Projet MIDL 2

Tom le Ber & Tony Perottino

Introduction :

Notre projet se base sur la librairie MMPose d'OpenMMLab, spécialisée dans le suivi et l'estimation du mouvement. Dans notre projet, nous nous concentrons sur l'analyse de vidéos des différents mouvements de mains réalisés par des pianistes lorsqu'ils jouent un morceau, vues de dessus.
Nous verrons que les modèles de MMPose permettent avec précision certains mouvements dans certaines configurations (par exemple la création du squelette de la main dans une configuration de piano avec une main unique), mais sont assez vite limités quand la tâche devient difficile (flou de mouvement sur la main, analyse des appuis sur les touches, prédiction de la prochaine note, etc.).

Au cours de notre rapport, nous avons exploré (dans l'ordre) les thèmes suivants :

- Installation d'MMPose, ses dépendances et des autres bibliothèques utilisées.
- Téléchargement de 10 différents modèles d'MMPose pour les mains.
- Application et comparaison des différents modèles de mains sur un dataset original, créé par Tony.
- Reconstruction vidéo d'un modèle donné sur une vidéo du dataset original.
- Analyse de trajectoire d'une main sur un piano (comment se elle se déplace, quelles touches sont pressées, par quels doigts et à quel moment).
- Création et implémentation d'un modèle de prédilection fonctionnel en apprentissage supervisé (sorte extrapolation linéaire à coefficient dynamique).

Tout notre travail est trouvable sur en ligne, sur le Google Colab suivant.

Méthodologie :

Liste des modèles potentiels :

Suite au téléchargement de la librairie MMPose et des autres dépendances, nous importons et comparons 10 différents modèles **TopDown 2D Hand KeyPoint**, tous issus de la version 1.0.0 d'MMpose et compatibles avec la dernière version (1.3.2) que nous utilisons.
La moitié des modèles sont basés sur le dataset **OneHand10k** tandis que l'autre moitié est basée sur le dataset **RHD2D**.
Parmi les 5 modèles de chaque dataset, 3 viennent de la catégorie **HRNetv2-w18** (standard, DARK et UDP), 1 de **ResNet-50** et 1 de **MobileNetV2**.

Dataset :

Pour tester et comparer ces 10 modèles, nous avons composé un dataset original de vidéos (6/9 de notre création), toutes accessibles à ce lien Google Drive.
Les vidéos de pianistes sont les suivantes :

- `chopin do#m rousseau.mp4` & `rachmaninoff solm kassia.mp4` : Ces deux vidéos non-originales possèdent une vue de dessus du piano. Dans ces deux vidéos, les mains ressortent car les prises sont faites dans l'obscurité et possèdent une visualisation des notes grâce à des logiciels proches de synthesia, embers ou seemusic. Elles ne correspondent malheureusement pas à ce qui est possible de faire avec les modèles mmpose qui ne peuvent regarder qu'une main à la fois.
- `pen spinning.mp4` : Cette troisième vidéo, non originale, est une vidéo de pen-spinning, qui nous sert à tester les modèles sur des positions variables de la main.

Elle est particulièrement utile dans la comparaison des modèles pour visualiser ceux qui sont non performants.

- `main.mp4` & `wide putin tony.mp4` : ces deux vidéos sont des vidéos originales. La première est une capture d'une main avec un énorme contraste entre la main et l'arrière-plan, qui nous a servi à confirmer que la présence d'un contraste et un cadrage de la main suffisamment grand (où la main prend la majeure partie de l'écran) aide énormément à la bonne capture des points clés. La deuxième est une vidéo prise sur un piano, cadrée assez proche (4 octaves) proche et sous la lumière du jour qui nous sert de vidéo témoin qui démontre la qualité exacte du modèle.
- `doigt 1.mp4`, `doigt 2.mp4` & `doigt sauts.mp4` : Ces trois vidéos sont des vidéos originales vues du dessus à la lumière artificielle sur un clavier complet, qui ont la particularité d'être jouées uniquement avec un seul doigt (index). Leur intérêt est de faciliter les prédictions de note et d'analyse d'appuis, en réduisant la quantité d'informations à un seul doigt. On peut alors comprendre les limites réelles du modèle de MMPose dans la prédiction de trajectoires. Ces 3 vidéos sont associées avec un fichier MIDI, qui permet d'avoir la réalité terrain. Un fichier au format MIDI est un format de données numériques qui ne contient pas de son audio, mais une série d'instructions (notes, rythme, vélocité) permettant à des instruments électroniques ou des logiciels de reproduire une œuvre musicale. Dans notre cas, on utilisera ce fichier comme moyen de vérification des modèles de prédictions.
- `Impro tony 2 mains.mp4` : La dernière vidéo est une filmée dans le mêmes conditions que les 3 précédentes à la différence que les deux mains jouent séparément sur leur moitié de l'écran (on peut alors analyser les mains en deux étapes), et est la vidéo la plus proche des objectifs fixés des vidéos 1 et 2 tout en étant compatible avec nos contraintes. Il s'agit d'une improvisation.

