

Critter & Guitari

EYESY™



日本語ユーザーズ・ガイド

株式会社アンブレラカンパニー

www.umbrella-company.jp

*本書は株式会社アンブレラカンパニーのオリジナル制作物です。
無断での利用や公開、複製を固く禁じます。

For EYESY OS version 2.2.

内容

EYESY™ ユーザーマニュアル.....	1
はじめに.....	3
同梱物	3
ビジュアルを映し出そう！	3
重要なご注意	4
1. EYESY のコンセプト.....	5
この不思議なアイコンも持ったボックスは一体何？！	6
1.1 EYESY はどこから来たのか?.....	6
1.2 EYESY のコンセプト.....	8
1.3 このマニュアルの使い方.....	9
2. EYESY の操作方法	10
ノブとボタンを知るとはたいへん価値があります！	10
2.1 システムコントロール.....	10
2.2 モード・パラメーターのコントロール	15
2.3 ファンクション・コントロール.....	17
リアパネル.....	19
サイドパネル(左側).....	21
3. EYESY の MIDI 構成について	22
4. EYESY へのアクセス.....	26
WiFi, WiFi, WiFi	26
4.1 アクセスポイント vs 既存ネットワークへの参加.....	26
4.2 エディターの説明.....	29
4.3 マイクロ SD カードのフォルダ/ファイル管理	31
5. EYESY のプログラミング	36
テキストエディタと夢を持っている人のために！	36
5.1 モードスクリプトについて	37
5.2 EYESY の API について.....	39
6. 付録資料.....	43
6.1 SD カードのディスクイメージの書き込み	43
6.2 ビデオキャプチャについて	44
6.3 WiFi と SSH の設定について.....	45

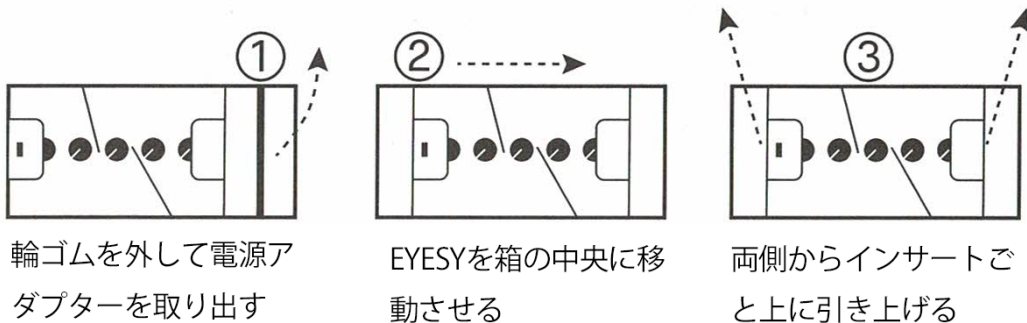
はじめに

EYESY の世界へようこそ！さあ、始めましょう！

同梱物

ボックスには 4 つのアイテムが入っています。製品箱から EYESY を取り出すときは以下のイラストを参考にしてください。

- クイックスタートガイド&保証書
- EYESY 本体
- USB-WiFi アダプター
- 電源アダプター. 本機専用です

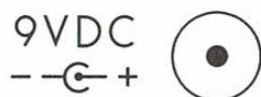


ビジュアルを映し出そう！

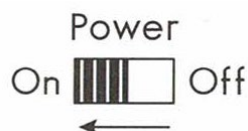
次の手順に従って、サウンドをビジュアル化します。

1. EYESY の HDMI®またはコンポジット端子から、モニター、プロジェクター、ビデオミキサー、ビデオキャプチャデバイス、またはその他のディスプレイに接続します。
2. 1/4 インチのモノラルまたはステレオオーディオケーブルをオーディオ入力ポートに接続します。

3. 電源アダプタを電源コンセントに接続し、そのプラグを EYESY の背面の左端のポートである 9VDC 電源ポートに接続します。必ず付属の AC アダプタを使用してください(極性はセンタープラスです！)



4. 電源スイッチを ON の位置に設定します。一番上の行のステータス LED は、起動時に薄緑色に点灯してから青色に点滅します。接続されたディスプレイに起動プロセスが表示されます。



5. 一番上の行の[オンスクリーンディスプレイ]ボタンを押します(そのアイコンは、一番下に文字が書かれた画面を示しています)。これで、ディスプレイに現在のシステム情報のオーバーレイが表示されます。ボタンをもう一度押すと、この表示をオフに切り替えることができます。
6. オーディオをビデオに変換するには、接続されているデバイスからオーディオを送信し、1/4 "オーディオケーブルをオーディオ入力に接続します。音源の音量を上げたり、または、Shift ボタンを使用することでゲインを調整できます。詳細については、第 2 章を参照してください。
7. 取り急ぎ十分に楽しんだら、電源スイッチをオフの位置にスライドさせます。電源を切断する前に、ステータス LED がオフになるまで必ず待ってください。

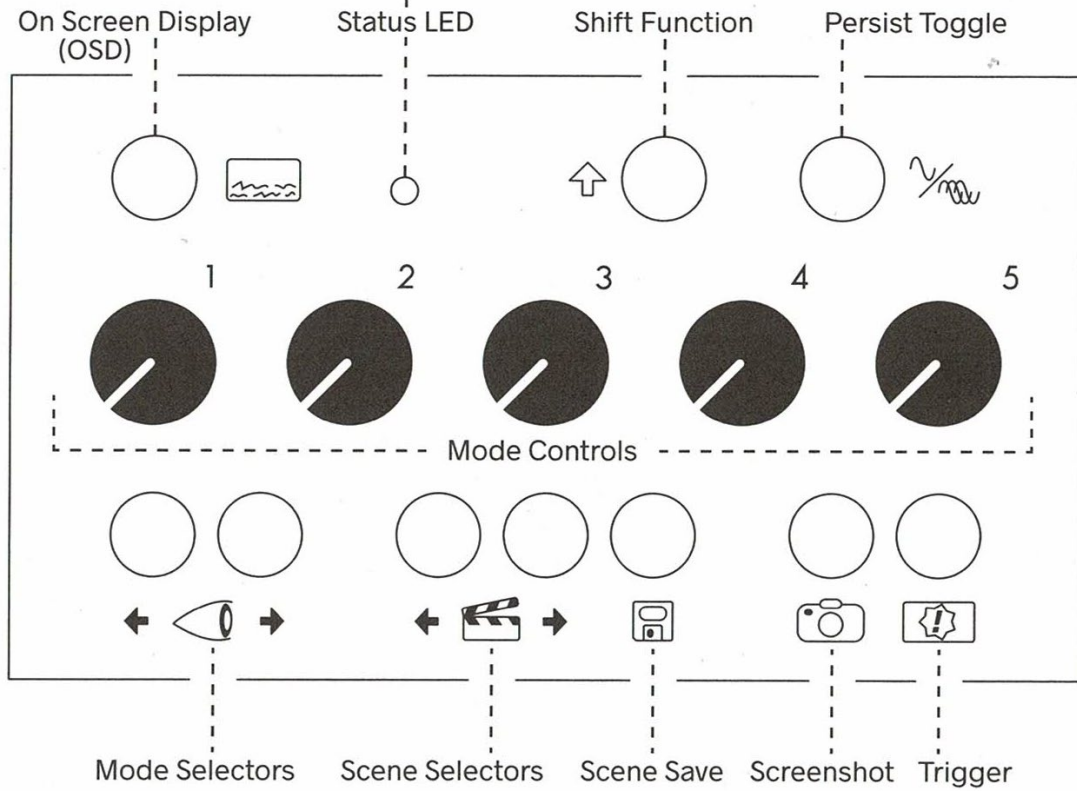
重要なお注意

1. このユーザーガイドと印刷されたすべての警告に注意してください。
2. ユニットの修理はユーザー自身が行うことはできません。必ずメーカーや代理店にお問い合わせください
3. この製品は、感光性発作を引き起こす可能性のある画像を生成する可能性があります。何らかの不快感、方向感覚の喪失、または身体への悪影響を経験した場合は、直ちに使用を中止し、医師に相談してください。

Control Layout:



Blue: Starting Up
White: Running
Yellow: Screenshot
: Green: MIDI Received :



1. EYESY のコンセプト

この不思議なアイコンも持ったボックスは一体何？！

EYESY の世界へようこそ！ このビデオ機器は様々な方法と可能性を秘めています。実際に使用されている基本的な概念を学んでいきましょう。

1.1 EYESY はどこから来たのか？

Critter & Guitar は過去にもビデオシンセサイザーをリリースしており、その使用目的はシンプルでした：

1. オーディオ入力を接続し、ライブサウンドを供給。
2. プロジェクターまたはその他のディスプレイのライブビデオを出力。
3. いくつかのノブを設定します。
4. 音楽の再生ごとにライブビジュアルが展開。

EYESY はこのような「セルフサービス」のアプローチを引き継いでいますが、他の多くのモードとオプションが存在します。

EYESY はすぐに使用できますか？

Yes! EYESY を接続して電源を入れると(第 0 章を参照)、ファクトリモードでビジュアル生成が自動的にスタートされます。

EYESY でビデオを生成する異なる手法がありますか？

Yes! フロントパネルの左下(眼球アイコンの周り)にある 2 つのモードセクタボタンを押すと、EYESY の現在のビジュアルモードを前後にシフトすることができます。プリロードされたファクトリモードは、ユニークな視覚効果を提供します。詳細については第 4 章を参照してください。

EYESY をビデオ・インストゥルメントとして演奏できますか？

Yes! EYESY のトップパネルにあるノブとボタンを使用すると、EYESY の動作と各ビジュアルモードに直接アクセスできます。コントロールオプションを拡張するには、MIDI コントローラーを MIDI 入力ポートに接続するか、USB-MIDI デバイスを使用します。

EYESY をプラグインしてベースの演奏に戻ることができますか？

Yes! オーディオが EYESY に供給されている場合は、ノブを少し調整するだけで、レスポンスビジュアルの安定したストリームが得られます。

EYESY を MIDI でコントロールできますか？

Yes! EYESY は MIDI ノートとクロックに対応しています。DAW や Ableton Live のようなソフトウェアを使用している場合は、DAW から EYESY に MIDI を送信して、EYESY のパラメーターを直接制御でき、音楽に同期させることができます。つまり、EYESY を使用して、再生している音楽に追従したビデオ伴奏をクリエイトできます。

EYESY で生成された静止画像をキャプチャできますか？

Yes! EYESY の一番下の列には、専用のスクリーンショットボタン(カメラアイコン)があります。このボタンを押すと、EYESY の現在の画像が画像ファイルとして保存されます。またパッチをハックすれば、最後にキャプチャしたスクリーンショットを現在の出力に戻すモードを作成することもできます！

