

平成 30 年度 特別研究報告書

覚醒検知機能を持った
目覚まし時計の検討

龍谷大学 理工学部 情報メディア学科

T150487 小西 静夏

指導教員 三好 力 教授

内容梗概

近年、者の夜型が進行し、目覚まし時計を使用しているにもかかわらず、スヌーズ機能を止めてからもう一度寝てしまう人や、アラーム音が聞こえない人などにより二度寝をしてしまい朝が起きられないという問題がある。既に多種多様な目覚まし時計が開発されているが、二度寝を防止する機能に欠けている。本研究では、二度寝を防止するために、カメラを用いて人が起きているかを認識し、アラームを鳴らすシステムの開発の検証を行った。

目次

第1章	研究背景	1
第2章	既存技術	
2.1	逃げ回る目覚まし時計 NANDACLOCKY(ナンダクロッキー)	4
2.2	めざましカーテン mornin' plus	5
2.3	光の目覚まし時計 inti 4(インティフォー)	6
2.4	爆音目覚まし時計 スーパーライデン	7
第3章	提案手法	
3.1	アラームを止めてしまう問題点について	8
3.2	アラーム音が聞こえない問題点について	10
3.3	提案システム	12
第4章	実験内容	
4.1	実験目的	13
4.2	実験内容	13
4.2.1		13
4.2.1	寝室のドアを開ける実験内容	14
4.2.2	洗面台にある歯ブラシを取る実験内容	15
4.3	実験結果	16
4.3.1	寝室のドアを開ける実験結果	16
4.3.2	洗面台にある歯ブラシを取る実験結果	19
第5章	考察	21
第6章	まとめ	22
	謝辞	23
	参考文献	
	付録	

第一章 研究背景

近年、若者の夜型が進行し、朝が起きられない人が多いことが問題視されている。そのため、スマートフォンの目覚まし機能やカーテンが開く目覚まし時計、逃げ回る目覚まし時計など多種多様な目覚ましが開発され、朝を快適に迎えることができるように工夫されてきている。

図 1.1 より、目覚まし時計を使わずに朝起きる人の割合は 60 代以下では 10%未滿となっており、ほとんどの人がスマートフォンのアラーム機能や目覚まし時計を使用していることがわかる。スマートフォンが普及し、スマートフォンのアラーム機能を使用する人が増えてきているが、アナログ、デジタルの目覚まし時計の使用率は 74%とまだ高いことがわかる。また、スマートフォンのアラーム機能と目覚まし時計を併用している人もいる。これより、朝起きるのに目覚まし時計が必須であることがわかる。

しかし、目覚まし時計を使用しているにもかかわらず、寝坊することがあるかどうかのアンケートでは、図 1.2 から全体ではよくあるが 12.5%、たまにあるが 34.3%と寝坊してしまうことが多く、そのうちの約 7 割が二度寝による寝坊であると回答している。

図 1.3 の目覚めに関するアンケートでは、二度寝をしてしまう人の割合は、ほぼ毎日が 17.3%、週に 3~4 回が 29.4%と二度寝をしてしまう人の割合が高いことがわかる。目覚まし時計を使用しても二度寝をしてしまう理由としては、図 1.3 の目覚まし時計の音が聞こえない、記憶がないというアンケートではほぼ毎日が 5.1%、週に 3~4 回が 13.5%とアラーム音が聞こえず寝坊してしまっている。アラーム音が聞こえていてもスヌーズ機能を止めてからもう一度寝てしまう人やスヌーズを無意識に止めてしまいアラーム音に気付かなかったなどの理由も挙げられている。また、起きることができても眠い状態である人が多く、すっきり起きることができていないこともわかる。

よって、寝坊の原因で最も多いのは二度寝であり、寝坊するまでには至らなくとも、二度寝をしてしまい、寝坊の危険性がある人の割合が高いことがわかる。二度寝の理由として、起きていないにもかかわらずアラームを止めてしまう人が多いため、起きるまでなり続けることが重要であると考えられる。また、アラーム音が聞こえない人も多いため、音に対する対策も重要であると考えられる。

