as can the accuracy of the beam and the 'margin of error' applied to the direction to simulate the messy unreliability of human dexterity.

When a direction has been chosen, the control switches to a power gauge which goes up and down. Again, a click allows the player to set a level, at which point a force is applied to the disc in the game. The physics of the Unity engine are then used to approximate what would happen if the player had made the same attempt at the Crokinole board. Within Crokinole, this force applicator works only in 2D, rotating around the Z axis of the piece. The applicator itself can work in all three dimensions though, allowing for force to be applied in any arbitrary direction.

Updates from the camera feed are suspended during the computer player's move until the external, captured board state is brought into alignment with what is shown on the screen. Prompts are provided for this, showing the players which piece needs to be moved and where. When these match, the usual cycle of creating and destructing game pieces is restarted and the player at the board is then in a position to take their turn.

Discussion

This paper does not make the argument that this is a direct equivalent to players enjoying Crokinole around a shared board. The Unity physics engine offers only an approximation, and the tactility that goes along with playing a physical dexterity game does not survive the translation to digitization. It feels satisfying to knock an opponent's piece off the board with a skilful flick, and that emotional aspect is lost in its major aspect when converted to a click. The loss of verisimilitude is considerable and impactful.

However, for the user-cases anticipated for the next phase of this work, it represents perhaps the only way that two players with differing physical accessibility needs might play a shared game of Crokinole without it being entirely digital, in a video game adaptation form. This hybrid approach allows for the physicality of play to be maintained as much as is feasible in a mixed-needs environment.

This paper does not include any user-testing, as it is the feasibility of the technical architecture of the concept that is most germane to the work. As such, no claim is made to the extent to which disabled gamers may find this a desirable option. We suggest only that this is a promising avenue that may be explored to bridge the gap between the tabletop and the desktop. Much accessibility work focuses on translation, or redundancy. Either providing input and output in multiple forms or allowing users to choose the specific form(s) in which input and output are provided. This project represents a literal translation of a tactile board-game into an accessible video game interface. It is early proof-of-concept work, outlined here in the hope that others are inspired to explore alternate implementations for other games and other scenarios.

One important aspect of this work too is the trajectory of future development. Original work around this idea focused on the use of everything from Bluetooth to laser distance finders to create a workable digital representation of game state. Everything from robot arms to drones were considered for capturing video footage and potentially reconstructing it based on the digital front-end. However, much tremendously innovative work in research suffers from a laboratory bias. They describe carefully managed experimental results obtained with algorithms diligently iterated upon under ideal circumstances. They rarely outline a feasible path by which these insights might be operationally replicated in the home environment.

This proof-of-concept on the other hand needs nothing more sophisticated than a webcam, some way to affix it above the board, and a home computer or laptop. The original version of this framework had the webcam attached via Sellotape to a ceiling light before the tablet-arm approach was put in place–it is, in other words, a technological solution which can be easily adapted outside of a research context. The largest barrier in anyone trying this out for real in a home environment is the cost of a Crokinole board itself.

We argue then in this paper that this is a promising direction for implementing board game accessibility in realms where there are perhaps insurmountable obstacles to actioning it within the design and production of individual titles. We do not believe, based on our current analysis, that there is promising territory to be explored in the intersection between dexterity board games and physical inaccessibility. We believe an interface between the two is required, and this work represents an early step towards architecting this in a way that can be directly employed in the messy context of real-life play.

Given the specific nature of the contribution outlined in this paper (whether there exists a bridge between physical games and physically impaired players), the proof-of-concept is not aimed at user groups beyond its immediate use-case. However, the technical feasibility of this tool also argues for potential wider usage for those with different accessibility needs. One might consider for example the incorporation of optical character recognition to permit for cards in play to be converted into text, and then their contents to be parsed via text-to-speech through the Unity front-end. Cognitive support tools such as outlined in Noble and Crabb (2016) could be incorporated. However these remain as untested possibilities.

