

## 29-Utilisation du canal Alpha

Dans ce tutoriel, nous verrons comment composer deux images en utilisant le *canal alpha* d'une matrice **char** Jitter à 4 plans comme masque de transparence. Nous allons explorer ce concept comme un moyen de superposer les sous-titres générés par l'objet *jit.lcd* sur une image de film.

La région supérieure gauche du patch du didacticiel contient un objet *jit.movie* qui lit le fichier *ozone.mov* à l'ouverture du patch. L'objet *metro* génère une nouvelle matrice à partir de l'objet *jit.movie* toutes les **33** millisecondes et interroge l'attribut **time** de l'objet (la position actuelle de lecture du film, en unités de temps QuickTime) en utilisant un objet *trigger*:



*Lecture du film, obtention de la position temporelle actuelle avec chaque nouvelle matrice*

- Commencez à visionner le film en cliquant sur la boîte *toggle* attachée au *metro*. L'objet *jit.movie* commencera à sortir des matrices ainsi que la position de lecture actuelle dans le film. Vous devriez voir le film (avec des sous-titres!) apparaître dans la fenêtre *jit.pwindow* en bas du patch.

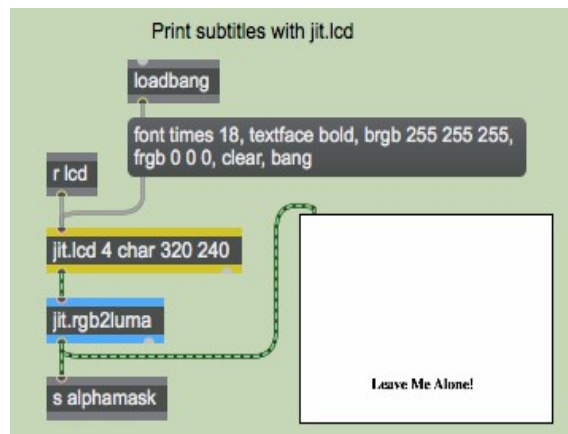


*Joli minou*

Tout d'abord, nous allons voir comment les sous-titres sont générés. Puis, nous étudierons comment les composer avec l'image du film en utilisant le canal alpha.

### L'objet *jit.lcd*

Les sous-titres de notre patch sont générés en envoyant des messages à l'objet *jit.lcd* (en haut du patch). Les arguments de *jit.lcd* spécifient **planecount**, **type** et **dim** de la matrice générée par l'objet (*jit.lcd* ne prend en charge que les matrices **char** à 4 plans). L'objet *jit.lcd* prend des messages sous forme de commandes QuickDraw, et les dessine dans une matrice de sortie lorsqu'il reçoit un **bang**. Nous initialisons notre objet *jit.lcd* en lui donnant des commandes pour définir **font** et **textface** pour dessiner le texte et sa couleur d'avant-plan (**frgb**) et d'arrière-plan (**brgb**) (dans les listes de valeurs RVB). Nous effaçons ensuite l'image interne de l'objet *jit.lcd* et envoyons une matrice vide avec un **bang**:

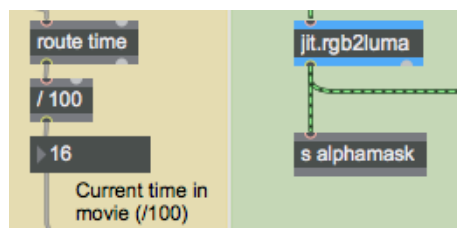


Un objet *jit.lcd* initialisé.

Remarque: l'objet *jit.lcd* possède le même ensemble complet de commandes et de fonctions QuickDraw que l'objet Max *lcd*. Bien que nous l'utilisons dans ce patch dans le but de générer des images de texte, *jit.lcd* peut également être utilisé pour générer toutes sortes de graphiques vectoriels bidimensionnels. Le didacticiel Max *Mouse Drawing - Introduction to Drawing*, illustre certaines des fonctionnalités de l'objet *lcd*, qui peuvent toutes être appliquées aussi facilement à *jit.lcd*.

L'objet *jit.lcd* génère sa matrice dans un objet *jit.rgb2luma*, qui convertit l'image à 4 plans produite par *jit.lcd* en une image à 1 plan en niveaux de gris. L'objet *jit.rgb2luma* génère une matrice contenant la valeur de luminosité de chaque cellule de la matrice d'entrée. Cette matrice 1-plan est ensuite envoyée à un objet *send* avec le nom **alphamask** et à un objet *jit.pwindow* afin que nous puissions la visualiser. Notez que l'objet *jit.pwindow* a son attribut **border** fixé à **1**. Par conséquent, nous pouvons voir une bordure noire de 1 pixel autour de l'image blanche à l'intérieur.