静止画を EYESY への素材として利用できますか？

Yes! モードのファイルから静止画像を使用可能です。

EYESY のモードを編集できるのですか？

Yes! EYESY のすべてのモードは、Pygame のグラフィックライブラリを使用して Python で記述されています。これは、正しい Python コードを学べばすべてのモードを編集できることを意味します。つまりオリジナルのモードを作成したり、誰かが作ったシェアパッチをダウンロードして利用する事もできます。

ということは、EYSY には決まった使い方はないということですか？？

Exactly ! EYESY はビジュアルを作成するための手段です。ライブパフォーマンス環境でそれが必要な場合もあれば、出力をキャプチャして編集したい場合もあります。他の要素から EYESY を操作してもらいたい場合もあります。テキストエディタ

を使用してカスタムモードをコーディングすることもできますが、他の人が作成した新しいモードを簡単にダウンロードする事もできます。目的にかなうものであれば何でもありという事です！

1.2 EYESY のコンセプト

EYESY のコンセプトについて、特にモードとシーンについて説明していきます。

EYESY はモードによってビジュアルを作成します。各モードはパッチと考えることができます。モードはフロントパネルから瞬時に切り替えることが可能であり、使用可能な各モードは、microSD カードに必要なサポートファイルとともに格納されています。

現時点でモードは 2 つの大きなカテゴリに分類されています。

- *scope mode* (スコープモード) は、入力されるオーディオ信号を直接ビジュアル化します。これは、古典的な「オシロスコープ」的なものですが、出力されるビジュアルは遥かに異なるものです。アニメーションは受信オーディオによって継続的に更新されていきます。
- *trigger mode* (トリガーモード) は、入力されるオーディオ信号をビジュアル化しますが、スコープモードのように継続的に描画することはありません。代わりに、オーディオ入力の音量が特定のスレッショルド値を超えると(たとえば、スネアドラムの打点を想像してください)ビジュアルがトリガーされます。MIDI ノートやクロッキングメッセージなどの他のイベントからもビジュアルをトリガーできます。また、トリガーボタンを使って手動でトリガーを生成することもできます。

モードの違いを大別するために、名前が S-で始まるファクトリモードはスコープモードで、T-で始まるファクトリモードはトリガーモードです。しかし、モードはこれらのアイデアを簡単にブレンドしたり、まったく新しい方向にモディファイする事も可能です。ファクトリーモードを理解することで、新しいモードや新しいカテゴリに関する独自のアイデアにも役立ちます。

EYESY のコントロールは、いくつかのカテゴリに分類されます。

- **Visual Controls (ビジュアルコントロール)** ローカルモードのパラメータを変更します。トップパネルの 5 つの番号付きノブで構成されるこれらのコントロールは、現在ロードされているモード内で定義されている設定に影響します。

- **Global Controls(グローバルコントロール)**ビジュアルコントロールより上のレベルで機能するため、モードを変更しても、これらのパラメーターは永続的に機能しています。これらには、オンスクリーンディスプレイ、パーシストグル、Shift ボタン、およびスクリーンショットボタンが含まれます。

EYESY のシーンは基本的にプリセットと同じです。各シーンには、使用されていたモードと、5つのノブすべての値、および[パーシストグル]ボタンの有効/無効の情報が保存されます。

次の章では、これらすべてのアイデアを実行に移す方法を探りていきます。モードとシーンを理解することで、着実に実行できるようになります。

1.3 このマニュアルの使い方

- 第1章のクイックスタート(第0章)からお読みください。
- 第2章にあるハードウェアの操作に関する情報は中心的なガイドとなります。さまざまなパネルコントロールとポートの機能について説明します。
- 追加のモードを EYESY にロードする場合は、第4章が役立ちます。I
- モードを編集したり、独自のモードを作成したりする場合は、第5章で説明しています。

EYESY では、今日の使い方が1週間後の使い方とは大きく異なる可能性があることを理解してください。つまり、今日は役に立たない情報はスキップしてかまいません。後の章では、時間の経過とともに価値が高くなるのがわかるかもしれません。

2. EYESY の操作方法

ノブとボタンを知ることはたいへん価値があります！

ここまでで、EYESY を接続して実行する方法し、デバイスの使用方法、システムを支える概念について説明しました。

この章では、ハードウェアコントローラーを実際に操作し、背面パネルのポートに目を向けて、ユーザーの視点から EYESY について説明します。これにより、EYESY の構造内の各コントロールの役割と機能、一般的な操作についての理解が深まります。

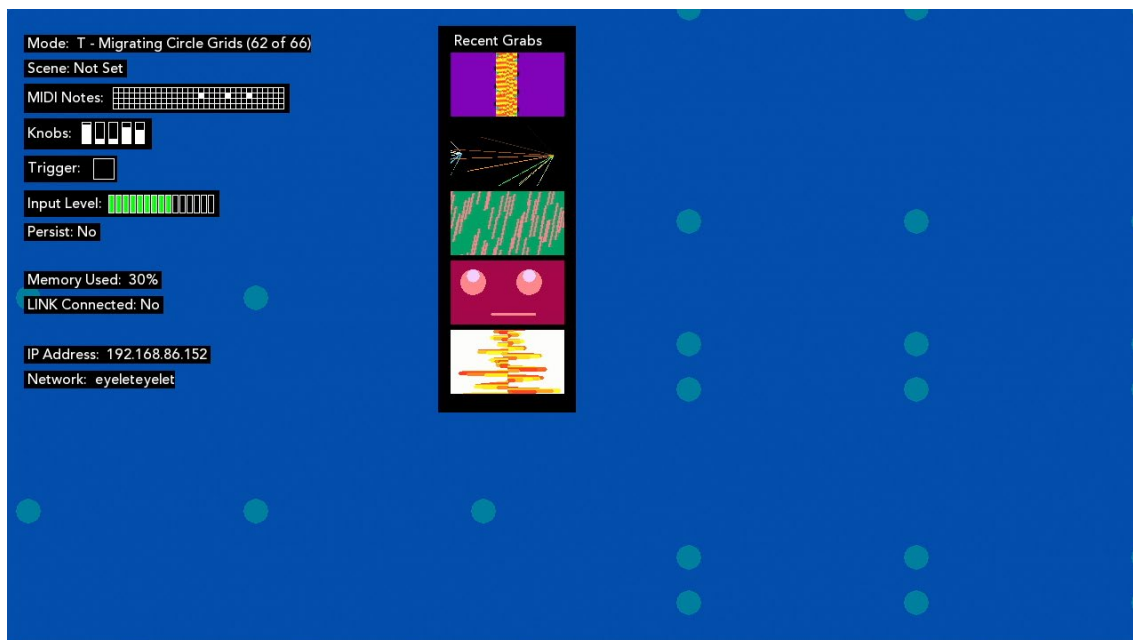
それでは、ノブを回してボタンを押して、実際に何が起こるか見てみましょう！ ...

2.1 システムコントロール

EYESY の一番上の行には、システムレベルのコントロールがあります。これらの項目について、左から右に説明します。



オンスクリーン・ディスプレイ・ボタン(OSD)は、現在のシステム情報のオーバーレイを呼び出すトグルです。以下の画像は OSD の例です。



情報は左側の 2 列に表示されます

- ・受信 MIDI ノートのインジケータ。
- ・ノブの現在の位置。
- ・スレッシュホールドトリガーが現在作動しているかどうかのインジケータ。
- ・着信オーディオ信号用のレベルメータ。
- ・持続トグルまたはバックグラウンドクリア機能が現在有効になっているかどうかのインジケータ。
- ・EYESY のメモリ (RAM) の現在使用されている量。
- ・EYESY がリスンしている着信 MIDI チャンネル。
- ・USB-WiFi アダプターが使用されている場合の、EYESY の現在の IP アドレス。(詳細については、第 4 章を参照してください。)
- ・EYESY が現在接続しているネットワークの名前。

- 現在のモード.
- 最後にロードされたシーン (該当する場合).
- 受信 MIDI ノートのインジケーター.
- 現在のノブ位置情報.
- トリガーが作動するスレッシュホールドを示すインジケーター
- 入力オーディオ信号のレベルメーター
- パーシストグルまたはバックグラウンドクリアー機能の有効/無効を表示
- EYESY の現在のメモリ使用量を表示
- EYESY に入力されている MIDI チャンネル
- USB-WiFi アダプターが使用されている場合の EYESY の現在の IP アドレス。(詳細はチャプター4をご覧ください.)
- EYESY が現在接続されているネットワーク名

右側のコラムには以下が表示されます:

- 最近にスクリーンGrab (キャプチャー)されたサムネイル

オンスクリーン・ディスプレイ・ボタンをもう一度押すと、OSD がオフに切り替わります。

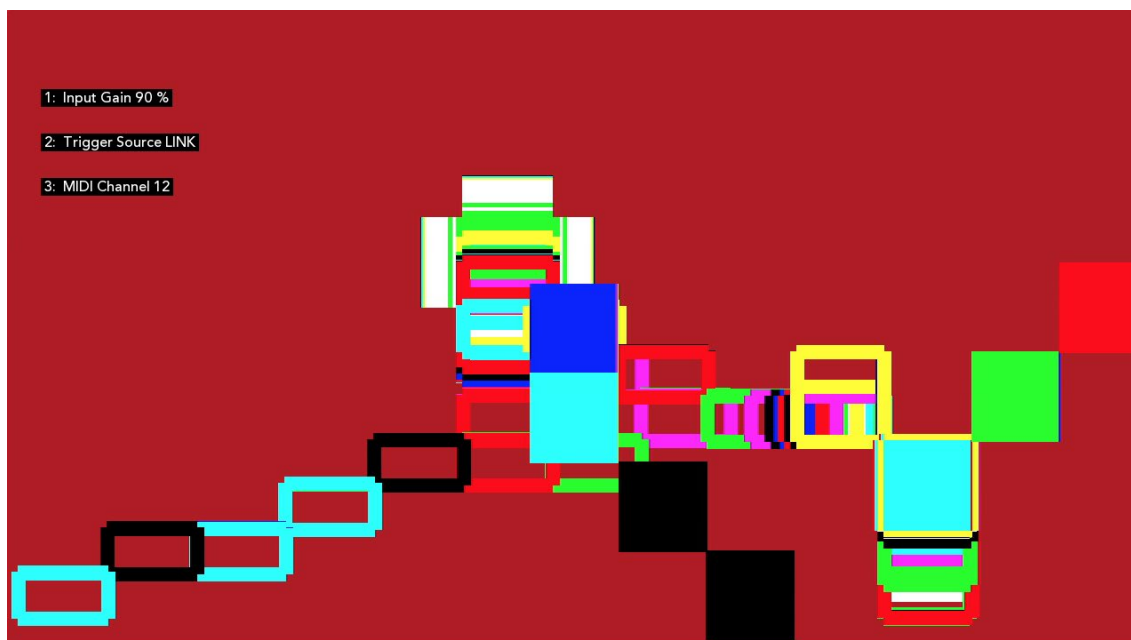
ステータス LED は、EYESY ハードウェアの状態を反映し、アクションの実行、メッセージタイプの受信を示すために、さまざまな色で点灯するライトです。