Sélection du meilleur modèle :

Pour l'étape de sélection des modèles, nous avons utilisé la vidéo `pen spinning.mp4`, car elle nous permet de tester tous les modèles sur des positions variables de la main et donc de facilement visualiser les moins performants, comme dit dans le dernier paragraphe.

Les vidéos `main.mp4` et `wide putin tony.mp4`, présentant beaucoup de contraste entre la main et le reste de l'image nous ont également appris qu'il était essentiel d'avoir un bon contraste dans notre vidéo ainsi qu'un bon cadrage de la main pour avoir des résultats optimaux.

Après plusieurs tests, le modèle le plus concluant à été le modèle **Onehand10k_Hrnetv2_Dark** venant de la catégorie **HRNetv2-w18** du dataset **Onehand10k**.

Application d'MMPose sur une vidéo de pianiste :

Une fois notre modèle choisi, on l'analyse sur chaque frame de la vidéo de notre choix afin d'en extraire à chaque fois les différents joints (points de jonction) de la main.

Pour améliorer la détection de notre main sur chaque frame, nous définissons une boîte encadrant la main (bien plus petite que la frame complète) permettant à MMPose d'obtenir une bien meilleure détection, la surface à analyser étant conforme aux dimensions d'entrée des modèles (ici de 256×256 pixels).

Pour déterminer les boîtes encadrant la main, nous utilisons une démarche proche d'un raisonnement par récurrence.

- L'hérédité est assez simple : si on cherche la boîte optimale pour la frame `t+1`, et que l'on considère que nous possédons la boîte optimale de la frame `t`, alors on peut prendre les extremum de l'ensemble des points en `x` et en `y`, appliquant un cadre à partir de ces extremums puis en l'agrandissant de 10%.

- Si la frame `t-1` n'existe pas ou ne contient pas de main détectée, on commence par analyser la main sur la frame entière puis on réduit progressivement le cadre dans lequel la main a été détectée en réalisant un pavage.

Pour la suite, nous sauvegardons pour chaque frame dans un fichier `.csv` les positions de chaque joint détecté.

Analyse des trajectoires :

Nous avons réalisé deux analyses de trajectoire.

La première analyse que nous avons réalisée est une évolution de la vitesse des différents joints de la main étudiée en fonction du temps. Pour cela, on utilise la fonction suivante, issue de la distance Euclidienne :

$$\sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2}$$

Vous retrouverez différents graphiques de vitesse dans la section **Résultats et visualisations**.

La deuxième méthode d'analyse de trajectoire que nous avons développée est beaucoup plus complexe et repose sur un **étalonnage des touches du piano** accompagné du **filtre de Savitzky-Golay** (au sens traitement du signal). Elle nous permet d'estimer pour chaque doigt, indépendamment les uns des autres, quelles touches du piano il a pressé à chaque moment tout au long de la vidéo.

On suppose ici que la vidéo représente une vue de dessus d'un piano, aligné horizontalement avec le cadre de la vidéo, et restant fixe toute la durée de la vidéo.

La procédure d'estimation des touches pressées par un doigt tout au long de la vidéo est la suivante :