What is clear though is that a mechanism that translates the tangible properties of physical games into a digital realm opens up additional possibilities for support, annotation and interaction that serves as a promising base for future inquiry. The specific interrelationship discussed in this paper (physical impairment and dexterity games) was selected because it currently represents a chasm of provision–there exist, as far as the author can tell–no physically accessible dexterity games. More importantly, there are no obvious solutions or guidelines that can be implemented to create games that would qualify as being accessible for this demographic. The priority has been to see whether there exists a feasible path to a solution, thus the tight scope of this project. Intriguing avenues have opened up as a consequence, but these remain within the domain of future work.

Importantly though to this work has been a tight focus on issues of disability justice. The difficulties in creating physically accessible dexterity games are real–there are few other design niches in gaming that represent quite such infertile soil for innovation to take hold. It is the view of the author that it is vital that those with disabilities have options if they wish to participate in a particular form of entertainment. Popular culture is a massive generator of social capital, and being excluded from participation is to be excluded from its construction and its interpretation. As argued in Heron (2016):

> This is disempowering. It is alienating. It is exclusionary. And it doesn't soften any of this much to know that it is often accidental and unintentional.

It must be acknowledged that it is inappropriate to put forward this technical proofof-concept as if it were a fully featured solution. As discussed earlier, it represents a translation rather than a direct parallel. The experience is not identical, and much of value is lost in the process of converting tactile sense pleasure into digital accessibility. While this work does highlight a solution, it shouldn't be interpreted as an argument for why we should abandon the search for genuinely accessible physical games. There is an ethical burden on researchers in this area to seek out genuine parity, even if the quest for such a thing seems unlikely to yield rewards.

Future Work and Limitations

There are numerous limitations associated with this prototype. Some of these are a matter of developing Quality of Life (QoL) systems to lower the setup barriers. Some of these are fundamental to the goal of designing research software with the home user in mind. Some are specific to the game chosen.

This software is tightly bound to the context of Crokinole, and one might argue a specific Crokinole board. There was a considerable amount of trial error involved in finding the correct hue and intensity values for the original black and woodgrain pieces. The correct ranges that could be used for colour masking turned out to be very narrow, and subject to lighting conditions–a cloud passing over the sun could result in some pieces no longer being identified. The move to primary coloured pieces alleviated that issue considerably, but it is still the case that there is a calibration cost that goes with setting up the algorithm.

Important to ensuring clear identification of pieces is that there is a meaningful contrast between the colour used for each player, but also between the board and the player pieces. However, there is an additional complicating factor in that pieces should also be meaningfully distinct in terms of their colour from the surrounding context–the OpenCV mechanisms used for identifying circles do not limit themselves to what is on the board. They span the full range of the video frame. Circular patterns in the floor upon which the board is set can complicate detection. To resolve this, the preprocessing routines in Crokinole.py discards identified circles that are outside the radius of the board. This is simple for Crokinole–the board is circular and calculations are straightforward. It would be more complicated for games with asymmetrical layouts or with board configurations that sprawl during play. It is likely for other game contexts that it would be necessary to deeply bespoke mechanisms for identifying the bounds of the game environment.

It is possible to simplify the process of creating digital representations of games through the employment of restrictions of context. These include playing on plain, non-reflective floors, in curtained rooms, with configurable lighting rigs. Creating a uniformity to the hues in the video footage would alleviate many complexities. Within a lab setting, this may be appropriate. Were this system to be extended for home use, it would represent a likely excessive barrier to play. In the end, per-game configuration seems the only feasible option, and the work done within this prototype shows that it is a approach that can function adequately. Variations of position, camera distance, lighting contexts and more were investigated during the process of building the tool. Arriving at reliable results required a loose approach to both the process of identifying configuration values, but also in terms of what is acceptably robust for play. The smoothing algorithms that were used to fix discs in a reliable position serve an additional role in that they limit the impact of visual distortions in the video feed.

Within this proof of concept tool, some convenience tools were built into the software to allow the user to fiddle with hue values until they found the magic combinations that would permit for acceptably reliable detection of colour ranges. This remains suboptimal. Future work with the algorithm should work on creating simplifications to this. Long term, this is perhaps something that would work well in a machine learning context. Short term, the number of discs present in a game of Crokinole is already known in advance and automatically iterating through hues until two sets of twelve discs are identified is straightforward to implement and likely sufficient in the majority of cases.