ステータス LED の点灯は、次のハードウェア状態を示しています。

Dim: LED が点灯していない場合、EYESY は電源を入れていないか、デバイスのシャットダウン状態です。

- 緑/水色で点滅: EYESY は現在起動中です。
- 白色: EYESY は通常動作中です。
- 黄色: スクリーンショット (Screenshot ボタンを押した)
- 緑色: LED が緑色に点滅すると、EYESY が MIDI メッセージ (ノートやコンティニュアスコントローラーメッセージなど)を受信したことを示します。ライトが緑色に点灯したままの場合、MIDI メッセージ (クロックメッセージの可能性が高い) の安定したストリームが EYESY に入力されています。

Shift ボタンを使用すると、他のシステムレベルの設定にアクセスできます。Shift キーを押しながらノブを回すと、さらに多くのオプションを選択できます。Shift を離すと、現在の選択が保存されます。Shift メニュー画面は以下の通りです。



利用可能な設定は次のとおりです:

- Knob1: *Audio Input Gain* – 入力オーディオのゲインを調整します。これは、オシロスコープモードを使用する場合に、適切なレベルに設定したい場合に役立ちます。トリガーモードでトリガーを確認する場合にも利用できます。内部のビデオシステムに影響を与えたり、オーディオを変更したりすることはありませんが、おそらくEYESYにとって最も重要なコントロールです。このノブは、現在のビジュアルモードで使用するためにEYESYによって処理される直前のオーディオレベルを調整します。このノブの目的は、オーディオ信号の強度を調整し、ビジュアル化のための適切なレンジにスケーリングすることです。

このノブの適切な設定は、接続している信号のタイプや、使用しているモードによって異なります。例えば:

- ギターやマイクをEYESYのオーディオ入力ポートに直接接続している場合は、この比較的弱い信号に対して、より高いゲイン設定が必要になります。

- シンセサイザーなどの出力、またはミキサーからのライン出力を接続していますか？ その場合、オーディオ入力ゲイン設定はおそらく少し低い設定になると思います。
- 私たちが想像できないことをしていますか？ その場合は使用しているモードに適切な設定が見つかるまで、ゲイン設定を調整してください！
- Knob2: *Trig Select* – モードがトリガーを取得する「場所」を設定できます。オプションは次のとおりです
 - Audio Input –オーディオのオーディオ入力ポートで、設定されたスレッショルド値を超えるとトリガーが発生します。
 - Ableton LINK Quarter Note – LINK セッションのテンポの各4分音符でトリガーが設定されます。EYESY は、既存のリンクセッションを使用してネットワーク(第4章を参照)に接続する必要があります。OSD は LINK Connected:Yes と表示されます。
 - Ableton LINK Eighth Note – 上記に同じですが8分音符でトリガーが設定されます。
 - MIDI Clock Quarter Note – TRS-MIDI ポート、または USB-MIDI アダプターから入力される MIDI クロックから4分音符をトリガーします。
 - MIDI Clock Eighth Note – 上記に同じですが8分音符でトリガーが設定されます。
 - MIDI Notes – ノートは、TRS-MIDI ポートまたは USB-MIDI アダプターから入力できます。EYESY をノートが送信されているのと同じ MIDI チャンネルで受信するように設定する必要があります。
- Knob3: *MIDI Channel Select* – 番号が1~16の場合、EYESY は指定された MIDI チャンネルから MIDI メッセージを受信し、他のすべてのチャンネルは無視されます。
- Knob4: Shift パラメータへのアサインはありません。
- Knob5: Shift パラメータへのアサインはありません。

それぞれのノブへの入力 Continuous Controller(CC)MIDI メッセージは、Shift パラメータには影響を与えません。(MIDIの詳細については第3章を参照してください。)

Persist Toggle button(パーシスト・トグル・ボタン)は、あらゆる状況で利用できるトグルエフェクトです。

各 EYESY モードには、ビデオの各フレームでトリガーされる 1 つの描画機能(draw function)が含まれています。EYESY のデフォルトの動作は、描画機能が実行される直前に画面をクリアすることです。(このクリアにより、画面をカバーするバックグラウンドコントロールを備えたモードによって色が設定されます)。しかし、このルールを破ると興味深い結果になる場合があります。

「パーシストトグル」により、このクリア機能を無効にすることができます。パーシスト(永続化)が切り替えられた場合、ビデオの新しいフレームごとに、前のフレームの上に描画されます。これにより、新しいタイプの画像が作成される可能性があります。これは、モードを変更する際に特に楽しいエフェクトになります。パーシストをもう一度押すと、オフに切り替わり、通常のクリア機能が復元されます。

2.2 モード・パラメーターのコントロール



コントロールの中央列には、5つのノブがあります。ノブの役割は選択されているモードによって異なります。ノブは、エレメントのサイズ、位置、量、色、変化率などのモードパラメータを制御できます。ノブは、モードの1つのパラメータを制御したり、マク

ロでラップした複数のパラメータにすることもできます。「ファクトリー」モードには5つのコントロールがあります。モードとシーンの違いについては第1章で説明しましたが、モードとシーンの間を移動するときの Foreground (フォアグラウンド) ノブの動作についてはここで追記が必要です。

異なるモードをロードした場合* 新しくロードされたモードの、ノブの現在の値が維持されます。*つまり、ノブ1を1~3時に回し、ノブ3を10時に設定した場合、EYESYは、ロードする次のモードにまったく同じ設定を提供します。

異なるモード間でノブが一貫して割り当てられている場合、これによりモード間のトランジションがより安定します。たとえば、ノブ4をカラー選択パラメーターとして使用する複数のモードをロードした場合、連続するモード間で同じ色の値を設定すると、モードが切り替わる際に同様に解釈されます。また、モードがまったく異なる方法でノブを割り当てている場合でも、別のモードに切り替えてから元の選択に戻ったときに、EYESYの動作は少なくとも設定を保持します。

異なるシーンをロードした場合*、フォアグラウンドノブの値は無視され、シーンに保存されている設定が優先されます。*つまり、シーンをロードすると、シーンに保存されているノブの値がEYESYにロードされます。つまりソフトウェアが保存していたメモリ位置と、実際に目で見えるノブ位置が示す値は異なります。

第1章で説明したように、各シーンは、シーンの作成時のEYESYのモード状態のスナップショットです。この状態には、使用されているモード、各ノブの値、およびパーシスト・トグル・ボタンの状態が含まれます。ノブを回してモードを切り替えると「オーガニック」体験が得られますが、シーンのロードは比較的突然の行為であり、別の時間に別の場所に保存された設定に直接ジャンプします。これは仕様によるものです。

ノブを回す事は常に絶対的なジェスチャーです。ノブの値の変化が検出されるとすぐに、EYESYのビデオエンジンに適用されます。これは、シーンがロードされた後でも当てはまります。ノブを回すとシーンに保存されている値が置き換えられ、物理的なコントロールとソフトウェア間の接続が復元されます。繰り返しになりますが、これは最初は突然のジェスチャーである可能性があります。保存されている値と現在のノブの位置が制御範囲の反対側にあるような場合は特にそうです。ただし、ノブを回すと、連続したスムーズな値のトランジションが作成されます。

2.3 ファンクション・コントロール

EYESY の一番下の行には、ナビゲーションとアクションに関するオプションがいくつかあり、すべてメープルの木製ボタンになっています。それぞれを左から右に見ていきましょう！



一番左側のペアのボタン(眼球アイコン)は【モード・セクター・ボタン】です。EYESY の microSD カードの Modes フォルダーに現在保存されているモードを後方(左)または前方(右)に切り替えられます。

次は【シーン・セクター・ボタン】(カチンコのアイコン)です。このボタンのペアを使用すると、保存されているすべてのシーンを後方(左)または前方(右)に切り替えられます。

【シーン・セーブ・ボタン】(フロッピーディスクのアイコン)は、選択されているモード、5つのノブの値、および[パーシストグル]ボタンの状態など、EYESY のパフォーマンス値を新しいシーンに保存します。この新しいシーンは、シーンリストの最後に配置されます。

【スクリーン・ショット・ボタン】(カメラのアイコン)を押すと、現在のビデオモードの出力を、EYESY の Grabs フォルダ内の静止画像ファイルにキャプチャします。オーバーレイがビデオディスプレイに表示されている場合でも、スクリーンショットにはオンスクリーンディスプレイは含まれません。

出力ファイルにはゼロ(0)から番号が付けられ、JPEG ファイル(たとえば、0.jpg、1.jpg など)として保存され、EYESY のビデオ出力の仕様 1280 × 720 ピクセル、72PPI の解像度に準じます。

これらのファイルにアクセスするには、EYESY エディターを使用してください。これについては、第 4 章で詳しく説明します。

【トリガー・ボタン】(星で囲まれたビックリマークのアイコン)はパネルの一番右下にあります。このボタンを押すと、トリガーイベントが生成されます。このトリガーイベントは、外部ソースからのトリガーイベントと同等です (Shift + Trigger Select ノブを使用して選択されたものの 1 つ、第 2 章を参照)。これは、トリガーモードで最も一般的に使用されます (第 1 章で説明)。

トリガーボタンを押すと、受信したサンプルの EYESY のバッファを正弦波で満たします。上記のトリガー動作とともに、これら 2 つの機能により、EYESY にオーディオが入力されていない場合でも、トリガーボタンが便利な代替手段になります。多くのモードはこのトリガーメッセージを利用して入力オーディオ信号を補足することもできるので、あらゆる種類のモードで試してみてください。

リアパネル



ほとんどの端子(ポート)は EYESY の背面にあります。左から右に紹介していきます。

パワースイッチです。スイッチをオフの位置すると、EYESY の OS がシャットダウン作業に入ります。EYESY は PC と同じように取り扱う必要があります(EYESY は LinuxOS のコンピューター)。完全に OS がシャットダウンして、LED が消えるまでは電源アダプターを取り外さないでください。途中で電源を外してしまうとコンピューターの起動中に電源コードを抜くようなものです！

9VDC 電源ポート EYESY に付属の電源アダプターを使用してください。電源の出力仕様は次のとおりです。9VDC、1000mA、極性はセンタープラス(TIP=+)。電源が EYESY で動作するかどうか不明な場合は、プラグを差し込まないでください！

コンポジット(RCA)出力は、EYESY のコンポジット方式のビデオ出力として機能します。電源投入時に、EYESY は最初に HDMI ディスプレイの接続を優先的に使用します。見つからない場合は、ビデオを出力するためにこのポートを自動的に選択します。このポートは、NTSC または PAL 形式の信号を送信できます。フォーマットを選択するには、第 4 章の最後のセクションを参照してください。