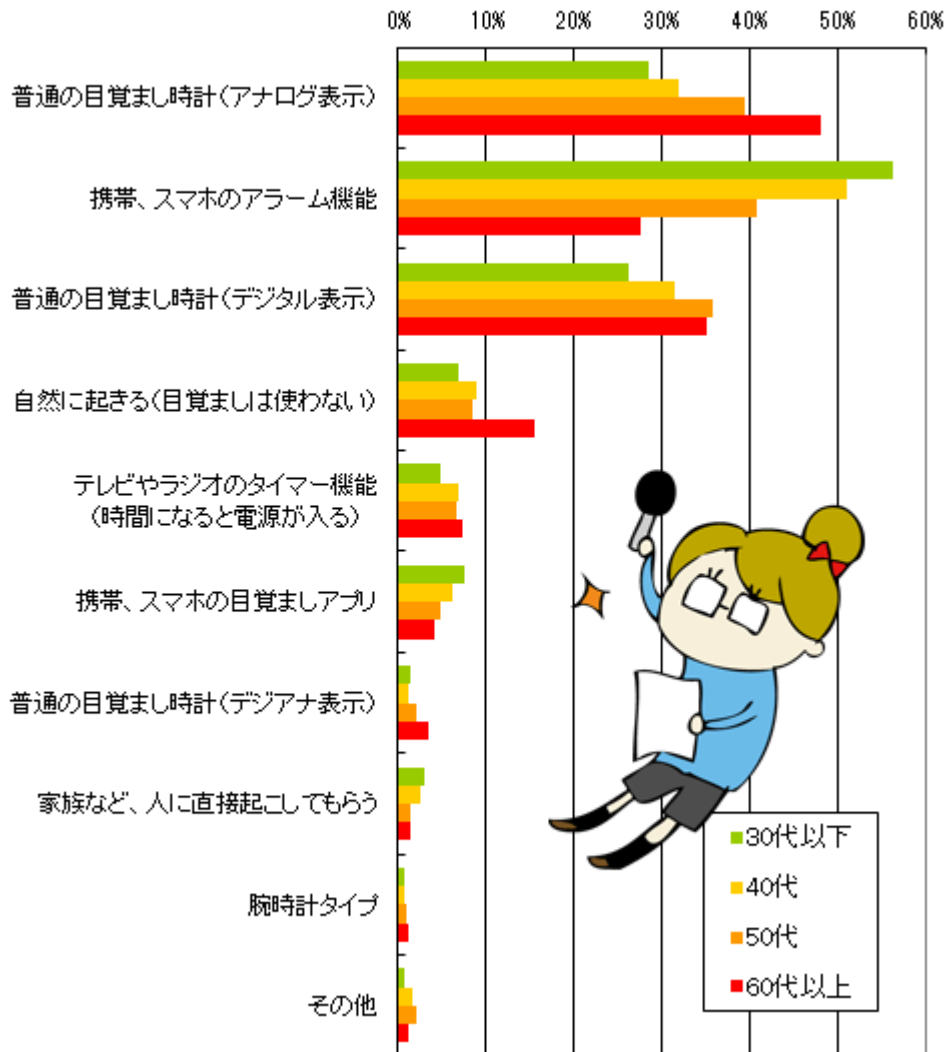


図 1.1 どのタイプの目覚まし時計を使用しているか

寝坊をすることはありますか？

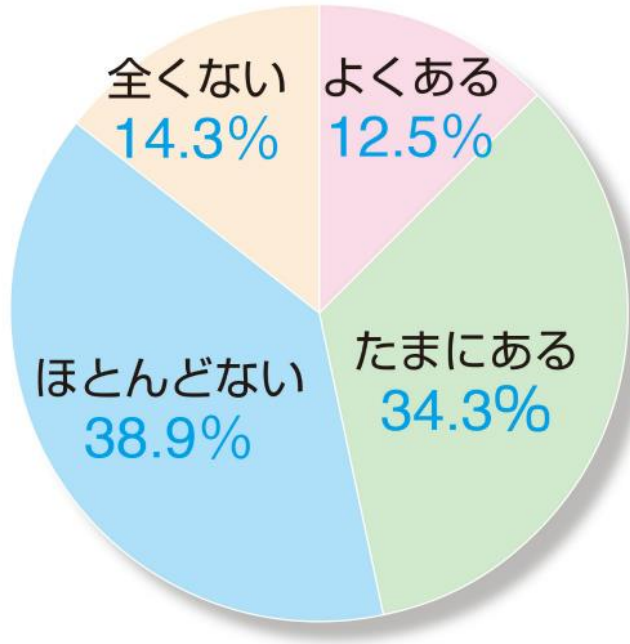


図 1.2 寝坊の割合

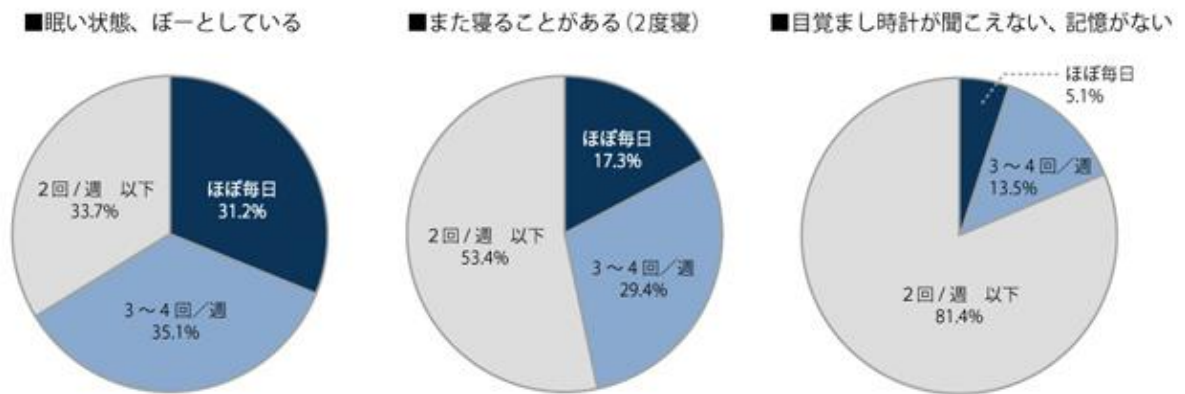


図 1.3 目覚めに関するアンケート

第二章 既存技術

2.1 逃げ回る目覚まし時計 NANDACLOCKY(ナンダクロッキー)

スヌーズ機能を設定してもスヌーズ機能を止めてしまい、スヌーズ機能の効果がない人におすすめの逃げ回る目覚まし時計である。

主な機能としては、60cm までの台から落ちることができ、ロボットのような言葉を発しながら床やカーペットの上を走り回る。走り回る時間は約 30 秒で、その後は動きをストップした状態でアラーム音が約 9 分間なり続けるため、ベッドの下などに入り込んだときはベッドを動かしてアラームを止めなければならない。スヌーズ機能もあり、起きるまで繰り返し逃げ回るといった仕組みである。



図 2.1 逃げ回る目覚まし時計 NANDACLOCKY(ナンダクロッキー)

問題点

部屋が広い場合は遠いところに行ってしまう、音が聞こえなくなってしまうことがある。また、走り回るときの騒音やアラーム音が大きすぎるため、集合住宅での使用は難しいという点がある。逃げた先が捕まえるのが困難な場所に入ってしまった場合、目覚まし時計を止めることに時間がかかってしまう。

2.2 めざましカーテン mornin' plus

アラームを設定した時間になると寝室のカーテンが自動で開き、太陽の光をめざましとし目は開いていても脳が起きていないだるさを解消して起こしてくれる目覚まし時計である。

取り付け方は、カーテンの端のフックを外し、mornin' plus を取り付け、チュートリアルに沿って mornin' plus とスマートフォンを接続するだけと簡単になっている。スマートフォンでアラームを設定し、設定した時間になると自動でカーテンを開閉する。曜日毎に設定できるため、毎回アラームを設定する手間を省うことができる。スマートフォンだけでなく手動で開閉することもできる。また、共有機能が搭載されており、家族全員で使用することができる機能や1台のスマートフォンから最大4台までの mornin' plus を使用することができます。

スマートフォンが近くになくても設定した時間になると動作するため、帰りの遅い日や長期旅行、出張中の防犯にも役に立つ。



図 2.2 めざましカーテン mornin' plus

問題点

防水ではないため、雨による水滴や、本体の結露などによって故障する可能性があるため、寒い日に暖房を入れるのを我慢するか、mornin' plus を取り外すかをしなければならない。起きるのが苦痛である冬に使うことができないという難点がある。

取り付け時の構造上、カーテンを完全に端まで動かすことができない。カーテンが開き切らないので全開にしたい人は気になってしまう。また、厚いカーテンでは、カーテンを端まで寄せると、カーテンの重さに mornin' の駆動力が負けてしまい、端のほうでカーテンが圧縮されカーテンレールを進まなくなる。

対応レールが決まっており対応していないレールだと使用することができない。また、両開きの場合は、mornin' plus は一つで一枚のカーテンにしか対応していないので2つ必要になる。天気が悪いと日光の量は減ってしまい、日光で目が覚めにくくなる。またそれだけでなく、建物や窓の位置によっては光の入りにくい場所は使用しても効果を得られない。

2.3 光の目覚まし時計 inti 4(インティフォー)

アラームを設定した時間に光るのではなく、アラームを設定した時間前に徐々に弱い光から10段階にかけて光を出して起こしてくれる目覚まし時計である。サンダースヌーズ機能という光が点滅する機能があり、アラームを設定した時間になると光の明るさがマックスになるが、そこから9分間、光を止めなかった場合にサンダースヌーズ機能が作動する。