- Pour chaque frame `t`, nous récupérons dans le fichier `.csv` résultat d'estimation de position d'MMPose les coordonnées estimées `(x, y)` du poignet (joint 0) et du bout du doigt concerné (ex : index = 8).
- On construit la série `y_rel = y_tip - y_wrist` des différences de hauteur entre le poignet et le bout du doigt et l'on estime au-dessus de quelle touche le doigt étudié est pour chaque frame de la vidéo.
- On découpe par segments notre série `y_rel`, chaque segment représentant un ensemble continu des frames au-dessus d'une même touche.
- On applique le filtre de Savitzky-Golay sur chaque segment, avec une fenêtre impaire de taille 3 à 7 et un polynôme de degré 2.
    - On choisit un polynôme de degré 2 car l'on remarque que lorsque le pianiste appuie sur une touche, son doigt s'enfonce, réduisant ainsi la distance entre `y_tip` et `y_wrist` (et lorsqu'il le relève, la distance s'agrandit de nouveau). Un mouvement d'appui de touche forme donc une parabole sur la série `y_rel`.
- Une fois notre série `y_rel` lissée, on mesure l'amplitude des pics, et nous gardons seulement ceux supérieurs à un seuil arbitraire.
- Lors de la comparaison des estimations aux données réelles, on constate que les prédictions sont souvent en avance de quelques frames (le pianiste prévoyant légèrement en avance la prochaine touche jouée). On avance donc artificiellement de quelques frames l'estimation obtenue.

Suite à l'application de cette méthode, nous déterminons ~95% des moments où une touche est réellement pressée, avec la bonne note de détectée ~80-90% du temps (les 10-20% restants détectent également l'appui mais par erreur sur une touche adjacente). Il y a également ~15% de fausses notes prédites.

Modèle de prédiction (Estimation de la prochaine note jouée) :

Dans cette partie, le but principal est de trouver la prochaine note jouée par le pianiste si on coupe la vidéo un peu avant qu'il joue cette note.

Pour cela, le modèle observe la main à deux instants clés : le départ (`P0%`) et un point d'intérêt à `80%` du trajet temporel (`P80%`). Le ratio de `80%` a été choisi de telle sorte à ce que le modèle ait suffisamment d'informations sans avoir les plus importantes.

Naïvement, prolonger la trajectoire donne juste de mauvais résultats car la vitesse le long de la trajectoire n'est pas constante. De plus, ajouter un coefficient fixe qui réduit l'impact du mouvement à partir du point le plus proche de la note (celui à `80%`) est une bonne stratégie mais moins performante. Cette stratégie pourrait s'écrire :

$$xy_{pred} = xy_{80\%} + (xy_{80\%} - xy_{0\%}) \times coef$$

Avec `coef = 0.12`, par exemple.

Un modèle préférable car plus performant doit adapter son coefficient dynamiquement. **Techniquement le modèle est une forme d'extrapolation linéaire à coefficient dynamique.**

Il s'écrit alors :

$$X_{pred} = x_{80\%} + (x_{80\%} - x_{0\%}) \times C(t_{restant})$$
$$Y_{pred} = y_{80\%} + (y_{80\%} - y_{0\%}) \times C(t_{restant})$$

Avec :

$$C(t_{restant}) = \begin{cases} 0 & \text{si } t_{restant} \leq 0.2 \\ 0.4 \times \left(\frac{t_{restant} - 0.2}{0.3}\right)^2 & \text{si } 0.2 < t_{restant} < 0.5 \\ 0.4 & \text{si } t_{restant} \geq 0.5 \end{cases}$$

Le temps `trestant` est la différence de temps entre le point à `80%` et celui à `100%`. Cette caractéristique est intéressante car elle va nous permettre d'adapter le coefficient de l'interpolation avec la fonction `C` à l'aide d'observations physiologiques générales d'un pianiste.

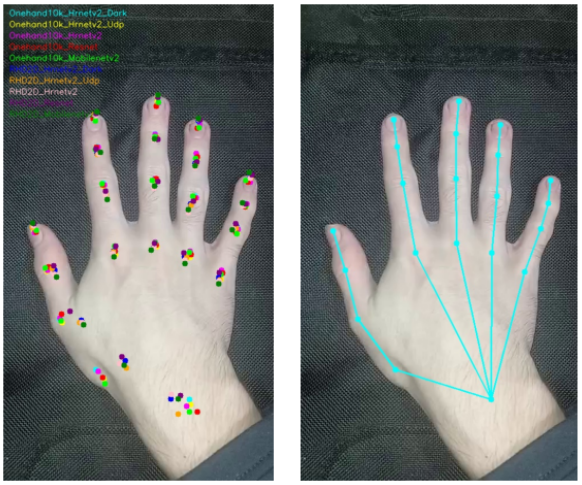
- **L'anticipation de la note jouée :** À l'approche de la note (`< 0.2s`), on considère que le pianiste est déjà positionné sur la bonne note ou l'est quasiment car tout pianiste anticipe son déplacement pour augmenter sa précision. Ici, l'extrapolation devient inutile : on récupère directement la note à `80%` (le coefficient vaut alors `0`).
- **L'accélération progressive vers la note :** Si l'écart entre deux notes devient important (entre `0.2s` et `0.5s`), il y aura alors progressivement moins d'informations à l'instant du point à `80%` de la note finale. On doit alors augmenter progressivement le coefficient de la projection pour augmenter l'importance de la vitesse dans la détection de la note finale.
- **Les notes suivantes sont souvent proches de la note initiale :** À comprendre que plus le déplacement est grand, plus tôt doit se déplacer la main. A plus de `0.5s`, on a un réel manque d'information, ce qui implique que soit le déplacement est grand alors on considère que le pianiste est déjà au milieu de son déplacement, soit il n'a pas encore bougé dans ce cas là la note est assez proche : d'où la valeur du coefficient à `0.4`. Ce coefficient est alors plafonné pour ne pas qu'il explose et donne une note beaucoup trop éloignée, voire hors du piano (en particulier avec l'axe `y`). On limite par là le manque d'informations du fait de la valeur élevée de `trestant`.