HDMI 出力ポートは、EYESY のビデオ出力として機能します。電源投入時に、EYESY は電源付き HDMI ディスプレイに接続されていることを最初に確認します。

HDMI ディスプレイが接続され、電源が入っていて利用可能な場合には、このポートを自動的に選択します。

Note: HDMI 出力ポートはビデオのみを送信します。EYESY からの音声は送信されません。

Note: コンポジット出力ポートと HDMI 出力ポートの両方を同時に使用することはできません。

マイクロ SD カード・スロット: ラベルのない microSD カードスロットがあります。このスロットには、内部マイクロコンピュータのオペレーティングシステムとモードストレージの場所として機能するプリロードされたカードが収納されています。このカードの内容はコンピュータで読み取ることができないため、新しいディスクイメージをカードに「書き込む」場合を除いて、このカードを取り出したり操作したりしないでください! WiFi 経由でカードのストレージにアクセスする方法については第 4 章を、OS の書き込みについては第 6 章を参照してください(「ファクトリーリセット」の実行についての項を含む)。

MIDI 入力ポート: 外部 MIDI コントローラーを接続できます。このポートは TRS-MIDI(TYPE A)です。TYPE B ケーブルは使用できません。EYESY の MIDI に関する情報は、第 3 章にあります

オーディオ入力ポート: 1/4"フォーン端子の入力です

サイドパネル(左側)

USB ポートは左側のサイドパネルにあります。このポートは、USB-WiFi アダプターまたはクラス準拠の USB-MIDI アダプターを接続するためのものです。



さらに多くの USB ポートが必要な場合は、USB ハブをこのポートに接続する事もできます。ハブや接続するその他のデバイスはクラスに準拠している必要があります。特別なソフトウェアドライバーをインストールする必要はありません。USB ポートを使用して、クラス準拠の MIDI コントローラーなどの他のデバイスに電力を供給する場合、EYESY は最大 500 ミリアンペア (0.5A) しか供給できないことに注意してください。より多くのアンペア数が必要な場合は、電源付きの USB ハブを使用してください。

An Epilogue on Basic Controls

EYESY が提供するコントロールは一般的に似ていますが、各モードによって異なります。EYESY は予め用意されたモードに加え、ユーザーが開発したモードなどを含め様々なモードがあり、それぞれにコントロールが異なります。

3. EYESY の MIDI 構成について

EYESY の MIDI インプリメンテーションチャートについて説明します。USB-MIDI を含む外部 MIDI コントローラーを EYESY に接続する場合、または EYESY を外部シーケンサーとペアリングする場合にも役立ちます。

外部 MIDI コントローラーを使用して、EYESY で MIDI を扱うには、専用の MIDI 入力ポートを使用する方法と、USB ポートを使用する方法があります。

先に進む前に、両ポートのいくつかの重要な特性について説明する必要があります。

MIDI 入力ポートは TRS-MIDI【タイプ A】ポートです。タイプ A は、3.5mm(1/8") TRS プラグで接続します。標準の 5 ピン DIN コネクタを備えた外部デバイスに接続するには、タイプ A オス 3.5mm(1/8") TRS～メス 5 ピン DIN 変換アダプターが別途必要です(タイプ A は、MIDI Manufacturers Association によって標準規格として採用されています)。タイプ B は EYESY では機能しないので注意してください。タイプ A と B は外見上は同じことが多いため注意が必要です。MIDI を使用する場合には必ず正しいアダプターのみを使用してください。

現在、多くの楽器やコントローラーには TRS-MIDI 出力が採用されていますので、この場合には、3.5mm(1/8") のオス-オス TRS ケーブルのみで EYESY と接続できます。しかし必ず EYESY に接続するデバイスがタイプ A 接続があることを確認する必要があります。そうでない場合は、タイプ A～ B に変換するアダプターなどが必要になると思います。

USB-MIDI デバイスは EYESY の USB ポートからも使用できます。互換性を確保するために、クラス準拠の USB-MIDI コントローラーまたはアダプターを選択することをお勧めします。ドライバーのインストールが必要な USB-MIDI デバイスは、おそらく期待どおりに機能しない可能性があります。EYESY は Linux コンピューターであり、多くのメーカーが Windows および MacOS 専用の USB-MIDI デバイスを開発していることを知っておくことが重要です。Linux / UNIX OS と互換性があるかどうか、USB-MIDI デバイスの製造元に確認してください。そうでない場合は、USB ポートではなく EYESY の TRS-MIDI ポートを使用してデバイスを使用することをお勧めします。

EYESY と適切に対話するには、このデバイスの MIDI インプリメンテーションを知る必要があります。EYESY は、指定されたチャンネルにある着信 MIDI メッセージのみを扱います(デフォルトでは MIDI チャンネル=1、Shift + MIDI チャンネル選択ノブで変更

できます)。また、MIDI メッセージの一部だけが、EYESY の動作モードに実際に影響を及ぼします。

つまり、コントローラーは EYESY が認識しないメッセージを送信する可能性があります。それは問題ありません。これらのメッセージは EYESY には影響を与えませんが、問題も発生しないため、「アクティブセンシングを無効にする」必要はありません。

EYESY が使用するさまざまなタイプのメッセージを調べることから始めましょう。

Control Change Messages

コントローラー番号 21～25 を使用したコントロール・チェンジ・メッセージは、対応するノブにマップされます。:

- Controller Number 21 > ノブ 1
- Controller Number 22 > ノブ 2
- Controller Number 23 > ノブ 3
- Controller Number 24 > ノブ 4
- Controller Number 25 > ノブ 5

各着信メッセージの値は、それぞれのノブの値を置き換えます。したがって、ノブ 2 を右端まで回しても、値が 0 の番号 22 の着信コントローラーメッセージは、コントローラー 2 の値をゼロに変更します。ノブ 2 をもう一度動かすと、その制御がリストアされます。

Note Messages

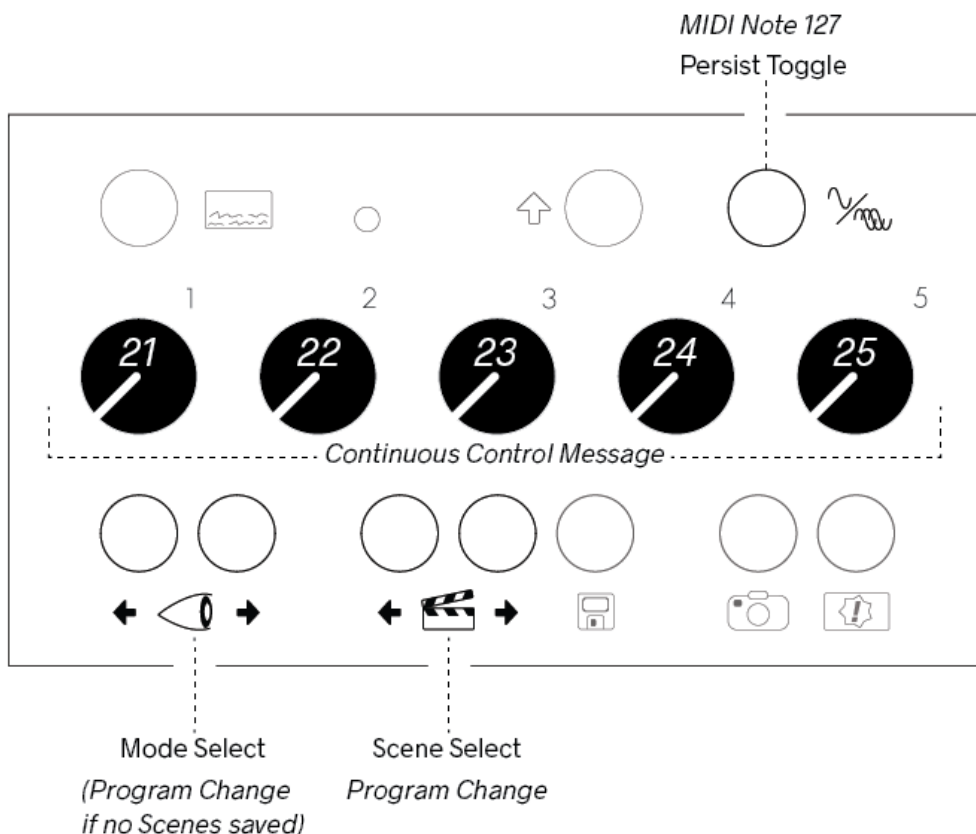
すべての着信ノートメッセージ(グローバル MIDI チャンネル上)は、EYESY で受信されます。これらのノートメッセージに関する情報は 2 つの方法で提供されます。

- EYESY は、すべての着信ノートメッセージをトラックし、常にピッチがオンまたはオフになっているタブを保持します。
- EYESY は、前のビデオフレーム以降の MIDI ノートを受信するたびに、通知を提供します。これは、モードで「ノートトリガー」効果を作成する場合に役立ちます。

どのモードでも、これらのステータスデータのいずれか、または両方を使用できます。

さらに、最後の MIDI ノート(ピッチ 127)は、自動クリア機能(auto clear function)の状態を切り替えるために使用され、パーシスト・トグル・ボタンを押すのと同様に機能します。したがって、ピッチ 127 の「Note On」メッセージは現在の状態を切り替えます(もしオンだった場合はオフになり、その逆も同様です)。

以下は MIDI とそれぞれのコマンドでオーバーライドできるハードウェア図です。



MIDI Clock Messages

EYESY では、いくつかの MIDI システムのリアルタイムメッセージも利用できます。多くのシーケンサーは MIDI クロックをサポートしていますが、他の環境 (Pure Data [Organelle]、Max、Reaktor など) にも組み込むことができます。

システム・リアルタイム・メッセージに関する MIDI の概念では、タイミング・クロック・メッセージは四分音符ごとに 24 回受信され同期を確立します。したがって、EYESY がタイミング・クロック・メッセージを受信するたびに、内部カウンタが 1 ステップ増加します。24 番目のステップの後、カウンタが戻り、再びカウントアップします。さらに、

受信されたリアルタイムのスタートメッセージがトリガーされると、次のクロックビートでカウンターがリセットされます。

四分音符あたり 24 ティックのクロックは、トリガーソースとして使用できます。Shift + Trigger Select ノブを使用すると、四分音符または八分音符 (24 または 12 ティックごと) にトリガー発信するイベントを選択できます。

MIDI クロックメッセージはシステム MIDI メッセージであることに注意してください。つまり、これらのメッセージは、EYESY が現在選択している MIDI チャンネルに関係なく受信されます。

Program Change Messages

着信するプログラム・チェンジ・メッセージは「シーン」を選択します。シーンが保存されていない場合、プログラム変更メッセージは「モード」を選択します。

Changing the Selected MIDI Channel

EYESY が MIDI メッセージを受信するチャンネルを変更するには、第 2 章の「Shift」メニューのセクションを参照してください。

4. EYESY へのアクセス

WiFi, WiFi, WiFi.