人間の身体は、目で光を感知することによって、体内時計を調整しているため、朝日を浴びることで照度 2500 ルクス以上明るい光を目から取り入れることで活性化し、目覚めを促進させるホルモンであるセロトニンの分泌を促し、睡眠を促すホルモンであるメラトニンの分泌を抑制させる。セロトニンとメラトニンをコントロールすることで、朝と夜の区別をつけ、目覚めをすっきりさせてくれる。人がしっかり目覚めることができ、体内時計をリセットすることができるのは照度 2,500 ルクス以上とされているため、家庭の電気は一般的に 700 ルクス程度と言われており、2,500 ルクスにはとどかないため、家庭の電気ではスッキリと目覚めることができない。

朝日が差し込まない部屋や、カーテンが開けられない部屋、紫外線が気になる、日の出時間など太陽はコントロールできないため、人工太陽として非常に効果的である。

光で起きることができなかった場合のために、アラーム音も完備されている。



図 2.3 光の目覚まし時計 inti 4(インティフォー)

問題点

コンセントをつなぐ必要がありほかの目覚まし時計に比べ、多くの電力を消費してしまう。また、他用のためにコンセントを抜き、つなぎなおすのを忘れてしまうなど、安定性が要求される目覚まし時計にとって大きなデメリットとなってしまう。

顔に光が当たるようにしないといけないため設置場所が限られてくる。布団を顔までかぶってしまうと光がさえぎられてしまう。

2.4 爆音目覚まし時計 スーパーライデン

圧倒的なボリューム音の目覚まし時計である。最大音量が100デシベルであり、これは車のクラクションなどに相当する大きさである。音量レベルは3つあり、その日によって選ぶことができる。



図 2.4 爆音目覚まし時計 スーパーライデン

問題点

音が大きすぎるため、集合住宅で使用すると迷惑になってしまう。

びっくりして起きるため、目覚めが悪い。また、目覚まし時計が耳元で鳴ってしまうと、100デシベルは聴覚に支障をきたすレベルなため危険である。

全体の問題点

逃げ回る目覚まし時計以外は、目覚まし時計が手の届く範囲にあるため、スヌーズ機能がついていたとしてもすぐにとめることができず二度寝を防止する機能に欠けている。逃げ回る目覚まし時計は、どこに行くかがわからず止めるのが困難なのが売りではあるが、手の届かないところに行ってしまう貴重な朝の時間がなくなってしまう可能性があり、不安定である。

第三章 提案手法

既存技術の問題点を解決するため、人が起きるまでなり続ける目覚まし時計を提案する。

3.1 アラームを止めてしまう問題点について

二度寝を防止するために一番大切なのはスヌーズ機能をベッド内で人の手で止めるのではなく、人がベッドから起き上がったことを確認するとアラーム音が止まりスヌーズ機能も停止することである。そこで、人が起きたことをカメラで検知するとスヌーズ機能が止まる目覚まし時計が望ましい。しかし、人がベッド内で起きたかどうかをカメラで検知するためには、寝返りをうつ、布団が動くなど人の動きであるかどうか、ものの動きであるかどうかを見極めなければならない。さらに、人が起き上がり、もう一度寝てしまった場合の動きも検知しなければならないため、システムを作成することが困難であると予想する。また、ベッド内で検知してしまうとベッドから出ていないため、二度寝防止に効果が少ないと考えられる。

問題点を解決するためには、ベッドの枕元でアラーム音は鳴るが、アラーム音を止めるためには、ベッドの外に出なければならないという方法が望ましい。また、アラーム音を止めるために朝の時間を使うのはもったいないため、朝に必ず行う人の行動を検知することが望ましい。朝に行う人の行動例として、寝室のドアを開閉する、歯磨きをする、トイレに行く、顔を洗うなどである。よって、二度寝を防止するために、アラームを設定した時間になると、枕元のスピーカーに音が流れ、アラーム音を止めるには、上記に示した例のカメラにドアが動いたこと、歯ブラシが取られたことなど、人が行ったものの動きを検知すると停止する目覚まし時計を提案する。

移動物体の検出方法には、フレーム差分法や背景差分法、運動解析(モーションキャプチャー)などがある。フレーム差分法とは、連続する画像の差分から動体を検出することができ、背景画像を考慮する必要がない点が大きな特徴である[8]。背景差分法とは、入力画像と背景画像の差分を計算することで動体を検出する方法なため、フレーム差分法には必要なかった背景画像が必要となる[9]。運動解析とは、人やものの特徴点ごとの座標値を算出し、動きが加わった際の変化量を算出する画像処理技術のことである[10]。今回は、カメラを動かさず一定の場所に設置するため、背景画像を考慮することなく処理を行うことができる。よって、フレーム差分法を用いて行うことが最適であると考え。フレーム差分法の例として、まず、カメラで動画を撮影し、撮影した動画を一定間隔でフレームを3枚取得してグレースケールに変換する。次に、3枚のフレームの差分の絶対値を計算し、2枚の論理積を計算し、論理積画像を作成する。論理積画像に二値化処理を行い、背景である黒色と前景である白色に分けたマスク画像を作成し、マスク画像にある白色の画素数が閾値以上であるならば動体検知が行われたと認識する。