Then there is the issue generalising this software to other games. Work is currently being undertaken to expand the codebase so it can also be used for Ice Cool. The identification of pieces is straightforward there. At the distance the webcam is set, the pieces in Ice Cool–despite being shaped like penguins–are picked up cleanly by the contour and circle identification regimes outlined above. However, Ice Cool has a third dimension to its tactility–flicking pieces upward is a thing that skilled players may often choose to do. Ice Cool also has a more complex board, which requires in turn its own modelling pass. This does not represent a fundamental obstacle to development, just one that requires time and gradual generalisation of the outlined algorithm.

The process of applying this approach to other games deserves some discussion, as it represents a technical and sociological challenge. Specifically, there are several processes that need to be undertaken:

1. The board of a game needs to be represented as a Unity object, either as a 3Dasset (as in Figure 3) or as some configuration of simple generated shapes (See for example 5).
2. The colour regions for the pieces of each player need to be identified and tweaked until they are resistant to fluctuations in lighting.
3. Pieces that do not present themselves as circles when viewed from above will need specialist application of computer vision techniques to capture.
4. Models for individual player pieces need to be made available–again, either as3D assets or as rough generated approximations.
5. The physical properties of pieces need to be represented in terms of their mass, size, and frictive traits.

When this had been accomplished, it would then be a straightforward process to convert the mechanisms for Crokinole over to using the new game context. The Unity algorithm does not implement game rules–it only permits for the simulation of physical engagement through digital means. However, the question arises as to who should be responsible for doing such work. As this is a proof-of-concept tool it does not have an installed user-base, and lacking such a thing it is hard to see how one might encourage publishers to produce such assets and configuration details. The technical barriers are not excessive, but the marketplace realities mean that initial work on building up a supporting library of games would need to be borne by researchers or a nascent community of contributors.

The system as it currently stands only handles translating the physical state to a digital state. We can conceive of no reasonable solution for replicating the digital to the physical. At least, none that would survive the change of context from the university lab to the living room table. The need for a player to physically recreate a digital state on a physical board may be a fundamental limitation. The need to mount a Crokinole board on a sturdy table limits the feasibility of under-the-board magnets, and attaching magnets to the discs would have significant impact on the physics of play. This is the issue with any approach we can envisage for automatic reconfiguration of physical board state.

However, we are always confronted here with the realities of physical adornment of real world artefacts. Even leaving aside the 'balance' issues of adjusting the way physics works, there are simple practical difficulties. Magnets would need to be powerful enough to work through a thick wooden board. That makes them inappropriate for Crokinole, but perhaps suitable for other games that employ cardboard. However, there is also a need to synchronize movement of magnets so that they do not fall out of alignment. That can be impossible to replicate in a physics based game unless it were to be simulated at a fraction of real-time speeds.

The simple geometry of a Crokinole disc makes it an almost ideal candidate for manipulation by robot arms, but the proportions of a board make it impossible with every-day robot rigs of the kind that you could feasibly imagine in a home environment. Using multiple arms set at various positions around the board could offer coverage, but at a co-ordination cost that would require specialist software to be configured for every physical arrangement of gameboard and game piece. Drones fitted with grippers solve a lot of these problems in theory, but they introduce an element of terror into play that falls out with the design of a game. Devices such as the CrazyFlie–fast, small and adaptable–are still extremely startling when they shoot at speed towards someone. A robot arm affixed to a remote control vehicle introduces complexities around velocity and inertia, and at a certain point in conceptualising these increasingly esoteric approaches it becomes obvious we're really just trying to justify buying new toys for the office. The more moving parts brought into the equation, the more difficult it is to imagine these solutions surviving transition into the home environment.

There are several games that could be converted over to this system that are likely to require more complex identification than contours and enclosing circles can provide. However, the symbology associated with board games contains within itself a happy solution to the problem–image recognition. Iconography attached to pieces (such as in Cube Quest) and similar attached to a game board can provide disambiguation opportunities that can be systemized via config files for each supported game. Where these are not present as part of the game, attaching reference images of our own to pieces can offer a convenient, low impact solution. This kind of after-market adaptation is an approach used by several games already, offering annotation via transparent stickers attached to pieces. Village requires each of the many dozens of meeples in a game to be paired to a sticker. Blood Bowl provides decals and decorations in the same way. Terror in Meeple City came with stickers representing clothes that could be added to each different class of piece. General purpose annotation stickers, paired with symbol dictionaries, could offer a way of creating a consistent way to deal with games with more complex pieces, or pieces that have the same form but differing meaning.