この章では、EYESY が外のデバイスと相互作用する方法に焦点を当てます。EYESY は、ファクトリーモードとオンボードコントロールで十分に楽しめますが、更に新しいモードや、モディファイされたモードを追加して拡張することができます！

この章では、付属の USB-WiFi アダプターでできることについて説明します。

- WiFi「アクセスポイント」を作成するか、既存の WiFi ネットワークに参加できません
- ファイル管理: モードの追加と削除、画像のアップロード、スクリーングラブのダウンロードなどを行うことができます。
- プログラムモード: 新しいモードを作成し、既存のモードをカスタマイズ可能です！

上記のほとんどは、EYESY のエディターページとコンピューター間で行われます。

まず、WiFi で EYESY に接続し、エディターの各部分について説明し、次に AbletonLink について少し説明します。

注意: 付属の USB-WiFi アダプターは Ralink5370 チップセットを使用しています。このアダプターは、2.4 GHz WiFi ネットワークにのみ接続します。私たちは他の WiFi アダプターを試しましたが機能しないものもありました。他の使用可能なアダプタについては、[forum](#) を参照してください。基本的には機能することが証明されている付属のアダプターを使用してください。

4.1 アクセスポイント vs 既存ネットワークへの参加

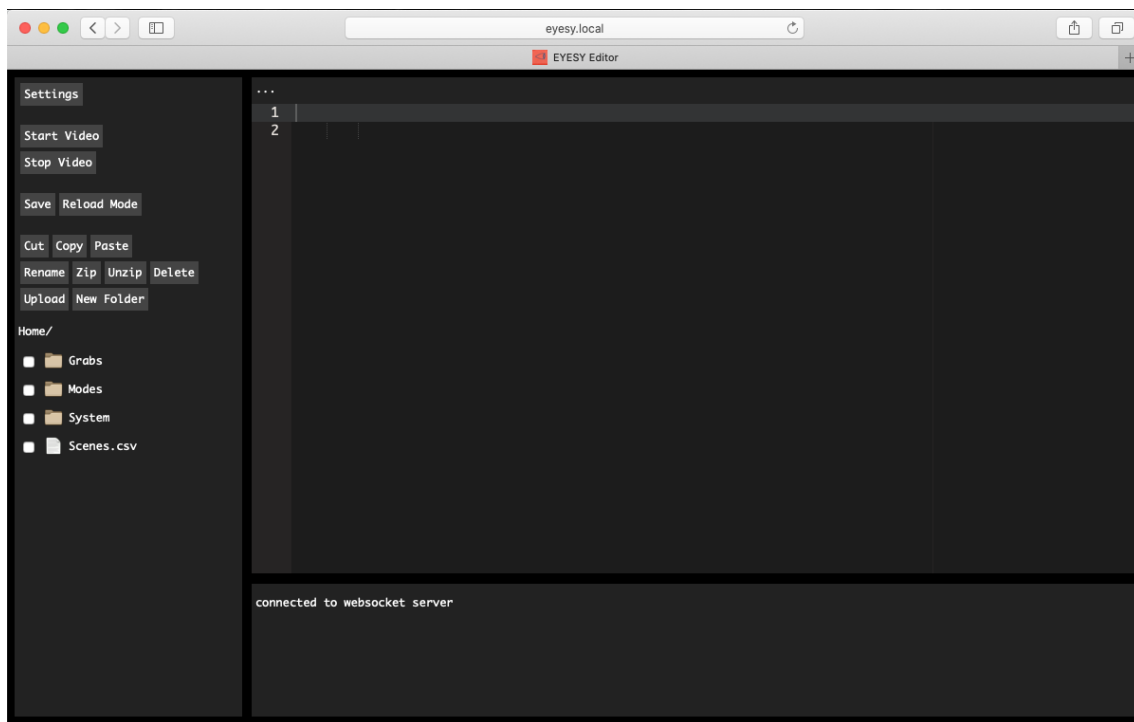
EYESY には、アクセスポイント (AP) モードと、通常モードの 2 つの WiFi 操作モードがあります。AP モードで EYESY は USB-WiFi アダプターを使用して独自のネットワーク (アクセスポイント、またはホットスポット) を作成します。通常モードでは、EYESY は USB-WiFi アダプターを使用して既存の WiFi ネットワークに接続します。ファイルの管理とプログラミングは、WiFi モードに関係なく同じ方法で行われます。

1つの注意点は、初めてWiFiを使用するときは、アクセスポイントモードを使用する必要があるということです。これにより、コンピューターをEYESYに直接接続できません(別のワイヤレスルーターは必要ありません)。コンピューターがAP経由で接続されたら、EYESYのエディターページにアクセスして既存のWiFiネットワークの資格情報を入力し、EYESYが参加できるようにします。(これにより、まったく同じネットワークのパスワードを入力する目的で既存のネットワークに参加しようとする問題が解消されます！)

Getting Connected

アクセスポイントモード経由でエディターページにアクセスするための必要な手順を見ていきましょう。:

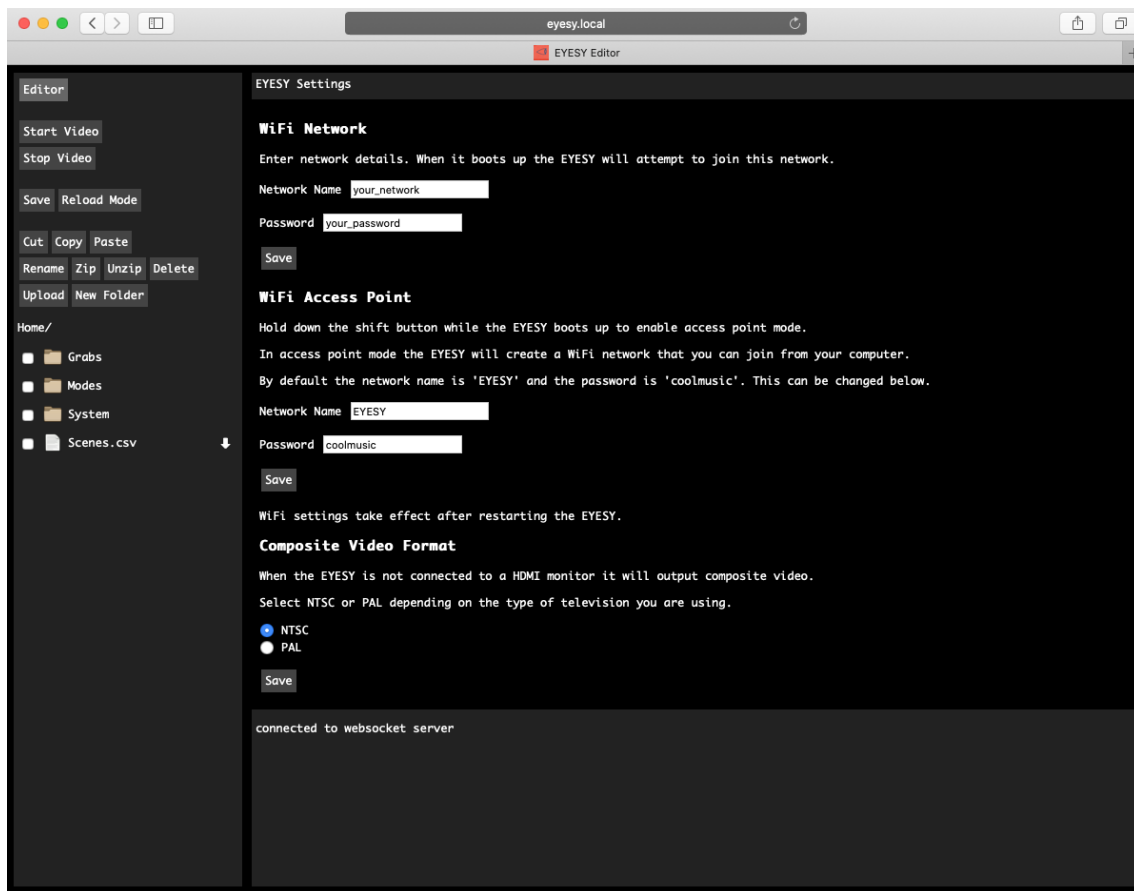
1. EYESY をオフにして、USB-WiFi アダプターを EYESY の USB ポートに接続します
2. Shift キーを押しながら、EYESY をオンにします。これにより、アクセスポイントが開始されます。ディスプレイに「EYESY」と表示されるまで Shift キーを押し続ける必要があります。
3. コンピューターの WiFi ネットワーク設定で、EYESY ネットワークを選択し、パスワードフィールドに「coolmusic」と入力します。
4. コンピューターのブラウザのアドレスバーに、**eyesy.local** または IP アドレスを入力します。IP アドレスは OSD にあります。AP が使用されている場合、IP アドレスは常に **http://192.168.12.1** になります
5. 以下のようなエディターが表示されます。



エディターで出来る事はたくさんありますが、ここでは、既存のネットワークのクレデンシャルの入力と使用に焦点を当てます。:

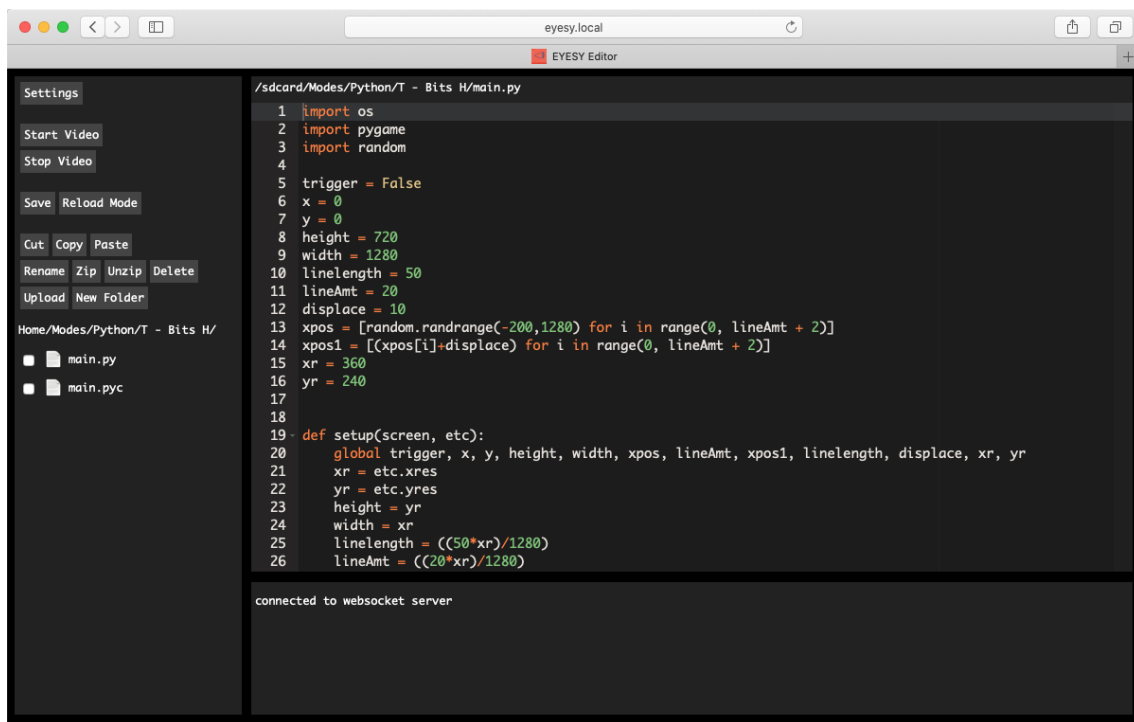
1. 左上隅の[Settings]をクリックします
2. 参加するネットワークの詳細を入力し、[save]を選択します(下のスクリーンショットを参照)
3. EYESY の電源スイッチを OFF 側にスライドします。LED が消えたら、電源スイッチをスライドさせて再度 ON に戻します(Shift キーを押したままにしないでください)
4. ネットワーク・クレデンシャルを正しく入力した場合、EYESY は起動後に既存のネットワークに自動的に参加します。OSD を表示すると、EYESY が接続されているネットワークの名前と IP アドレスが表示されます。IP アドレスは接続セッションごとに変わる可能性があるため、次の手順に進む前に OSD を確認することをお勧めします。
5. コンピューターのブラウザのアドレスバーに、**eyesys.local** または IP アドレスを入力します。