図 3.1 にフレーム差分法を用いた動体検知のフローチャートを示す。

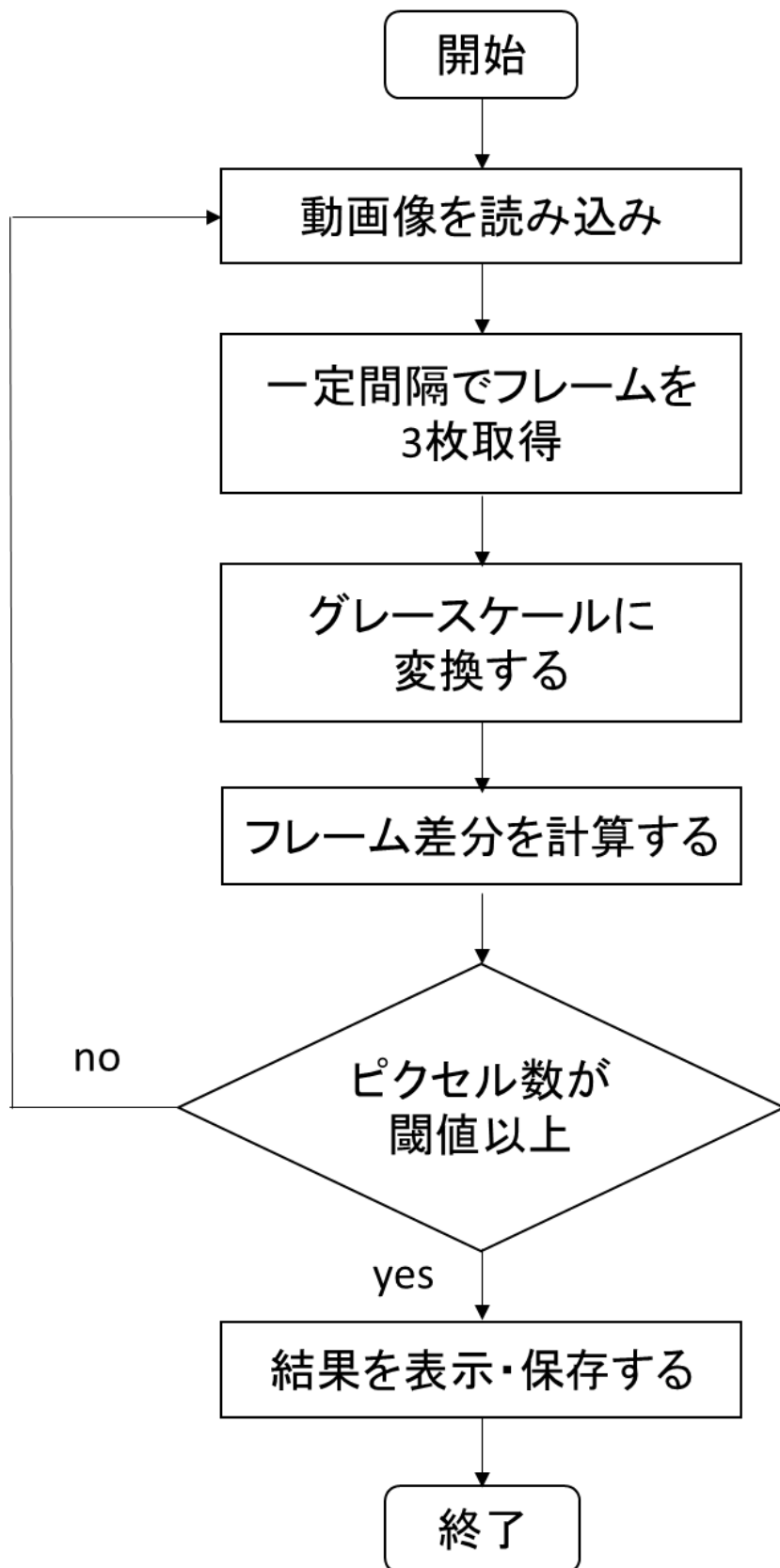


図 3.1 フレーム差分法を用いた動体検知のフローチャート

3.2 アラーム音が聞こえない問題点について

アラーム音の音色を変えることによって起床のしやすさに影響が出るという[11]。聞きなれている音や心理的に影響を与えない音は起床時に聞こえにくく、日常的に聞きなれていない音や心理に影響を与える音は起床を促しやすい[11]。しかし、日常的に聞きなれていない音や心理に影響を与える音は、不快と感じるからこそすぐにアラームを止め、二度寝してしまうという傾向が多く見られる[12]。この傾向は、ベッドの近くに目覚まし時計がある場合には問題点となるが、提案している目覚まし時計では、ベッドから離れた場所に目覚まし時計を置き、起き上がらなければアラームを止めることができないことが前提であるため問題点とはならない。不快と感じ、すぐにアラームを止めてしまう傾向と提案している目覚まし時計は相性が良いと考えられる。よって、日常的に聞きなれていない音や心理に影響を与えやすい音をアラーム音として使うことが望ましいと考える。

使用するアラーム音は、心理に影響を与えやすいと考えられる人が不快に感じる虫の羽音や咀嚼音、歯医者で歯を削られる時の不快音を鳴らす[13]。しかし、いきなり不快音では目覚めが悪いため、起きなければ遅刻するぎりぎりの時間に不快音を鳴らし、余裕をもって起きることができる時間帯にはさざ波や鐘の音など目覚めよく起きられる音を鳴らす。

よって、アラームが設定された時間になると、快適なアラーム音が鳴り、人が起きているかを調べるために動画を撮影し動体検知を行い、一定時間内に動体検知が行われなかった場合、不快なアラーム音を鳴らすという目覚まし時計を開発する。