A noter que nous avons opté pour une progression quadratique de `C` sur `[0.2;0.5]` qui représente mieux la réalité terrain d'après nous, car cela permet de corriger la note prédite progressivement.

Résultats et visualisations :

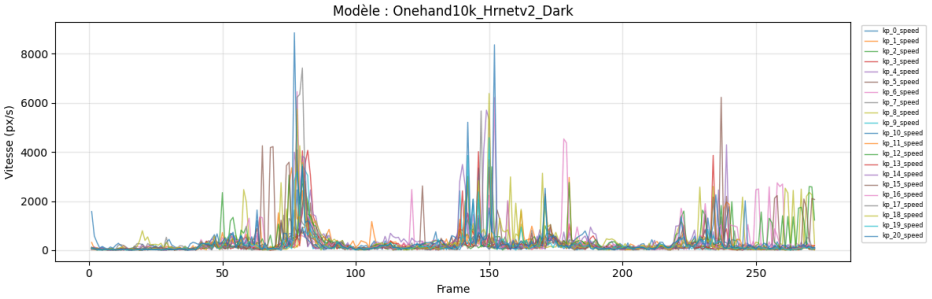
L'entièreté de nos résultats est trouvable dans le Google Drive suivant, dans le dossier results.

Sélection du meilleur modèle & Application d'MMPose sur une main :

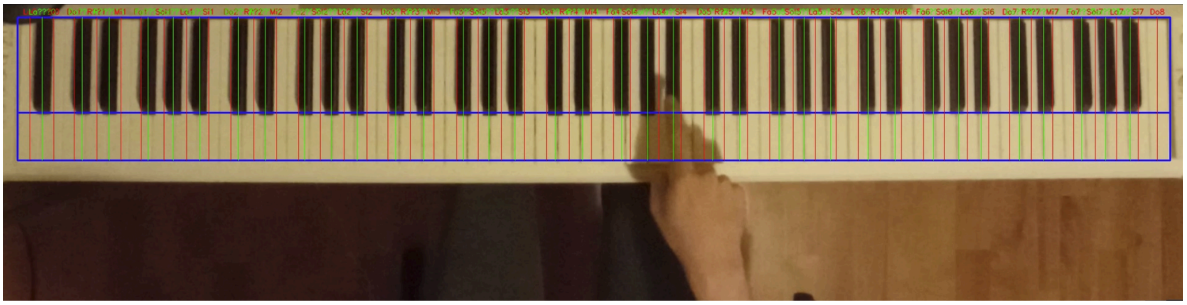


Analyse des trajectoires :

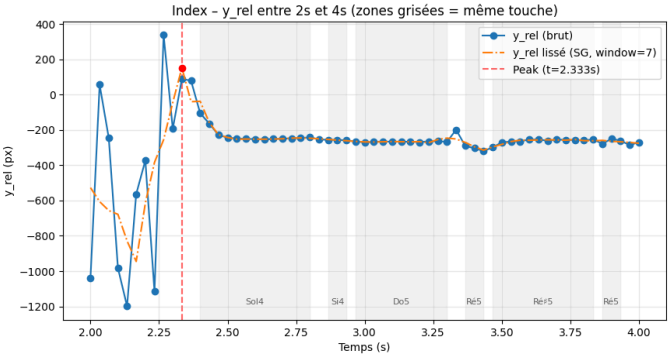
Evolution de la vitesse de chaque joint en utilisant le meilleur modèle pour `doigt 1.mp4` :