既存のネットワークの詳細はいつでも変更できます。EYESY が生成する AP の名前とパスワードを変更することもできます。WiFi 設定の変更は、次に EYESY が起動するまで有効になりません。以下のスクリーンショットは、ネットワークフィールドと AP フィールドを示しています。:



4.2 エディターの説明

エディターはEYESYをリアルタイムで変更できます。コンピューターのFinder / ファイルエクスプローラーと同じように、ファイルやフォルダーを移動、追加、削除、およびコピーできます。特定のモードのコードを表示、編集、およびリロードすることもできます。プログラミングを行うために、エディターにはコンソールがあり、モードをデバッグできます。これらの機能が3つのセクションに分かれています。



左側には、ビデオ出力とファイルブラウザおよび管理をコントロールするためのボタンが付いたセクションがあります。ファイルブラウザをクリックして、たとえば Home / Modes / Python / T-Bits H / main.py に移動すると、コードのフィールドに main.py のコードが表示されます。このコードは編集可能です！コードの下には、EYESY の端末出力を表示し、コードで何が起きているかを示すウィンドウとして役立つコンソールフィールドがあります。コード内の print コマンドは、変数の値などをここに出力します。

The Control Pane

上から順に、コマンドは次のとおりです。:

- *Settings*: 上記のように、EYESY のネットワーク設定ページに移動します。
- *Start Video*: このボタンは、ビデオ「エンジン」を再起動させます。起動時にはビデオ出力がすでに開始されているため、現在実行中のビデオでもう一度これを押すと、ビデオ出力が停止したのち再起動します。ディスプレイに再び起動画面が表示され、Modes フォルダーに最初のモードが表示されます。コンソールには、起動中のビデオエンジンの出力が表示されます。
- *Stop Video*: このボタンはビデオの「エンジン」を停止し、ディスプレイは黒い画面になります。コンソールに「Exiting」と表示されます。このボタンは、プログ

ラミングエラーのためにモードの実行を停止する必要がある場合に役立ちます。

- *Save*: 現在編集モードを保存します。
- *Reload Mode*: 現在ディスプレイに表示されているモードをリロードします。プログラミング中は、必ず木製のモードセレクトボタンを使用して、編集しているのと同じモードを表示する必要があります。そうしないと、表示していないモードに変更を加えることになります。
- *Cut, Copy, Paste, Rename, Zip, Unzip, Delete, Upload, New Folder*: (切り取り、コピー、貼り付け、名前の変更、Zip、解凍、削除、アップロード、新しいフォルダーの作成): 一般的なファイル管理コマンド。
 - ファイルブラウザの各項目の左側に白いボックスがあります。コマンドを選択する前に、ボックスを「チェック」する必要があります。(これは、貼り付け、新しいフォルダー、またはアップロードには適用されません。)
 - フォルダを EYESY との間でコピーするには、最初にフォルダを圧縮する必要があります。フォルダは圧縮せずにアップロードまたはダウンロードすることはできません。
- *File Browser*: 起動時に、Home / が、3 つのフォルダと「Scenes.csv」というファイルの上に表示されます。ホームフォルダには、これらのフォルダとファイルが含まれています。フォルダ(例: Home / Modes /)をクリックするか、ファイルをクリックしてコードのフィールドに表示することができます。ファイルブラウザとファイル管理の詳細については、以下をご覧ください。

4.3 マイクロ SD カードのフォルダ/ファイル管理

すでに述べたように、EYESY を実行するには microSD カードが必要です。EYESY に同梱されているカードには、OS 用とストレージ用の 2 つのパーティションがあります。OS パーティションはエディターからアクセスすることはできません。ストレージパーティションのホームフォルダには、使用したいすべてのものがあります。ホームにアクセスする唯一の方法は WiFi 経由です。モードは保存され、ホームから直接実行されます。(シーンとスクリーンショットもここに保存されます)。

- ホームフォルダには、*Modes* という名称のフォルダが必ず含まれている必要があります。(大文字と小文字も区別されます)。ロードするすべてのモードは、以下に詳述するように *Modes* 内に存在していません。

- *Home*には、撮影したスクリーンショットを格納するための **Grabs** というフォルダーと、保存されているすべてのシーンを記録するための **Scenes.csv** というファイルが含まれます。これらのフォルダが存在しない場合、これらのアイテムは必要に応じて自動的に作成されます

以下は、ホームのディレクトリ・リストの例です。ここでは、4つのモードを確認できます。各モードには、必要な main.py ファイルと、Grabs フォルダおよび Scenes.csv ファイルがあります。

Grabs/

- 0. jpg
- 1. jpg
- 2. jpg

Modes/Python

- S – Circle Scope Connected/
 - main. py
- S – Feynman/
 - main. py
 - main. pyc
- T – Density Cloud/
 - main. py
 - main. pyc
- T – Spanish Flash Cards/
 - font. ttf
 - main. py
 - main. pyc
 - spanish. py

System/

- ap. txt
- wpa_suppl icant. conf

Scenes. csv

モードについての注意事項:

- 上記の4つのモードフォルダでは、すべてに必要な main.py ファイルがあります。これらの Python スクリプトファイルに EYESY モードの最小要件が含まれている限り、それらはすべて有効なモードとして扱われ、デバイスの起動時

に読み込まれます。(プログラミングモードまたは編集モードについては、第 4 章を参照してください。)

- これらのモードフォルダには、main.pyc ファイルも含まれている場合があります。これらの仲介ファイルは、EYESY の Python コンパイラによって自動的に作成されます。これらのファイルはスクリプトを高速化できますが、PYC ファイルは必須ではありません。(ここにリストされている最初のモードには main.pyc ファイルがなく、EYESY によってまだ実行されていない新しいモードであることを示しています。)
- モードに必要なその他のファイルも、モードフォルダに含める必要があります。これには、画像、追加の Python スクリプト、フォントファイル、またはその他が含まれる可能性があります。
- モードフォルダにはサブフォルダを含めることができます。サブフォルダは、含まれているモードで認識できます。

Modes フォルダ内で、4 つのモードが選択され、その内容が使用されます。したがって、上記の例では、EYESY で使用できる 4 つのモードは、S-Circle Scope Connected、S-Feynman、T-Density Cloud、および T-Spanish FlashCards です。また、ファクトリモードでは、先頭の S-はスコープモードを示し、先頭の T-はトリガーモードを示しています(第 1 章を参照)。

また、トップレベルには Scenes.csv ファイルがあります。このカンマ区切り値のファイルは、コードペインまたはコンピューターのテキストエディターで開くと読み取り可能です。各行は次のようになります。

```
bouncing-  
ball, 0. 0, 0. 2668621700879765, 0. 5141739980449658, 0. 7526881720430108, 1. 0,  
False
```

These values represent:

Mode, Knob 1, Knob 2, Knob 3, Knob 4, Knob 5, Persist Toggle button

もしあなたに興味(と勇気)がある場合は、これらのシーンの値を編集したり、行を並べ替えたりできます。ただし、万が一の場合に備えて、最初にファイルのローカルコピーを作成することを強くお勧めします。(手に負えなくなった場合は、いつでも Scenes.csv を削除できます。新しいファイルが必要な場合は、EYESY が作成します。)

モードを操作するときは、各フォルダー内のファイル进行处理するのではなく、フォルダーを操作することをお勧めします。例えば:

- 外部ソースから EYESY に新しいモードを追加する場合は、アップロードする前に必ずフォルダー全体を圧縮(zip)してください。アップロードしたら、チェックボックスを選択して[Unzip/解凍]コマンドを使用します。
- モードをコンピューターにバックアップする場合は、フォルダーをコンピューターに zip コピーします。
- モードの名前を変更する場合は、フォルダーの名前を変更します (main.py はそのままにします！)
- モードを複製して新しいコピーに変更を加える場合は、モードフォルダーをコピーして貼り付け、コピーに別の名前を付けます。
- モードを削除する場合は、必ずモードフォルダ全体を削除してください。

最後に、ホームにある他のフォルダまたはファイルは、通常の操作では EYESY によって無視されることに注意してください。これは、「ライブ」バージョンを編集している間、ホームで作業モードのバックアップフォルダを保持できることを意味します。または、これは、モードのさまざまな「セットリスト」やさまざまなシーンファイルを作成する場合にも便利です。セットリストのアイデアは、Modes フォルダー、または Scenes.csv ファイルのいずれかを複製してから、将来使用するために保存するバージョンの名前を変更することです。ステージに上がる前に、適切なファイルとフォルダーに適切な名前が付いていることを確認してください！！

The Code Pane

コードペインはコードエディタであり、行番号と構文の強調表示/色付けを備えています。基本的なテキストエディタと同じように編集/プログラミングが可能です。実際には、編集/プログラミングでは、コードペインを、コントロールペインにあるコマンド、およびコンソールに表示される構文エラーまたはランタイムエラーと組み合わせて使用します。

モードのコードを表示するには、ファイルブラウザを使用して、編集するファイル(モードの main.py ファイルなど)を選択します。変更を加えた後、コントロールペインに *Save*、*Reload* コマンド(および場合によっては *Start Video*、*Stop Video* コマンド)が必要になります。したがって、コードエディタでモードスクリプトを変更した後、*Save* をクリックしてそれらの変更をコミットします。次に、*Reload* をクリックして、ディスプレイの

ビジュアル出力を更新します。構文エラーまたはランタイムエラーがある場合、OSD をオンにしている場合は、コンソールまたはディスプレイに表示されます。

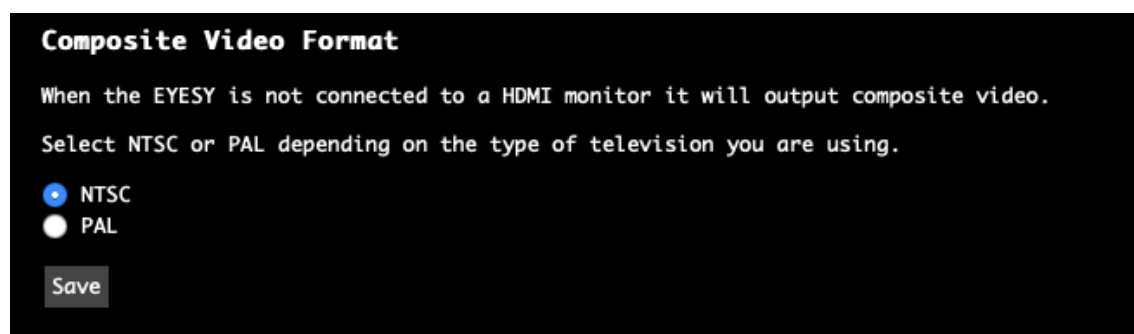
プログラミングの詳細については、次の章を参照してください!