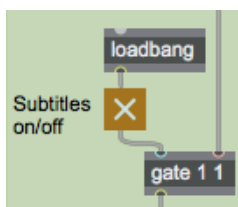
Notre objet *jit.lcd* reçoit également des messages d'autres parties du patch (via l'objet *receive* nommé **lcd** qui lui est attaché). Les sous-titres sont générés automatiquement en recherchant certains moments dans la lecture du film:



Analyser les valeurs de temps de l'objet *jit.movie*.

L'objet *jit.movie* affiche sa position de lecture actuelle à chaque tick de l'objet *metro*, grâce au **getttime** que nous avons entre les deux. L'attribut **time** est envoyé à la sortie droite de l'objet *jit.movie*, où nous pouvons utiliser un objet *route* pour le dépouiller de son sélecteur de message (**time**). Nous divisons la valeur par **100** afin de pouvoir rechercher un temps exact avec plus de précision. Étant donné que *metro* n'interroge le temps que toutes les **33** millisecondes, il est tout à fait possible que nous sautions complètement une durée spécifique - diviser de la valeur du temps par **100** permet de trouver plus rapidement le point du film que nous voulons.

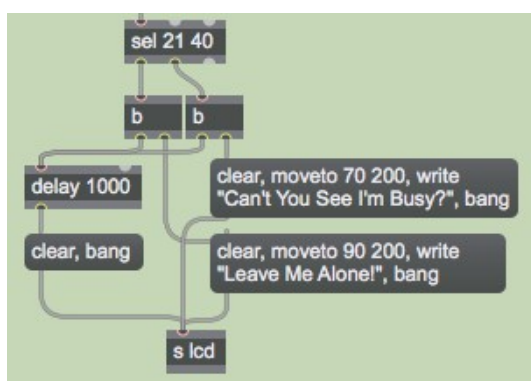
Les valeurs temporelles sont envoyées via un objet *gate* où vous pouvez désactiver les sous-titres si vous le souhaitez:



Contrôlez le flux des valeurs de temps avec un objet *gate*.

- Cliquez sur la boîte *toggle* attachée à *gate*. Les sous-titres devraient disparaître. Vous pouvez réapparaître les sous-titres en cliquant à nouveau sur la boîte *toggle*.

Les sous-titres sont finalement générés lorsque les valeurs de temps **21** et **40** dépassent l'objet *gate*. L'objet *select* envoie un **bang** lorsque ces valeurs arrivent. Cela déclenche des commandes des boîtes de *messages* vers *jit.lcd*:

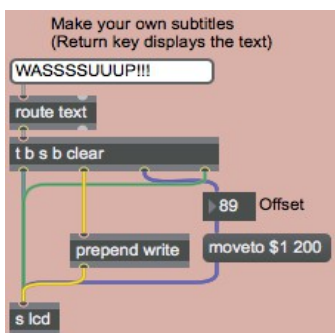


Effectuer le sous-titrage en fonction des valeurs de temps.

Le message **clear** vers *jit.lcd* efface la toile de dessin, remplissant tous les pixels de blanc (la couleur de fond que nous avons choisie). Le message **moveto** déplace le curseur de l'objet *jit.lcd* à une coordonnée spécifique à partir de laquelle il dessinera les commandes suivantes. Le message **write** dessine du texte dans la matrice en utilisant **font** et **textface** actuellement sélectionnées. Une fois que nous avons écrit nos sous-titres, nous envoyons à l'objet un **bang** pour qu'il produise une nouvelle matrice. Avec chaque sous-titre, nous envoyons également un **bang** à un objet *delay*, qui efface et renvoie la matrice **1000** millisecondes plus tard, effaçant ainsi le sous-titre.

## Créez vos propres sous-titres

La région du patch du didacticiel à droite (sur fond rose) vous permet d'utiliser l'objet *textedit* pour générer vos propres sous-titres. La boîte de *nombre* intitulée **Offset** détermine le décalage horizontal du texte. L'objet *trigger* vous permet d'envoyer toutes les commandes QuickDraw nécessaires à l'objet *jit.lcd* dans le bon ordre.