図 3.2 に作成予定の目覚まし時計のフローチャートを示す。

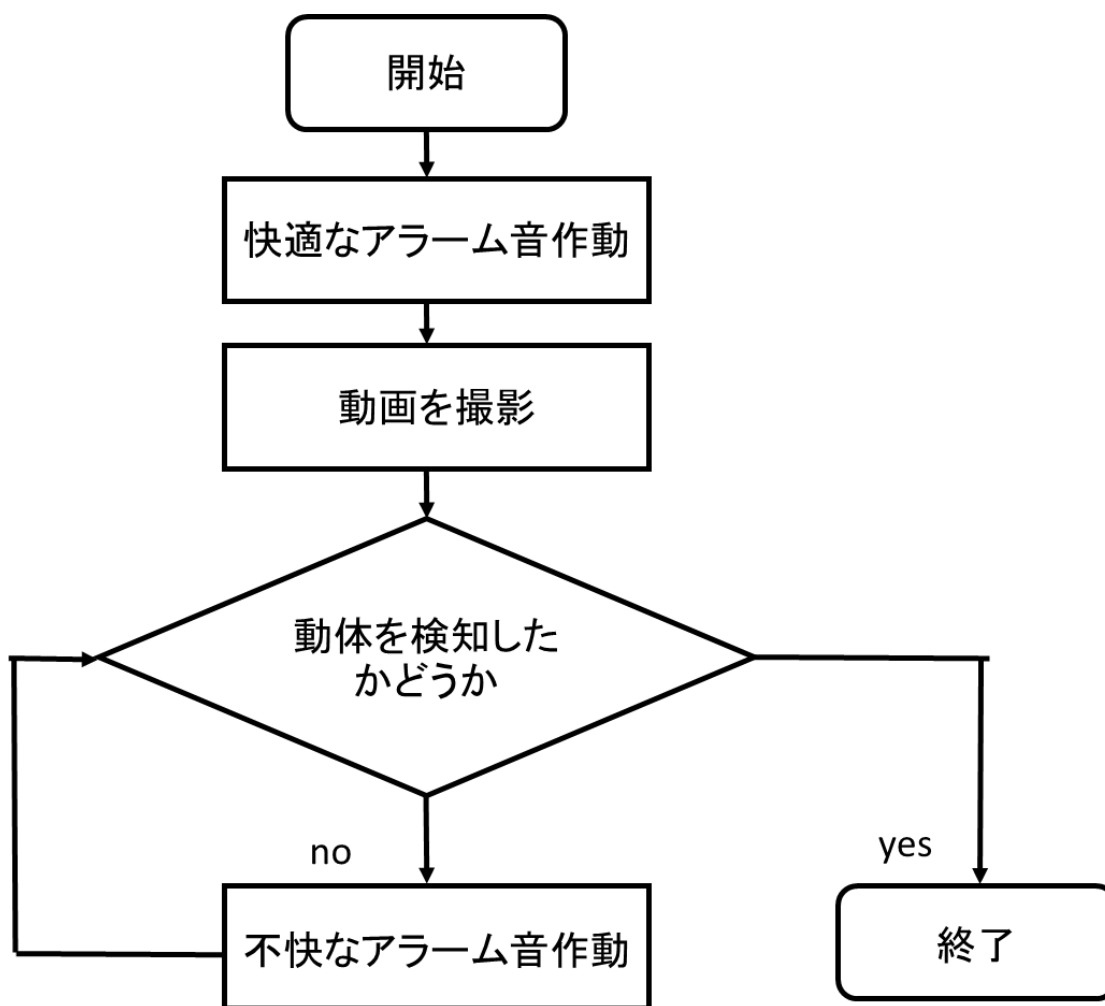


図 3.2 作成予定の目覚まし時計のフローチャート

3.3 提案システム

ベッドの枕元でアラーム音は鳴るが、アラーム音を止めるためには、ベッドの外に出なければならないという方法が望ましく、また、アラーム音を止めるために朝の時間を使うのはもったいないため、朝に必ず行う人の行動を検知することが望ましい。よって、二度寝を防止するために、アラームを設定した時間になると、枕元のスピーカーに音が流れ、アラーム音を止めるには、カメラにドアが動いたこと、歯ブラシが取られたことなど、人が行ったものの動きを検知すると停止する目覚まし時計を提案する。アラーム音は、はじめは快適なアラーム音が鳴り、一定時間内に動体検知が行われなかった場合、不快なアラーム音を鳴らす。これにより、二度寝を防止し、心地よく目覚めることができる。

下記にワンルームでの使用例の図を示す。スマートフォン、スピーカー、カメラは無線で接続されているものとする。アラームのセットはスマートフォンで行い、カメラは寝室のドアと洗面台の歯ブラシにセットし、寝室のドアが開く、洗面台の歯ブラシを取る行動が順に行われるまでスピーカーから流れるアラーム音は止まらない仕組みである。歯ブラシを取る、寝室のドアが開く順で行われた場合は目覚まし時計を使用している本人ではないと認識されるため、アラームは止まらない。カメラは、今回の例では2台であるが、一人暮らしの場合、家族と一緒に暮らしている場合などで設置する場所や、設置する台数を適切なものに変更することが可能である。

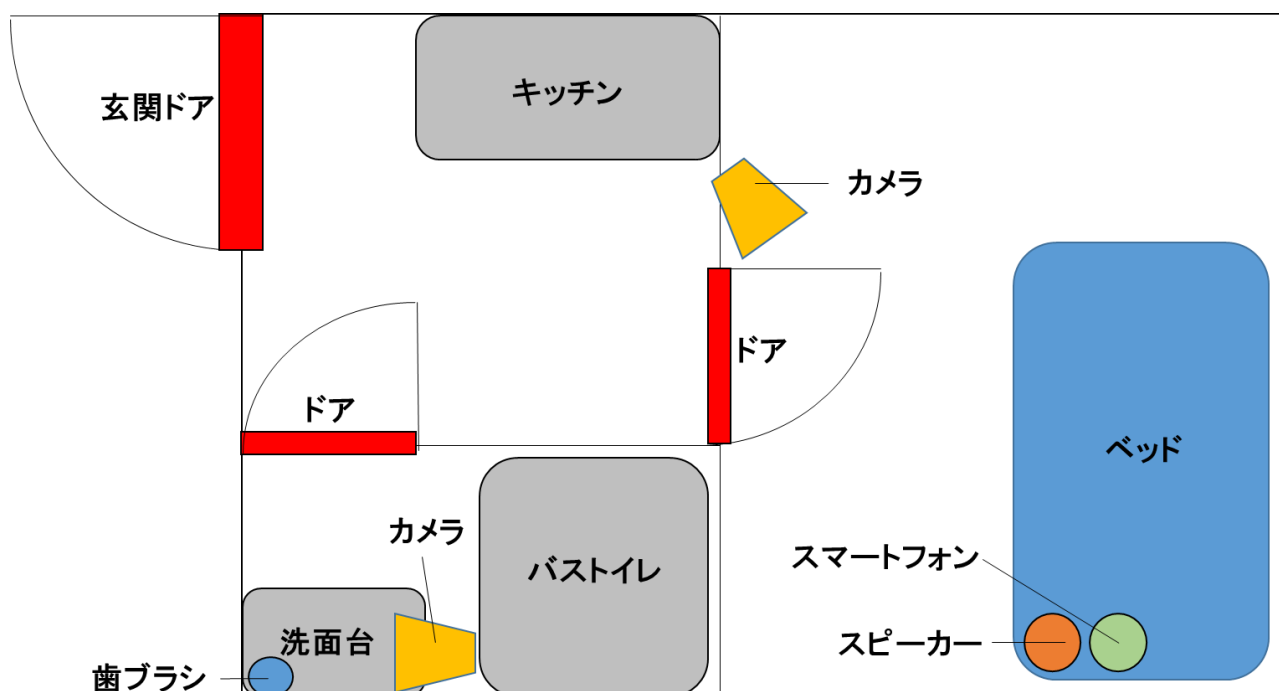


図 3.3 設置図の例

第四章 実験

4.1 実験目的

本実験では、人が起きたことを認識するために必要である動体検知を動画画像から静止画像を抽出して正確に認識することができるかを確認するために行った。人が朝起きて必ずする行動として、「寝室のドアを開ける」、「洗面台にある歯ブラシを取る」の2つの例の実験を行った。

4.2 実験内容

4.2.1 実験環境

本実験では、ラズベリーパイとラズベリーパイのカメラモジュールを使用し動体検知を行うためのプログラムをOpenCVを用いて作成した。詳細と図を下記に示す。

- ・ハード : Raspberry Pi 3 Model B_Single (品番 : 1225826)
Raspberry Pi PiNoir Camera V2 (品番 : 9132673)
キーボード
マウス
ディスプレイ
- ・OS : Raspbian GNU/Linux 9.4 (stretch)
- ・ソフト : OpenCV

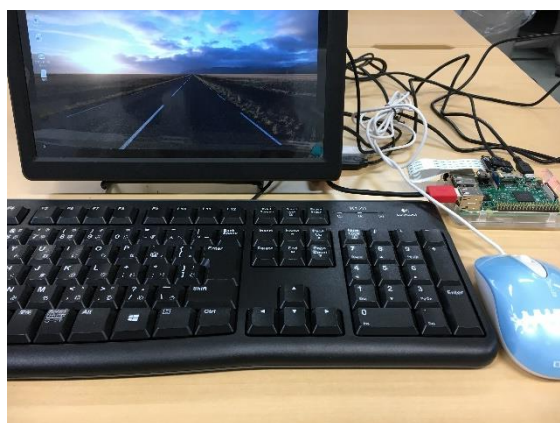


図 4.1 実験環境の全体図



図 4.2 ラズベリーパイとカメラモジュール

4.2.2 寝室のドアを開ける実験内容

「寝室のドアを開ける」実験では、三好研究室の入り口のドアを寝室のドアと仮定して実験を行った。ドア付近にある棚にラズベリーパイを設置し撮影を行った。ドアを90度まで開く動画を撮影し、動画像から静止画像を抽出し正確に動体検知を行うことができるかどうかを検証した。今回の実験では、動画像の撮影時間は5秒と設定する。図4.1に実験の様子を図で示す。また、動体検知を行ったと判定する閾値を設定し、白色領域のピクセル数が閾値以上となった場合に画像を保存するようにプログラムを作成した。閾値は、10、50、100、300、500、700、1000、1500の8つを比較した。