WiFi は素晴らしいですが、USB ドライブも使えるのでしょうか?

EYESY の前身 (ETC) と兄弟 (Organelle) での経験から、ファイルとプログラミングを管理する方法が 1 つしかない方が、多くの方法よりも優れていることがわかりました。そのため、USB ドライブを接続しても、EYESY は USB ドライブからの読み取りや USB ドライブへの書き込みを行いません。すべてのファイル管理とプログラミングは、USB-WiFi アダプターを介して行われます。

Composite Video Formats

[Settings] ページには、WiFi、ファイル管理、プログラミングに関係のない選択肢が 1 つあります。composite video format (コンポジットビデオフォーマット) の設定です。



設定に保存された変更は、再起動後に有効になります。(起動時に、EYESY は最初に電源付き HDMI ディスプレイに接続されているかどうかを確認します。見つからない場合は、コンポジットポートを使用してビデオを出力します。したがって、HDMI ケーブルを EYESY に接続しないでください。コンポジット出力を使用するには、ディスプレイにも正しい入力を設定することを忘れないでください!)

5. EYESY のプログラミング

テキストエディタと夢を持っている人のために！

EYESY モードが作られる場所であるファイル管理について理解したところで、プログラミングを始めましょう。

まず最初に、この章は必要でない方は読み飛ばしても構いません！EYESY はコーディングの世界を知らなくても、オフィシャルやシェアパッチなどを利用するだけでもかなり機能的で表現力豊かです。

この章では、USB-WiFi アダプターを使用してコンピューターのエディターにアクセスすることに慣れていることを前提としています。そうでない場合は、前の章(第 4 章)を参照してください。

この章では、EYESY がどのように機能するか、およびこのフレームワーク内で使用できるメソッドに焦点を当てていますが、コーディング方法については説明しません。

現在はインターネットや書籍などで、プログラミング一般や特定の言語(Python など)について学ぶのに役立つ豊富なリソースが利用可能です。プログラミングを学びたいのであれば入門するのに適した題材はたくさん存在しています。

さて、すでにコーディングを楽しんでいる(または検討している)場合、または表現する必要のあるビデオのアイデアがある場合は、次に進みましょう。そのレバーを引くだけでボンネットが飛び出します。

この章では、EYESY モードのコンセプトと要件から始めます。各モードはそれほど多くを必要としませんが、そこに何が必要かについて説明することで、EYESY がどのように動作するかをよく理解できます。次に、EYESY のアプリケーションプログラムインターフェイス(API)の概要を説明します。これらはどんなモードでも利用可能な機能と特性です。

*** 弊社ではプログラミング言語(Python、Pygame など)に関するサポートは行っておりません。**これらは汎用のプログラミング言語であり、これらを学ぶには専門的な知識を書籍やインターネットなどを通じて学ぶ必要があります。EYESY を使用して自分でプログラミングしたモードを作成したい場合には Python や Pygames などに関する知識が必要ですが、すでに用意されている多彩なモードや、ユーザーによるシェアパッ

チを追加して遊ぶだけでも、EYESY は無限の楽しみを秘めています！

5.1 モードスクリプトについて

EYESY が実際にどのように機能するかについて、使用している言語と EYESY のシステム動作について少し説明した後、モードの基本的な要件について説明します

Python, Pygame, and EYESY

EYESY のモードは **Pygame** のグラフィックライブラリを使用して **Python** で記述されています。つまり、Python は使用されているプログラミング言語であり、描画を実行するグラフィックライブラリは Pygame から提供されています。

Python には、多数のモジュールを備えた豊富な標準ライブラリがあり、そのいずれもモードで使用が可能です。モジュールの完全なリストは、[Python のドキュメント](#)を確認してください。ファクトリモードを通過すると、glob、imp、math、random、time inuse などのモジュールが表示されます。ライブラリモジュールがどのように機能するか、または Python 自体について詳しく知るには、上記のドキュメントリンクから始めるのが最適です。Python は標準的なプログラム言語となるため、その言語を学ぶには専門の書籍、またはインターネット上の記事などを参考にしてください。

Pygame は、もともとビデオゲーム開発用の Python モジュールです。EYESY は、Pygame のグラフィックライブラリを利用してビジュアルを作成しています。利用可能なモジュールのリストは、[Pygame のドキュメント](#)を参照してください。一部のファクトリモードでは、pygame.freetype や pygame.gfxdraw などのライブラリを使用しますが、他のグラフィック関連モジュールも利用が可能です。

Pygame は、グラフィック出力をサーフェスにレンダリングします。これが、Python スクリプトを Pygame 出力に接続する方法です。これについては次のセクションで説明します。

EYESY が起動すると、microSD カードの Modes フォルダ内のすべての有効なモードが識別されます。そして、これらすべてのモードを同時にロードします。この並列動作により、モード間の切り替えをスムーズに瞬時に行いますが、いくつかの注意点が 있습니다。:

- **メモリを大量に消費するモードについて:** 大量の RAM を必要とするモードをロードする場合(たとえば、`setup()`中に多数のイメージをロードする場合など)、別のモードが呼び出されて出力を生成する場合でも、このメモリが使用されることに注意が必要です。したがって、一般的に EYESY のパフォーマンスが遅い場合は、メモリを占有するモードが関連している可能性が高いことを覚えておく必要があります。これをトラブルシューティングする 1 つの方法は、テスト中にロードするモードの数を減らして見ることです (Modes フォルダーから怪しいモードを移動してみる)。
- **サポートファイルは常に最小限に抑える:** 画像をロードするモードでは、モードフォルダ内の画像は 10MB 以下に保つことをお勧めします。カメラや携帯電話から直接写真をダウンロードする場合は、写真を加工して、解像度を EYESY の HDMI のデフォルトの出力解像度 (1280 x 720) に近いか、それよりも低くすることを勧めます。

Minimum Mode Requirements

各モードには、グラフィカル出力を生成するための基本的な 4 つの要素があります。

1. **“pygame” モジュールのロード**
Python の他のモジュールと同様に、これは先頭の `import` ステートメントから実行されます。したがって、EYESY モードのスクリプトはすべて…`import pygame…`で始まる必要があります。
2. **“setup” ファンクションを追加する**
これは実際にはオプションですが、強くお勧めします。 `setup()` は、モードがロードされると自動的に呼び出されます。これは、ビデオがレンダリングされる前に 1 回だけ必要な操作を実行するためのクリーンな方法です。
3. **“draw”ファンクションを追加する**
`draw()` は、ビデオのフレームごとに 1 回実行され、ビデオ出力の各フレームをペイントする関数です。ここでの一般的なアクティビティには、最後のフレーム以降のコントロールの変更の取り込み、更新されたオーディオ情報による再描画、受信したトリガーやその他のメッセージに基づく変更の実行などがあります。このブロックのコードはフレームごとに実行する必要のあるアクションを含みます。
4. **出力を“screen”にルーティングする**
EYESY は、`screen` と呼ばれる Pygame サーフェスを作成します。このサーフェスは、HDMI ポートを介して表示するために送信されるものです

A Simple Mode Example

これらすべての点を考慮して、ここに非常に簡単なサンプルの EYESY モードをあげます…

```
import pygame

def setup(screen, etc):
    pass

def draw(screen, etc):
    size = 640
    position = (510, 500)
    color = (255, 0, 0)
    pygame.draw.circle(screen, color, position, size, 0)
```

この基本的な main.py の例を見ていきます。まず、上記のステップ 1 で説明したように、pygame モジュールをロードします。次に、setup()関数を定義します。これは空白ですが、必要に応じて使用するためのプレースホルダーとしてあらかじめ追加しておくといでしょう。

次は draw()関数です。関数の最初の 3 行は、pygame.draw.circle()関数の呼び出しに使用されるローカル変数を定義しています。そして、例の最初の引数が示すように、この関数は画面のサーフェスを指しており、EYESY ビデオ出力ポートの 1 つへと描画メッセージを出力します。

これは最も単純なモードであり、その結果も同様に単純です。画面の中央付近、すべてのフレームに、永遠に赤い円が描かれます。

5.2 EYESY の API について

EYESY のモードの一般的なフレームワークと要件について説明したので、今度は EYESY を操作するときに見える API(アプリケーション・プログラミング・インターフェイス)を見てみましょう。etc オブジェクトにはいくつかの変数が含まれており、それらはすべてどのモードからでもアクセスできます。