Créez vos propres sous-titres

- Désactivez le sous-titrage automatique à l'aide de la boîte *toggle* située au-dessus de *gate*. Tapez du texte dans la zone de saisie et appuyez sur la touche Retour. Le texte apparaîtra en surimpression sur l'image.



*Le nouveau sous-titre sur l'image*

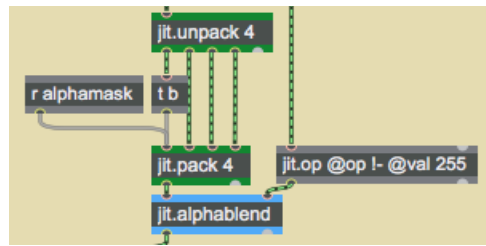
Maintenant que nous avons compris comment les titres sont générés, voyons comment ils sont composés sur le film.

## **Le canal Alpha**

Le *canal alpha* d'une image ARGB définit sa transparence lorsqu'elle est composée avec une deuxième image. Si un pixel a un canal alpha de 0, il est considéré comme totalement transparent lorsqu'il est combiné avec une autre image. Si le canal alpha d'un pixel est défini sur **255**, il est considéré comme complètement opaque et s'affichera dans toute son opacité lors de la superposition. Les valeurs intermédiaires font en sorte que le pixel s'estompe progressivement entre la première et la deuxième image. Dans les matrices **char** Jitter à 4 plans, les données stockées dans le plan **0** de la matrice sont considérées comme le canal alpha.

Détail technique: les systèmes de couleur et les environnements logiciels diffèrent quant à savoir si le canal alpha décrit la transparence ou l'opacité d'une image. Dans les films (et donc dans Jitter), une valeur alpha de 255 signifie que le pixel est entièrement opaque. Vous pouvez rencontrer des programmes où l'inverse est vrai (c'est-à-dire qu'une valeur alpha de 255 indique une transparence totale). L'attribut **mode** par défaut de **0** traite les valeurs alpha croissantes comme plus opaques, tandis que la définition de l'attribut **mode** à **1** fait que l'objet traite les valeurs alpha croissantes comme plus transparentes.

L'objet *jit.alphablend* utilise les valeurs stockées dans le canal alpha (plan 0) de la matrice arrivant dans l'entrée gauche pour effectuer un fondu enchaîné (sur une base cellule par cellule) entre les matrices arrivant dans ses deux entrées. Notre patch remplace le plan 0 de la matrice de sortie de l'objet *jit.movie* par la sortie de l'objet *jit.lcd*. Nous utilisons ensuite ce nouveau canal alpha avec l'objet *jit.alphablend* pour effectuer un fondu enchaîné entre le film et une copie inversée de lui-même:

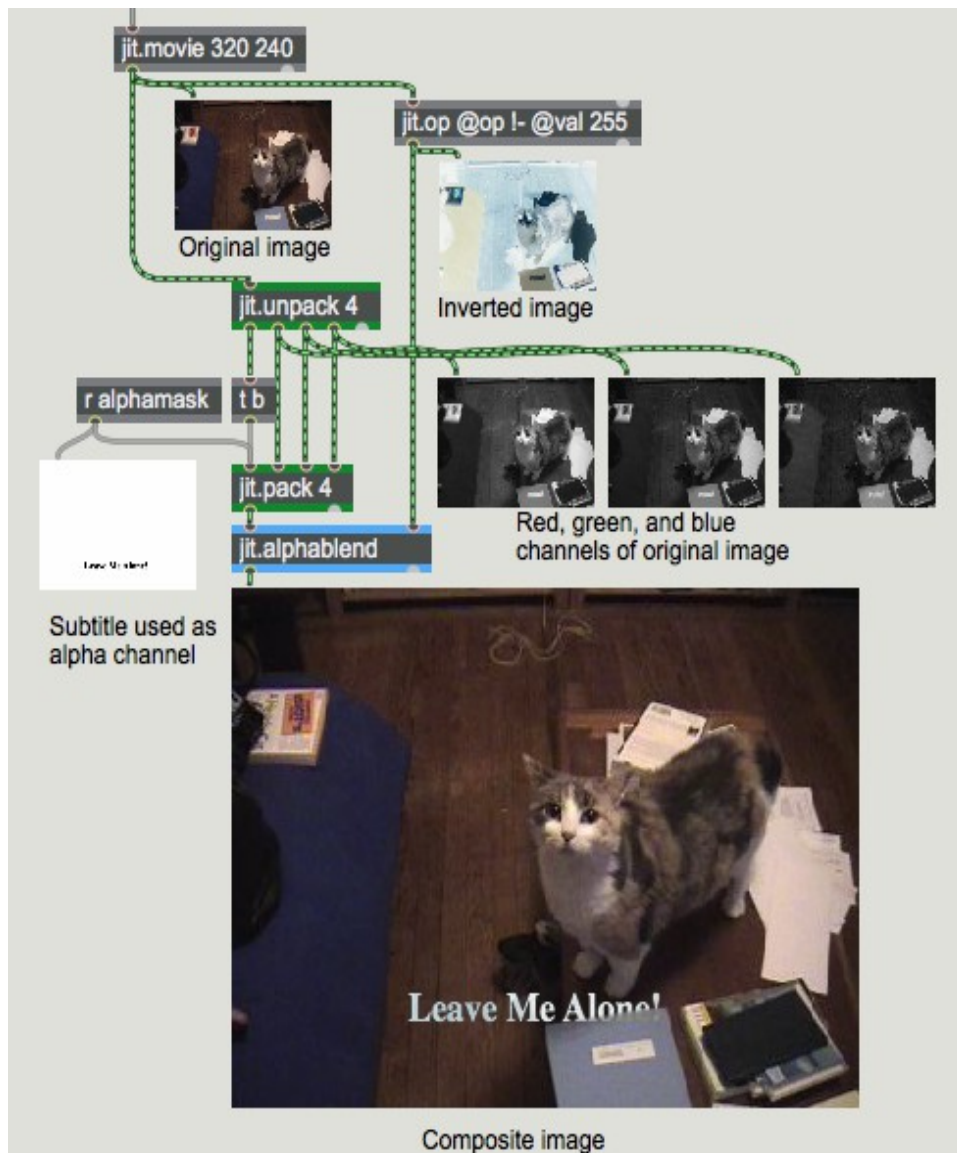


Insertion d'un nouveau canal alpha avec l'objet *jit.pack*.

Nous utilisons les objets *jit.unpack* et *jit.pack* pour retirer le canal alpha d'origine de notre film. La matrice 1-plan contenant le sous-titre arrive à l'objet *jit.pack* depuis l'objet *receive* situé au-dessus de lui. Remarquez comment l'objet *trigger* est utilisé pour forcer *jit.pack* à sortir de nouvelles matrices même si aucune nouvelle matrice n'est arrivée de *receive* (*jit.pack*, comme l'objet *Max pack*, ne sortira une matrice que lorsqu'il aura reçu une nouvelle matrice ou un **bang** dans son entrée la plus à gauche). L'objet *jit.op* crée un négatif de la matrice d'origine du film (en soustrayant les valeurs de matrice de **255** à l'aide de l'opérateur `!-`). L'objet *jit.alphablend* utilise ensuite notre nouveau canal alpha - les valeurs blanches dans la matrice de sous-titres permettent de conserver l'image originale, tandis que les valeurs noires apportent l'image inversée de la matrice de droite.

Différentes techniques sont souvent utilisées pour le sous-titrage. La technique consistant à superposer un texte blanc sur une image (parfois entourée d'une bordure noire) est beaucoup plus courante que la technique utilisée ici, qui consiste à remplir un masque alpha avec une image inversée. Cependant, cette méthode de sous-titrage nous donne une excuse parfaite pour utiliser l'objet *jit.alphablend*, et peut vous donner des sous-titres plus lisibles dans les situations où l'image de fond a des zones de fort contraste.

L'image ci-dessous montre le processus de composition avec des objets *jit.pwindow* montrant les étapes intermédiaires:



*Le processus de composition, montrant les étapes intermédiaires*

## Sommaire

L'objet `jit.lcd` offre un ensemble complet de commandes QuickDraw pour dessiner du texte et des graphiques à 2 dimensions dans une matrice Jitter. L'objet `jit.rgb2luma` convertit une matrice ARGB à 4 plans en une matrice à niveaux de gris à 1 plan contenant des données de luminance. Vous pouvez remplacer le canal alpha (plan 0) d'une image par une matrice à 1 plan à l'aide de l'objet `jit.pack`. L'objet `jit.alphablend` effectue un fondu enchaîné de deux images, cellule par cellule, sur la base du canal alpha de la matrice de gauche. s