In essence, what is needed as part of the future work is for this proof-of-concept software to be built up into something that works as a user-facing tool. It needs more comprehensive configurability; a lower barrier to use; a greater range of supporting systems for individual context and playing environment. Exploring the opportunities, and complexities, of these is part of the next phase of work associated with the technical architecture of the tool.

Conclusion

The work outlined in this paper represents a promising proof-of-concept for bridging what may be an otherwise insurmountable gap in board gaming. There are currently no identified physical dexterity games that can be easily played by people with severe physical accessibility needs. Almost every combination of impairment and game design has several good candidates that can be assessed for playability. This though is one of the intersections that is- on the basis of the Meeple Centred Design map of the accessibility landscape (Heron et al., 2018a)–completely unoccupied. It is not clear what a physically accessible dexterity game might look like, or how it might play. The difficulty associated with envisioning its manifestation hints strongly at its unattainability.

With that in mind, we are faced with two possibilities. One is to accept players with severe physical accessibility needs simply won't be able to enjoy games in this mould. The other is to explore external solutions that offer alternate forms of interaction.

The approach in this paper has been to look at translating the physical activity of flicking a piece in a game. Instead of the fine and gross motor control expectations that we traditionally associate with Crokinole, a player can simply click a control to set direction, click again to set power, and then see what impact it would have on the game. While there are limitations to this approach, it demonstrates success in effectively addressing what seems like an otherwise intractable problem.

Endnotes

1 https://meeplelikeus.co.uk/board-game-recommender-for-people-with-disabilities-beta/
2 https://meeplelikeus.co.uk/rhino-hero-2011-accessibility-teardown/
3 https://meeplelikeus.co.uk/telestrations-2009-accessibility-teardown/
4 https://meeplelikeus.co.uk/ice-cool-2016-accessibility-teardown/
5 https://meeplelikeus.co.uk/funemployed-2014-accessibility-teardown-nsfw/
6 https://meeplelikeus.co.uk/ice-cool-2016/
7 https://meeplelikeus.co.uk/stay-in-your-lane/
8 https://meeplelikeus.co.uk/cube-quest-2013-accessibility-teardown
9 https://meeplelikeus.co.uk/rhino-hero-2011-accessibility-teardown/
10 https://meeplelikeus.co.uk/rhino-hero-super-battle-2017-accessibility-teardown/
11 https://meeplelikeus.co.uk/terror-meeple-city-2013-accessibility-teardown/
12 https://meeplelikeus.co.uk/junk-art-2016-accessibility-teardown/
13 https://meeplelikeus.co.uk/meeple-circus-2017-accessibility-teardown/
14 https://meeplelikeus.co.uk/ice-cool-2016-accessibility-teardown/
15 https://www.youtube.com/watch?v=OMceVbo3Tm4
16 https://meeplelikeus.co.uk/village-2011/
17 https://meeplelikeus.co.uk/blood-bowl-2016/
18 https://meeplelikeus.co.uk/terror-meeple-city-2013/

References

Bhaduri, S., Ortiz Tovar, J. G., & Kane, S. K. (2017). Fabrication games: Using 3d printers to explore new interactions for tabletop games. *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition*, 51–62. https://doi.org/10.1145/3059454.3059463

Bisong, E. (2019). Numpy. In *Building machine learning and deep learning models on google cloud platform* (pp. 91–113). Springer. https://doi.org/10.1007/978-1-4842-4470-8_10

Clement, R. (2019). Playing Canadian: A brief history of tabletop games in Canada. *The Spaces and Places of Canadian Popular Culture*, 227.