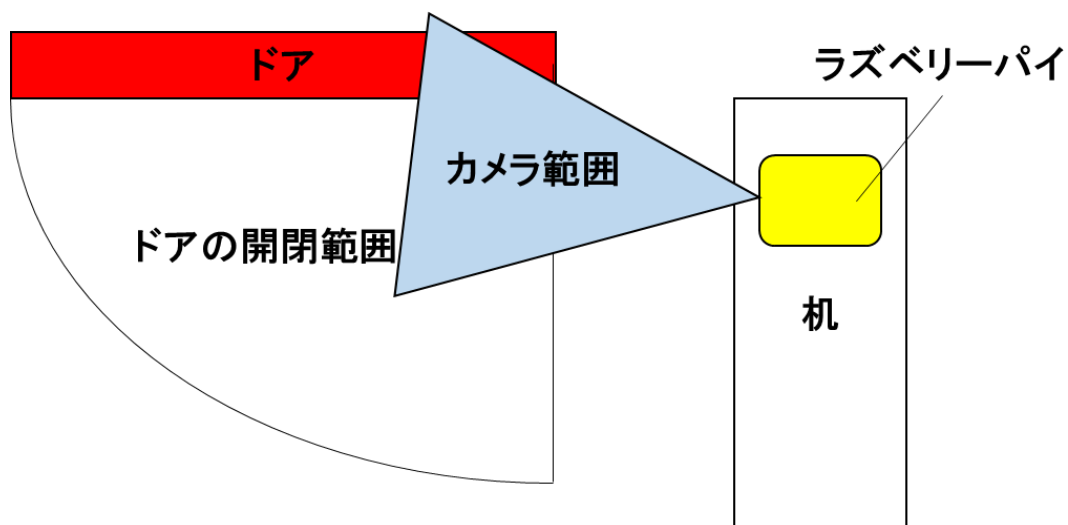


図 4.3 寝室のドアを開ける実験環境(上から見た図)

4.2.3 洗面台にある歯ブラシを取る実験内容

「洗面台にある歯ブラシを取る」実験では、三好研究室にあるロッカー横の台所を洗面台と仮定して実験を行った。流し台の横に歯ブラシを設置し、その横にラズベリーパイを設置し撮影を行った。今回の実験では、歯ブラシの代わりにペンを代用した。歯ブラシを取る動画像を撮影し、動画像から静止画像を抽出し、4.2.1の実験よりもものの動きが小さく、一瞬であるが正確に動体検知を行うことができるかどうかを検証した。図4.2に実験の様子を図で示す。今回の実験では、動画像の撮影時間は5秒と設定する。また、動体検知を行ったと判定する閾値を設定し、白色領域のピクセル数が閾値以上となった場合に画像を保存する。閾値は、10、50、100、300、500、700、1000、1500の8つを比較した。

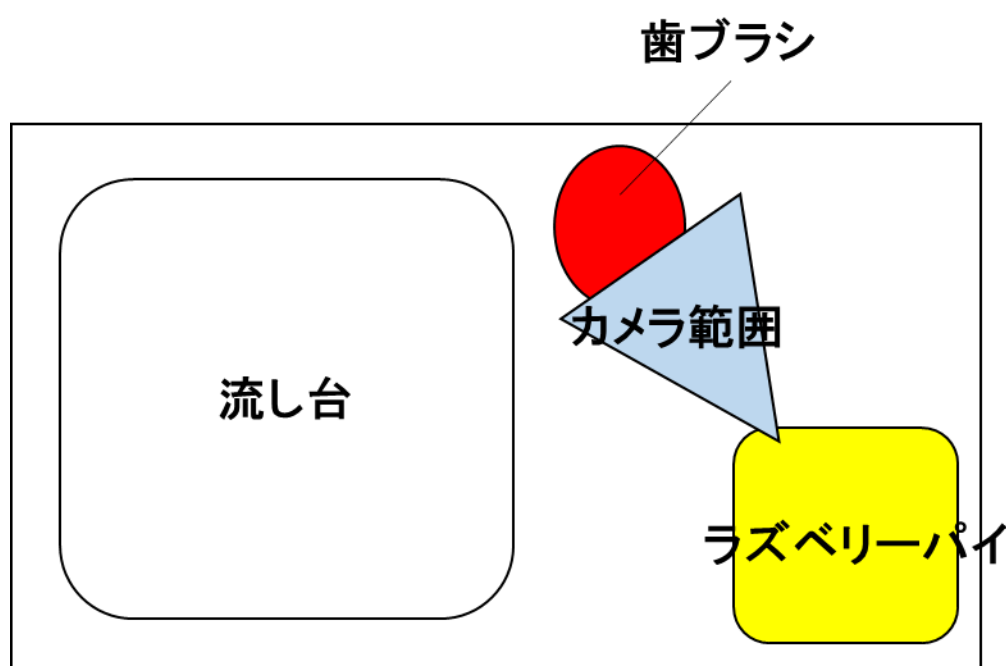


図 4.4 洗面台にある歯ブラシを取る実験環境(上から見た図)

