

自由部門：20003

すなケツキ！

「すなケツキ！」は、砂場にカラフルなお絵描きができるシステムです。

夢のようなわくわくする空間を作りだしましょう！

はじめに

みなさん、幼い頃に夢中で砂場で遊んだ思い出はありませんか？
山や川を作ったりして 立体的に楽しめるのは砂場遊びのいいところです。
また、砂場遊びは遊ぶ人たちの創造性の育成や体のバランス感覚を養います。
その砂場に色が付けられたら、もっと楽しくなると思いませんか？
砂場にした絵が動き出したりしたら？



砂場

+



お絵かき

=



すなケッチ!



私たちは そんな わくわくする砂場あそびを
実現するシステムを作ります！

対象者

子供から大人まで・砂場あそびに興味のある人

「すなケッチ！」の概要

砂場で遊ぶと・・・

塗りたい場所に
好きな色を塗る
ことができる！



描いた絵は、写真や
パタパタアニメとして
保存できます



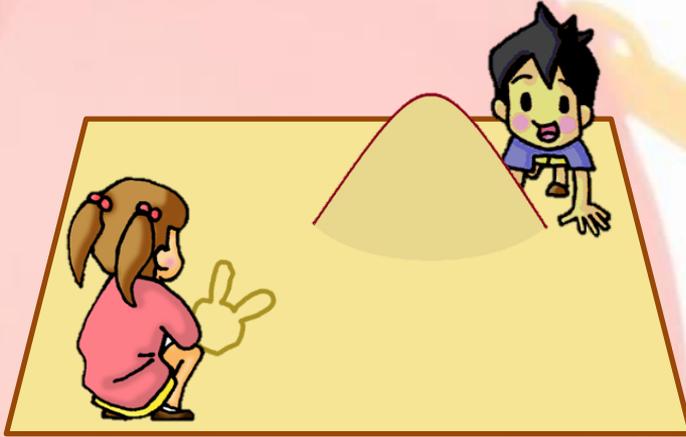
うちわを砂場に
かざす事でシステ
ムの操作ができる

砂場の凹凸に
合わせて色が
塗れる！

複雑な操作はなく簡単に砂場に色を塗れます！

基本的な遊び方

①



自由に砂場に絵を描いて遊びます。

②



絵に色を塗ることができます。

③



うちわを利用して配色方法の変更や
写真を撮ることができます。

④



たくさん書いて完成！
パタパタアニメもつくれます！

絵を描いていく

紙に絵を描くように指で線を引き
線の最初と最後を結んで
形を砂場の上で作る、絵を描きます。



砂場で遊ぶように山や谷などを
作って、立体的な形を作ります。

選択した色の塗り方で
色を塗ることができます。



また、高低差のあるところには、
色の濃淡のグラデーションが付きま



うちわによる操作

うちわに描いてあるマーカーで簡単に操作を切り替えられるので、手の汚れを気にせずに遊べます。

色の塗り方の切り替え

- ・ランダムに色を塗る
- ・声で塗る
- ・置いて塗る

写真の撮影

うちわをかざしてから3秒後に砂場全体の写真が撮れます。絵を少しずつ変えながら撮影することでパタパタアニメが作成できます。

アニメ再生

作成したパタパタアニメが砂場の上で再生されます。

色の塗り方

ランダムに色を塗る

砂場の低くなっているところ、高くなっているところを取得し、自動で色が塗られます。



**手間なしで簡単に
アート作品ができる！**



声で塗る

描いた絵に塗りたい色を声に出します。塗りたい色を言った後に、描いた絵を触ると塗りたい色に塗られます。



**直感で色を塗りやすく
絵が描きやすい！**



置いて塗る

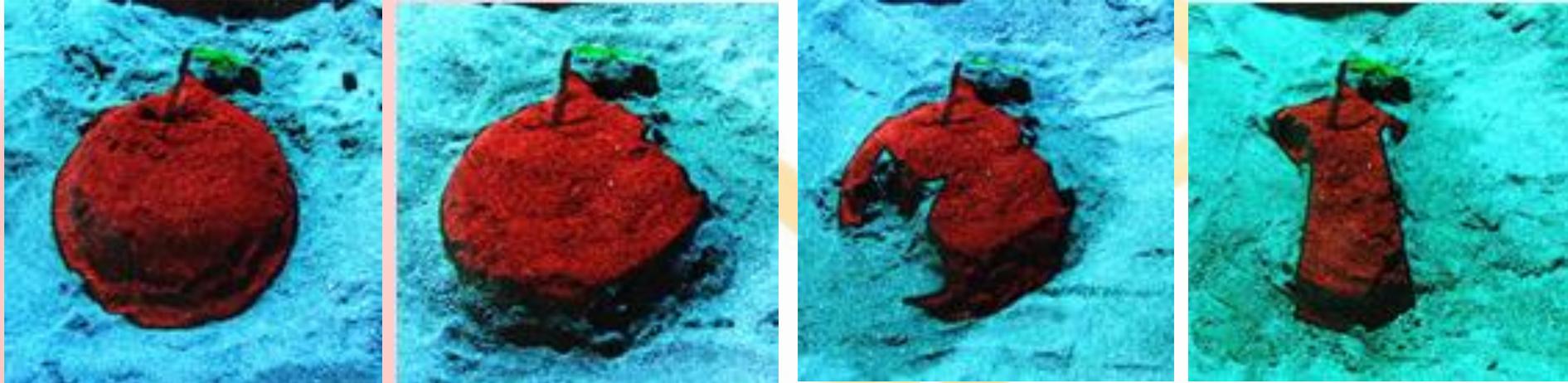
塗りたい色と同じ色をしたモノを砂場の塗りたいところに置くと、置いたモノの色を読み取って、それと同じ色に塗ることができます。