Heron, M., Hanson, V. L., & Ricketts, I. W. (2013). Accessibility support for older adults with the access framework. *International Journal of Human-Computer Interaction*, *29*(11), 702–716. https://doi.org/10.1080/10447318.2013.768139

Heron, M. J. (2016). Cultural integration and the accessibility of gaming. *The Computer Games Journal*, *5*(3-4), 91–94. https://doi.org/10.1007/s40869-016-0028-x

Heron, M. J., Belford, P. H., Reid, H., & Crabb, M. (2018a). Eighteen months of meeple like us: An exploration into the state of board game accessibility. *The Computer Games Journal*, *7*(2), 75–95. https://doi.org/10.1007/s40869-018-0056-9

Heron, M. J., Belford, P. H., Reid, H., & Crabb, M. (2018b). Meeple centred design: A heuristic toolkit for evaluating the accessibility of tabletop games. *The Computer Games Journal*, *7*(2), 97–114. https://doi.org/10.1007/s40869-018-0057-8

Judge, J., & Stirling, J. (2003). Fine motor skill performance in left-and right-handers: Evidence of an advantage for left-handers. *Laterality: Asymmetries of Body, Brain and Cognition*, *8*(4), 297–306. https://doi.org/10.1080/13576500342000022a

Leavers, V. F. (1992). The dynamic generalized Hough transform: Its relationship to the probabilistic Hough transforms and an application to the concurrent detection of circles and ellipses. *CVGIP: Image understanding*, *56*(3), 381–398. https://doi.org/10.1016/1049-9660(92)90049-9

Longmuir, P. E., Tyrrell, P. N., Corey, M., Faulkner, G., Russell, J. L., & McCrindle, B. W. (2013). Home-based rehabilitation enhances daily physical activity and motor skill in children who have undergone the Fontan procedure. *Pediatric cardiology*, *34*(5), 1130–1151. https://doi.org/10.1007/s00246-012-0618-8

Morris, C., & Bartlett, D. (2004). Gross motor function classification system: Impact and utility. *Developmental medicine and child neurology*, *46*(1), 60–65. https://doi.org/10.1017/S0012162204000118

Noble, K., & Crabb, M. (2016). Projection mapping as a method to improve board game accessibility. *ACM SIGACCESS Accessibility and Computing*, (116), 3–9. https://doi.org/10.1145/3023851.3023852

Pan, J. W., Komar, J., & Kong, P. W. (2021). Development of new 9-ball test protocols for assessing expertise in cue sports. *BMC Sports Science, Medicine and Rehabilitation*, *13*(1), 1–12. https://doi.org/10.1186/s13102-021-00237-9

Perez-Marmol, J. M., Ortega-Valdivieso, M. A., Cano-Deltell, E. E., Peralta-Ramırez, M. I., Garcıa-Rıos, M. C., & Aguilar-Ferrandiz, M. E. (2016). Influence of upper limb disability, manual dexterity and fine motor skill on general self-efficacy in institutionalized elderly with osteoarthritis. *Journal of Hand Therapy*, *29*(1), 58–65. https://doi.org/10.1016/j.jht.2015.12.001

Singh, H. (2019). Advanced image processing using OpenCV. In *Practical machine learning and image processing* (pp. 63–88). Springer. https://doi.org/10.1007/978-1-4842-4149-3_4

Suzuki, S., et al. (1985). Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, *30*(1), 32– 46. https://doi.org/10.1016/0734-189X(85)90016-7

Terrill, A. L., MacKenzie, J. J., Reblin, M., Einerson, J., Ferraro, J., & Altizer, R. (2019). A collaboration between game developers and rehabilitation researchers to develop a web-based app for persons with physical disabilities: Case study. *JMIR Rehabilitation and Assistive Technologies*, *6*(2), e13511. https://doi.org/10.2196/13511

Wang, J. H.-T. (2004). A study on gross motor skills of preschool children. *Journal of research in childhood education*, *19*(1), 32–43. https://doi.org/10.1080/02568540409595052

Wasterfors, D., & Hansson, K. (2017). Taking ownership of gaming and disability. *Journal of Youth Studies*, *20*(9), 1143–1160. https://doi.org/10.1080/13676261.2017.1313969

Welzl, E. (1991). Smallest enclosing disks (balls and ellipsoids). In *New results and new trends in computer science* (pp. 359–370). Springer. https://doi.org/10.1007/BFb0038202

Xie, G., & Lu, W. (2013). Image edge detection based on OpenCV. *International Journal of Electronics and Electrical Engineering*, *1*(2), 104–106. https://doi.org/10.12720/ijeee.1.2.104-106