4.3 実験結果

4.3.1 寝室のドアを開ける実験結果

まずは、ドアを開けたときの像画像から取得されたグレースケールのフレームを図 4.5-図 4.16 に示す。動画の撮影時間は5秒である。



図 4.5 フレーム 1



図 4.6 フレーム 2



図 4.7 フレーム 3



図 4.8 フレーム 4



図 4.9 フレーム 5



図 4.10 フレーム 6



図 4.11 フレーム 7



図 4.12 フレーム 8



図 4.13 フレーム 9



図 4.14 フレーム 10



図 4.15 フレーム 11



図 4.16 フレーム 12

以上の 12 枚のフレームが取得された。次に、白色領域のピクセル数が閾値 1000 以上のフレームを図 4.17-図 4.27 に示す。



図 4.17 フレーム 1



図 4.18 フレーム 2



図 4.19 フレーム 3



図 4.20 フレーム 4

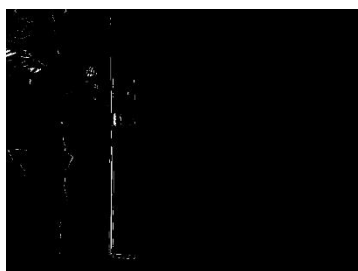


図 4.21 フレーム 5

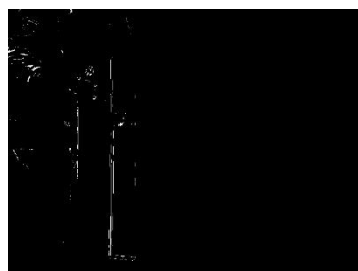


図 4.22 フレーム 6

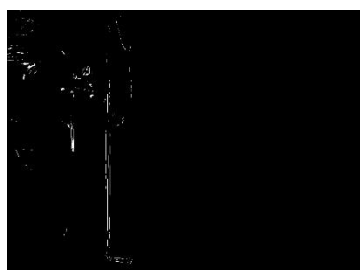


図 4.23 フレーム 7

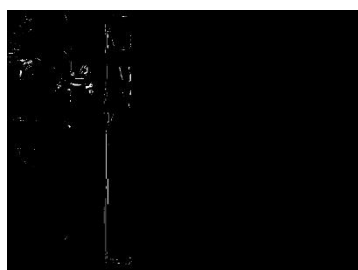


図 4.24 フレーム 8

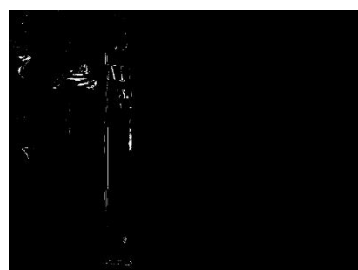


図 4.25 フレーム 9

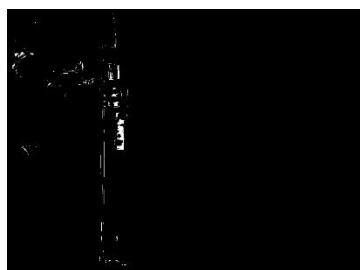


図 4.26 フレーム 10



図 4.27 フレーム 11

表 4.1 閾値別の閾値以上のフレーム数

閾値	フレーム数
10	103
50	91
100	58
300	30
500	20
700	16
1000	12
1500	0

変化のない動画像では閾値に関係なくフレーム数は0となった。

閾値 10、50、100、300、500、700、1000 では、白色領域のピクセル数が閾値以上の場合にフレームが保存されているため、動体検知が正しく行われていることがわかる。閾値 1500 では、白色領域のピクセル数が閾値以上の場合にフレームが保存されていないため、動体検知が正しく行われていないことがわかった。

4.3.2 洗面台にある歯ブラシを取る実験結果

まずは、洗面台にある歯ブラシを取る像画像から取得されたグレースケールのフレームを図 4.28-図 4.32 に示す。動画の撮影時間は 5 秒である。



図 4.28 フレーム 1



図 4.29 フレーム 2



図 4.30 フレーム 3



図 4.31 フレーム 4



図 4.32 フレーム 5

以上の 5 枚のフレームが取得された。次に、白色領域のピクセル数が閾値 500 以上のフレームを図 4.33-図 4.36 に示す。

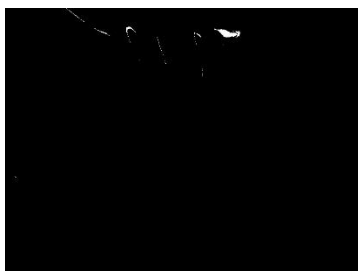


図 4.33 フレーム 1

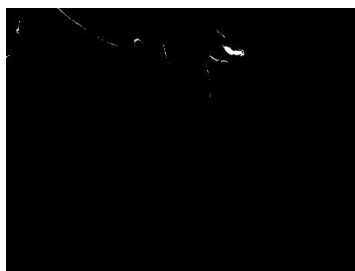


図 4.34 フレーム 2

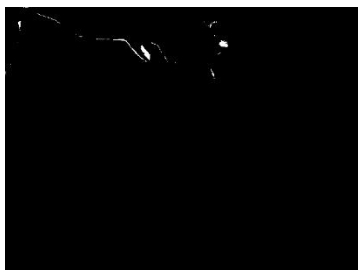


図 4.35 フレーム 3

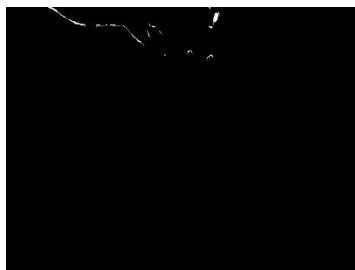


図 4.36 フレーム 4

表 4.2 閾値別の閾値以上のフレーム数

閾値	フレーム数
10	21
50	15
100	9
300	7
500	5
700	0
1000	0
1500	0

変化のない動画像では閾値に関係なくフレーム数は0となった。

閾値 10、50、100、300、500 では、白色領域のピクセル数が閾値以上の場合にフレームが保存されているため、動体検知が正しく行われていることがわかる。閾値 700、1000、1500 で実験を行うと白色領域のピクセル数が閾値以上の場合にフレームが保存されていないため、動体検知が正しく行われていないことがわかった。

第五章 考察

寝室のドアを開ける実験と歯ブラシを取る実験では、静止画像中の変化の大きさに違いがある。寝室のドアを開ける実験は静止画像中の変化が大きい、歯ブラシを取る実験は静止画像中の変化が小さい。これら2つの実験より、静止画像中の変化が大きい場合は、閾値が大きい場合でもフレーム数を獲得することができるが、静止画像中の変化が小さい場合は、静止画像中の変化が大きい場合に比べてフレーム数を獲得することができる閾値は小さくなる。よって、静止画像中の変化が小さいと考えられる歯ブラシを取る動作のときに正しく動体検知している閾値を常時設定しておくことですべての動作を正しく認識することができるため、閾値は500以下が適切である。実験結果より、提案手法の有効性が示された。

問題点として三つの問題点が挙げられた。まず、一つ目の問題点として、目覚まし時計を使用している本人でない人がドアを開けてしまった場合、歯ブラシを取ってしまった場合は、人を識別して行っていないため、動体検知をしたと認識してしまい、アラーム音が止まってしまうという問題点がある。一人暮らしでない場合には、間違っただけで目覚まし時計を使用している本人ではない他の人の行動でアラーム音が止まらないようにするために、朝起きて必ずする行動や朝起きてする行動順が限られてしまう。家族などの同居人がいる場合、アラーム音が鳴ったときに同居人に起こしてもらおうことも可能ではあるが、同居人に一度起こしに来てもらい、その場で起きずに二度寝してしまった場合などの最悪のケースも考えられる。また、大きなものが落ちた場合などにも動体検知したと認識してしまい、これらの問題点を解決するには、個人認証をすることで解決されるのではないかと考える。

二つ目の問題点として、歯ブラシを取るなどの静止画像中の変化が小さい行動は、動体検知したと認識されにくいいため、閾値を変更しなければならない問題点がある。その人の朝起きたときに必ずする行動によっては閾値を変更しなければならないため、自動で閾値を変更するプログラム作成する必要があると考える。

三つ目の問題点として、カメラの置き方や台数である。カメラの置き方は目覚まし時計を頻繁に使用する朝起きて必ず行う行動に合わせて正しい位置に設置している。そのため、仕事や学校までに少しだけ睡眠をとりたい場合や昼寝など、朝ではないときに目覚ましをセットしても朝起きたときと同じ行動をしなければアラーム音は止まらない。また、昼寝などの場合は、ベッドでは寝ずにリビングのソファなどで寝ることも考えられる。この問題を解決するには、起床後の動作について検討していかなければならないと考える。