- etc. audio_in - EYESY のオーディオ入力チャンネルによって登録された最新の 100 のオーディオレベルのリスト。左右の入力チャンネルは 1 つのモノチャンネルに統合されます。100 個のオーディオ値は、最小-32,768 から最大+32,767 の範囲で、16 ビットの整数値で格納されます。トリガーボタンを押すと、このリストに正弦波が入力され、EYESY へのオーディオ入力がシミュレートされます。
- etc. audio_trig - トリガーイベントを示すブール値 (*boolean value*)。トリガーソースは、Shift + Trigger Select Knob で選択します。オーディオがソースとして選択されている場合、最後のフレームが draw() 関数を介して描画されてから、オーディオが固定のスレッショルドレベル(最大値の約 80%)を超えると、トリガーイベントが発生します。さらに、Trigger ボタンを押すと、etc.audio_trig が true に設定されます。
- `etc. xres` - 現在の出力解像度の水平*float*。
- `etc. yres` - 現在の出力解像度の垂直*float*。
- etc. knob1 - ノブ 1 の現在値を表すフロート。ノブが再び動かされるまで、現在選択されている MIDI チャンネルの番号 21 の着信 MIDI コントロールチェンジメッセージがノブ 1 の値を置き換えます。
- etc. knob2 - ノブ 2 の現在値を表すフロート。ノブが再び動かされるまで、現在選択されている MIDI チャンネルの番号 22 の着信 MIDI コントロール変更メッセージがノブ 2 の値を置き換えます。
- etc. knob3 - ノブ 3 の現在値を表すフロート。ノブが再び動かされるまで、現在選択されている MIDI チャンネルの番号 23 の着信 MIDI コントロール変更メッセージがノブ 3 の値を置き換えます。
- etc. knob4 - ノブ 4 の現在値を表すフロート。ノブが再び動かされるまで、現在選択されている MIDI チャンネルの番号 24 の着信 MIDI コントロール変更メッセージがノブ 4 の値を置き換えます。
- etc. knob5 - ノブ 5 の現在値を表すフロート。ノブが再び動かされるまで、現在選択されている MIDI チャンネルの番号 25 の着信 MIDI コントロール変更メッセージがノブ 5 の値を置き換えます。
- etc. lastgrab - 最後に撮影したスクリーンショットの画像を含む Pygame サーフエス ([Screenshot] ボタンを使用)。1280x 720 で、スクリーンショットのフルサイズと一致します。このサーフェスのサイズは
- etc. lastgrab_thumb - 最後に撮影したスクリーンショットのサムネイル画像を含む Pygame サーフエス ([Screenshot] ボタンを使用)。このサーフェスのサイズは 128x72 です。

- `etc.midi_notes` - 128 のさまざまな MIDI ノートピッチを表すリスト。このリストの各値は、そのノートが現在オンであるかどうかを示します。たとえば “middle C” (MIDI ノート 60) が押されているときに実行されるスレッシュホールド関数を次のように作成できます…

```
if etc.midi_notes[60] : yourFunctionHere()
```

- `etc.midi_note_new` - 最後のフレームが描画されてから (`draw()` 関数を介して)、メッセージで少なくとも 1 つの新しい MIDI ノートが受信されたかどうかを示すブール値。
- `etc.mode` - 現在のモードの名前の文字列 (string)。
- `etc.mode_root` - 現在のモードのフォルダへのファイルパスの文字列 (string)。これにより、`/sdcard/Modes/Python/CurrentModeFolder` などが返されます。これは、画像、フォント、またはその他のリソースをモードのフォルダからロードする必要がある場合に役立ちます。(`setup()` 関数はこれを行うのに適切な場所です。)

これらすべての変数に加えて、EYESY オブジェクトには覚えておくべき 2 つの関数があります。:

- `etc.bg_color()` - 背景色を設定します。通常は「`etc.color_picker_bg(etc.knob5)`」と指定されますが、任意のノブを使用して背景色をコントロールできます。この関数は、ノブの値 (0-1 から) を取得して RGB 値に変換し、それを背景色に使用します。
- `etc.color_picker()` - この関数は、指定されたノブの値を色に変換します。通常は「`etc.color_picker(etc.knob4)`」と指定されますが、カラーピッカーには任意のノブを使用できます。この関数が呼び出されると、この色の赤、緑、青のコンポーネントを表す 3 つの整数のタプルが返されます。ファクトリモードでは、この関数によってローカル変数 (通常は色) が設定されていることがよくあります。

```
color = etc.color_picker(etc.knob4)
```

特定のノブに基づいて要素の色を簡単に設定できます。

モードをプログラムするときは、モード全体でカラーピッカーと背景色セクターに同じノブを使用することを検討してください。これは、同じマッピングを使用してモードを切

り替えるときに、前の背景色と次の背景色が一貫することで、トランジションがより流動的になるためです。

これは EYESY モードのエディットをする際にあなたが利用できる API コンポーネントを表しています。値がハードウェアコントローラーに基づく変数の場合、シーンを変更すると、しばらくの間使用されていた値が上書きされる可能性があることに注意してください(シーンの詳細については、第 2 章を参照してください)。また、draw() 関数は、生成されるすべてのビデオフレームに対して呼び出されるため、draw() 関数内の変数をクエリすることは、モードの出力を動的に更新する効果的な方法です。

最後に、この説明書では、Python の実用的な知識を持ち、それを調整し、まったく新しいモードのアイデアを考え出し、それを機能させる方法については、具体的に説明していません。モードを作成するためには、プログラム言語について学び、あなたのアイデアを活かす必要があります。またそれほど難しく考えなくても、学ぶにつれ「ハッピー・アクシデント」に遭遇して芸術が生まれることもあります。

6. 付録資料

6.1 SD カードのディスクイメージの書き込み

microSD カードは、モードの保存に加えて、EYESY のオペレーティングシステムも保存します。

microSD カードに新しいディスクイメージを書き込むと、EYESY が「工場出荷時」の状態にリセットされます。これには OS 用とモードストレージ用などのパーティションが含まれています。これは、最新の EYESY OS にアップデートしたり、microSD カードの問題を修正したりするのに役立ちます。

これにより、microSD カードの中身は完全に消去されるため、必要なものはすべてバックアップしてください。 モード、スクリーンショット、その他のファイルのダウンロードについては、第 4 章を参照してください。古い OS を利用できるようにしておきたい場合は、新しいカードを使用することもできます。EYESY に同梱されているカードと同じ仕様の新しいカードを使用する場合、それらの仕様は次のとおりです。

SanDisk Edge 8GB microSDHC C4UHS-I

次の手順に従って、新しい SD カードを作成します:

1. microSD カードのディスクイメージをコンピューターにダウンロードします
 - OS が [EYESY-2.2](#) の場合は 8GB またはそれ以上の microSD カードが必要です
 - OS が [EYESY-2.1](#) の場合は 8GB またはそれ以上の microSD カードが必要です
2. **オプション:** ディスクイメージを解凍する場合は、次のいずれかのプログラムを使用してください:
 - Windows: [7-Zip](#)
 - Mac: [The Unarchiver](#)
 - Linux: [Unzip](#)
3. フラッシャープログラムをコンピューターにダウンロードします。
<https://www.balena.io/etcher/>
4. EYESY の電源をオフにします。

5. エンクロージャーの背面 (MIDI 入力ポートと HDMI ポートの間) にある細いスリットの位置を確認します。
6. ピンまたはペーパークリップなどを使用して黒い microSD カードを押し込み、カードを取り出します。カードがゆっくりと飛び出します。
7. microSD をコンピューターに挿入します (アダプターまたはカードリーダーが必要な場合があります。これらは別売です)
8. Etcher プログラムを使用して、解凍した OS を microSD カードに書き込みます。Etcher が終了すると、コンピュータに「このディスクは読み取れません」のようなメッセージが表示される場合があります。このメッセージは正常であり、[取り出し] をクリックして続行できます。
9. microSD カードをコンピューターから取り外し、EYESY に再度挿入します。SD カードが誤ってデバイスの中に落ちてしまわないように、回路基板上のソケットに確実に挿入されることを慎重に確認してください。SD カードが安定していなければ、それはおそらくソケットにはまっていないので十分に注意してください！ 同じピン/ペーパークリップを使用して、「カチッ」という音が聞こえる (感じる) まで押し込みます。
10. EYESY の電源を入れます

6.2 ビデオキャプチャについて

優れたビジュアルが完成したら、コンピューターを使用して作業をビデオとして記録したり、ビジュアルをストリーミングしたり、ビジュアルを VJ ソフトウェアにルーティングしたりすることができます。これらのことを行うには、EYESY のビデオ出力をキャプチャする必要があります。残念ながら、ほとんどのコンピューターにはビデオ入力がありません。したがって、このプロセスはビデオ「キャプチャカード」(別売) を使用して行う必要があります。

キャプチャカードを選択する前に、次の事に注意してください。:

- **ビデオ入力:**一部のキャプチャカードは HDMI ビデオのみを受信します。他のカードは HDMI とコンポジットビデオを受信できます-これにより、EYESY の両方の出力フォーマットをキャプチャできます (ただし、一度に 1 つのフォーマットのみ)。
- **ソフトウェアの互換性:**ビデオフィードは、他のプログラムで利用できる場合とできない場合があります。Elgato キャプチャカードなどのデバイスは、Elgato ソフトウェアを使用してビデオファイルを記録することしかできません。また

は、Blackmagic キャプチャカードなどのデバイスをサードパーティのソフトウェアで使用することもできます。つまり、EYESY ビデオは、Adobe Premiere、Quicktime など録画したり、OBS などのプログラムでストリーミングしたり、VJソフトウェアの他のビジュアルと組み合わせたりすることができます。

*これらの互換性については各ビデオカードやソフトウェアのメーカーにお問い合わせください。

ほとんどのキャプチャカードにはオーディオ入力とビデオ入力の両方がありますが、EYESY には音声出力はありません。つまり「録音」するために追加のオーディオラインをキャプチャカードに送信する必要があります。

以下は典型的なビデオ・キャプチャのワークフローの一例です

1. キャプチャカードをコンピュータに接続します。
2. EYESY のビデオ出力をキャプチャカードの入力に接続しますが、まだ電源をオンにしません
3. **もし必要な場合:**オーディオ出力をキャプチャカードに接続します
4. 互換性のあるビデオソフトウェアを開きます
5. キャプチャカードからビデオやオーディオを受信できるようにソフトウェアを構成します。一般的には「キャプチャ設定」ウィンドウなどの名称です。
6. EYESY の電源をオンにします。
7. EYESY のビデオ出力がビデオソフトウェアで受信されていることを確認します
8. VJソフトウェアでビデオやオーディオを録画、ストリーミング、または他のビデオとミックスします。

6.3 WiFi と SSH の設定について

EYESY は SSH サーバーを実行しているため、Web インターフェイスを使用せずにログインして WiFi をセットアップできます。まず、起動時に Shift キーを押しながら、AP モードで EYESY を起動します。これで、コンピュータでパスワード「coolmusic」を使用して EYESY ネットワークに参加します（詳細については、セクション 4.1 を参照してください）。コンピュータが EYESY ネットワークに接続されると、端末を使用して SSH 経由で EYESY にログインできます。

UNIX ベースの端末 (Mac または Linux OS) では、次のコマンドを実行します。

```
ssh music@eyesy.local
```

次に、パスワード「music」を入力します

これで接続されます。WiFi ネットワークを設定するには、wpa_passphrase コマンドを使用します。

```
wpa_passphrase "Network Name" "Password" >>  
/sdcard/System/wpa_supplicant.conf
```

ネットワーク名とパスワードをあなたのネットワーク・クレデンシャルに置き換えます。

「exit」と入力してログアウトできます。次に EYESY をオンにすると、このネットワークへの参加が試行されます (Shift キーを押している場合を除き、その場合は AP モードで起動します)。