**細かい配色が可能なので、
砂場に本格的な絵が描ける！**



パタパタアニメ



砂場で描いた絵を少しずつ描き変えていき
絵を変えていくそのたびに写真を撮ります。
撮った写真を組み合わせることでパタパタアニメができます。
できたパタパタアニメは砂場で再生することができます。

**自分で描いた絵を砂場で動かすことができるので楽しく
作品の幅も広がります！**

システムの構成



プロジェクター



kinect

入力

出力

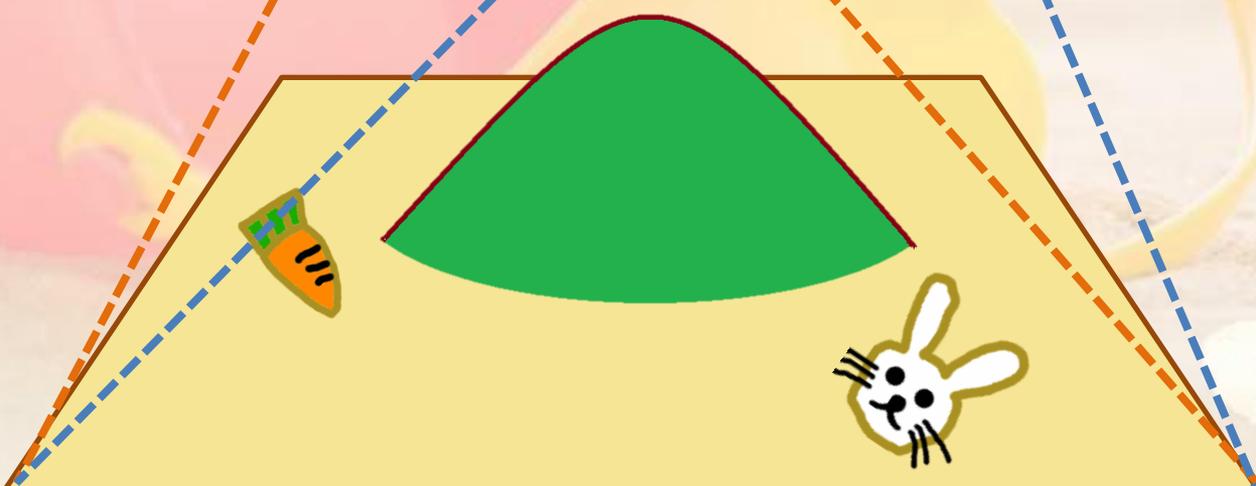
④ プロジェクターで
砂場に色を付ける

① Kinectから砂場までの
距離情報を取得

処理用PC

② 距離画像の作成

③ 配色の決定

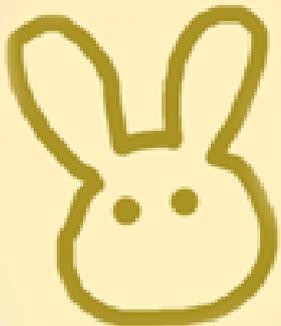


なぜ色が塗れるのか



Kinectで距離を取得し
左の図のような距離画像を作成します。

①絵を描く



②Kinectで取得した距離情報から距離画像を作成する



③配色方法の設定に合わせて色の指定をする

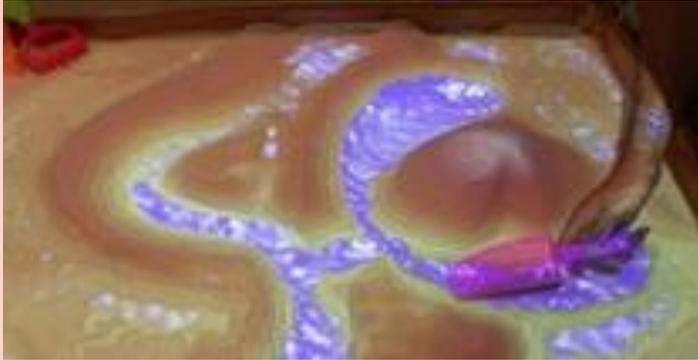
- ・ランダム
- ・音声認識する
- ・モノの色を読み取る

④色がつく

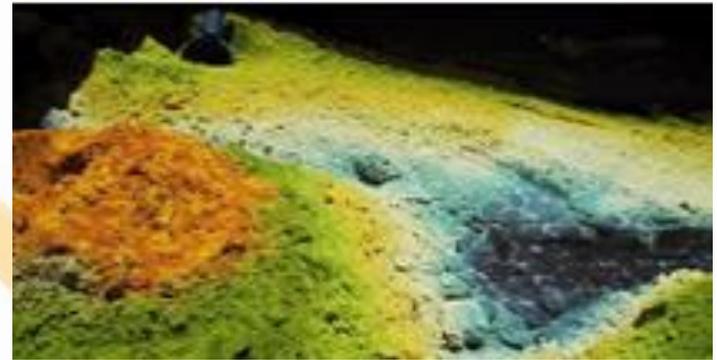


類似品との相違点・独創的な点

AR砂場



デジタル砂遊び



砂場の高低をkinectで認識して自動で色を付ける。

すると、山や谷があるようにプロジェクターで色が付けられる。
また、深いところには水も表示できる。

「すなケツキ！」は、砂場で描いた絵や作った山や谷に任意の色が自由に塗れる砂場遊びです。複数人で協力して砂場アートを作ることができます。砂場アートがパタパタアニメになり再生することもできます！

開発計画・開発環境

	4月	5月	6月	7月	8月	9月	10月
要求分析	→						
設計			→				
実装				→			
テスト						→	

開発環境 : Windows8 VisualStudio C#, Kinect
使用ライブラリ : Microsoft Kinect SDK, Speech Platform