第六章 まとめ

本研究では、朝を二度寝せずに快適に起きるための覚醒検知機能を持った目覚まし時計について考えた。既存の目覚まし時計では二度寝に対する機能に劣っており、二度寝をしてしまう人が多いため、ラズベリーパイとカメラを用いて動画像を撮影し、ベッド外で動体検知を行う手法を提案した。実験結果から、フレーム差分法を用いた動体検知を正しく認識することに成功した。今後の課題は、アラーム音をはじめは心地の良い音から徐々に不快音に変えていくシステムなど、その人にとって最適なアラーム音を設定できるようにすることである。

謝辞

本研究を進めるにあたり、ご指導を頂いた卒業論文指導教員の三好力教授に感謝いたします。また、日常の議論を通して多くの知識や示唆を頂いた三好研究室の皆様、友人、知人の皆様に感謝いたします。

参考文献

[1] 時計についてのアンケートランキング

http://chosa.nifty.com/hobby/chosa_report_A20140110/7/

[2] 寝坊対策はここまでやる！絶対に寝坊しない目覚ましグッズまとめ

<https://nurse-web.jp/suimin/oversleep/>

[3] みんなの“目覚め”を調査しました

<http://nemuri-lab.jp/enquete/research/481/>

[4] 逃げ回る目覚まし時計 NANDACLOCKY(ナンダクロッキー)

<https://gigazine.net/news/20150705-nandaclocky/>

[5] めざましカーテン mornin' plus

<https://mornin.jp/>

[6] 光の目覚まし時計 inti 4(インティフォー)

<https://www.namaiki-100.com/entry/hikarimezamasii.inti4>

[7] 爆音目覚まし時計 スーパーライデン

<https://gigazine.net/news/20150718-super-raiden/>

[8] フレーム差分法

<https://algorithm.joho.info/image-processing/frame-difference-method/>

[9] 背景差分法

<https://algorithm.joho.info/image-processing/background-subtraction-method/>

[10] 運動解析(モーションキャプチャー)

<http://www.kk-co.jp/visible/motion.php>

[11] アラーム音の音色の違いが起床に与える影響

http://www.ias.sci.waseda.ac.jp/GraduationThesis/2012_summary/1W090033_s.pdf

[12] 寝坊しない方法

<https://matome.naver.jp/odai/2142595620102256801/2142595706803326203>

[13]目が覚めるスマホのアラーム音とは

<http://www.standby-media.jp/news/124890>

[14]動体検知

<https://algorithm.joho.info/programming/python/opencv-frame-difference-surveillance-camera-py/>

付録

```
目覚まし時計
#!/bin/bash
#アラーム設定
#何秒後に鳴る
python
/home/pi/AlarmClock/move/siyou/aram.py 5
#音楽を再生
python
/home/pi/AlarmClock/move/siyou/ongaku2.py
&
# -e:存在しているかどうか
if [ -e frame0.jpg ]; then
#写真を削除
# rm:削除コマンド
rm
/home/pi/AlarmClock/move/siyou/frame0.jpg
echo "写真を削除します"
fi
while [ ! -e
/home/pi/AlarmClock/move/siyou/frame0.jpg
]
do
#動画を撮影する
python
/home/pi/AlarmClock/move/siyou/video.py
#動画を動体検知する
python
/home/pi/AlarmClock/move/siyou/test.py
done
echo "起きてます"
#バックグラウンドをすべて終了する
if [ "$(jobs -p)" ]; then kill $(jobs -p);
fi
```

動画を撮影

```
import time
import picamera
camera = picamera.PiCamera()
camera.resolution = (640, 480)
camera.start_preview()
#time.sleep(1)
camera.start_recording('video.h264')
camera.wait_recording(5)
camera.stop_recording()
camera.stop_preview()
```

```
アラームの時刻を設定する
# coding:utf-8
import time, sys
timing_limit = int(sys.argv[1])
while timing_limit >= 0:
    if timing_limit == 0:
        print "朝だよ"
    else:
        print timing_limit
        timing_limit -= 1
        time.sleep(1)
```

アラーム音の設定

```
# coding:utf-8
from mutagen.mp3 import MP3 as mp3
import pygame
import time
filename = 'FF7.mp3' #再生したい mp3 ファイル
pygame.mixer.init()
pygame.mixer.music.load(filename) #音源を
読み込み
mp3_length = mp3(filename).info.length #音
源の長さ取得
pygame.mixer.music.play(1) #再生開始。1 の
部分を変えると n 回再生(その場合は次の行の
秒数も×nすること)
time.sleep(mp3_length + 0.25) #再生開始後、
音源の長さだけ待つ(0.25 待つのは誤差解消)
mp3_length + 0.25
pygame.mixer.music.stop() #音源の長さ待っ
たら再生停止
```

```

動体検知
#!/usr/bin/env python
# -*- coding:utf-8 -*-
# coding= shift_jis
import cv2
import numpy as np
# フレーム差分の計算
def frame_sub(img1, img2, img3, th):
    # フレームの絶対差分
    diff1 = cv2.absdiff(img1, img2)
    diff2 = cv2.absdiff(img2, img3)
    # 2つの差分画像の論理積
    diff = cv2.bitwise_and(diff1, diff2)
    # 二値化処理
    diff[diff < th] = 0
    diff[diff >= th] = 255
    # メディアンフィルタ処理 (ゴマ塩ノイズ除去)
    mask = cv2.medianBlur(diff, 5)
    return diff
def main():
    # 不審物判定の閾値
    min_moment = 100
    # カメラのキャプチャ
    cap = cv2.VideoCapture('video.h264')
    # フレームを3枚取得してグレースケール変換
    frame1 = cv2.cvtColor(cap.read()[1],
cv2.COLOR_RGB2GRAY)
    frame2 = cv2.cvtColor(cap.read()[1],
cv2.COLOR_RGB2GRAY)
    frame3 = cv2.cvtColor(cap.read()[1],
cv2.COLOR_RGB2GRAY)
    # カウント変数の初期化
    cnt = 0
    while(cap.isOpened()):
        # フレーム間差分を計算
        mask = frame_sub(frame1, frame2,
frame3, th=50)
        # 白色領域のピクセル数を算出
        moment = cv2.countNonZero(mask)
        # 白色領域のピクセル数が一定以上
        なら不審物有りと判定
        if moment > min_moment:
            print "動体検知された:"
            print (cnt)
            #保存するファイルの名前
            filename = "frame" + str(cnt)
+ ".jpg"
            filename2 = "mask" + str(cnt)
+ ".jpg"
            #ファイルを保存する
            cv2.imwrite(filename2, mask)
            cv2.imwrite(filename, frame2)
            cnt += 1
        # 結果を表示
        cv2.imshow("Frame2", frame2)
        cv2.imshow("Mask", mask)
        # 3枚のフレームを更新
        frame1 = frame2
        frame2 = frame3
        frame3
cv2.cvtColor(cap.read()[1],

```

```

cv2.COLOR_RGB2GRAY)
        # qキーが押されたら途中終了
        if cv2.waitKey(1) & 0xFF ==
ord('q'):
            break
        cap.release()
        cv2.destroyAllWindows()
if __name__ == '__main__':
    main()

```