Etalonnage du piano pour `doigt 1.mp4` :



Lissage du bruit d'MMPose via le filtre de **Savitzky-Golay** sur `doigt 1.mp4` entre `3s` et `7s` :



Résultat de l'estimation des notes jouées sur `doigt 1.mp4` :

```
Notes détectées (41) :
[ 2.530s]  Sol4   Index       frame_76.jpg    (1025.60, 605.80)  amp=?px       NON-DETECTEE
[ 3.133s]  Do5    Index       frame_94.jpg    (1130.05, 600.34)  amp=10.69px   OK
[ 3.566s]  Ré♯5   Index       frame_107.jpg   (1197.83, 506.09)  amp=14.57px   OK
[ 4.233s]  Ré5    Index       frame_127.jpg   (1170.23, 595.74)  amp=73.06px   OK
[ 5.466s]  Ré♯5   Index       frame_164.jpg   (1200.23, 558.80)  amp=16.14px   OK
[ 6.133s]  Ré5    Index       frame_184.jpg   (1171.37, 600.43)  amp=19.90px   OK
[ 6.500s]  Sol4   Index       frame_195.jpg   (1019.41, 608.14)  amp=26.39px   OK
[ 7.200s]  La4    Index       frame_216.jpg   (1045.93, 578.50)  amp=10.61px   INEXISTANTE
[ 7.500s]  Sol♯4  Index       frame_225.jpg   (1051.51, 536.93)  amp=25.83px   OK
[ 7.833s]  La♯4   Index       frame_235.jpg   (1086.79, 530.64)  amp=57.45px   OK
[ 8.266s]  Si4    Index       frame_248.jpg   (1112.60, 550.16)  amp=6.98px    INEXISTANTE
[ 8.533s]  Do5    Index       frame_256.jpg   (1138.50, 592.54)  amp=27.15px   OK
[11.133s]  Si4    Index       frame_334.jpg   (1106.68, 604.21)  amp=25.32px   INEXISTANTE
[11.600s]  Fa4    Index       frame_348.jpg   ( 991.58, 590.65)  amp=6.73px    OK
[12.000s]  Sol♯4  Index       frame_360.jpg   (1048.86, 536.90)  amp=27.76px   OK
[12.533s]  Do5    Index       frame_376.jpg   (1134.11, 576.64)  amp=22.17px   OK
[13.040s]  Do5    Index       frame_392.jpg   (1134.20, 590.00)  amp=?px       NON-DETECTEE (DELAI ENTRE TOUCHES)
[14.400s]  Si4    Index       frame_432.jpg   (1135.90, 595.70)  amp=?px       NON-DETECTEE
[14.966s]  Do5    Index       frame_449.jpg   (1133.35, 608.54)  amp=4.45px    OK
[15.333s]  Ré5    Index       frame_460.jpg   (1170.47, 608.04)  amp=4.46px    OK
[16.666s]  Do5    Index       frame_500.jpg   (1128.94, 606.47)  amp=11.63px   OK
[16.933s]  Do5    Index       frame_508.jpg   (1135.32, 601.85)  amp=6.00px    INEXISTANTE
[18.033s]  Si4    Index       frame_541.jpg   (1099.31, 605.65)  amp=16.32px   OK
[19.866s]  Sol4   Index       frame_596.jpg   (1027.41, 598.78)  amp=19.45px   OK
[20.400s]  Do5    Index       frame_612.jpg   (1128.07, 584.41)  amp=7.75px    OK
[20.866s]  Ré♯5   Index       frame_626.jpg   (1198.39, 518.90)  amp=24.14px   OK
[21.366s]  Ré5    Index       frame_641.jpg   (1183.85, 573.01)  amp=33.87px   OK
[22.433s]  Ré5    Index       frame_673.jpg   (1170.20, 590.98)  amp=30.02px   MAUVAISE TOUCHE (Ré♯5)
[23.333s]  Ré5    Index       frame_700.jpg   (1173.75, 593.81)  amp=13.74px   OK
[23.733s]  Sol4   Index       frame_712.jpg   (1021.74, 602.75)  amp=25.05px   OK
[24.600s]  La4    Index       frame_738.jpg   (1062.83, 523.87)  amp=37.76px   MAUVAISE TOUCHE (Sol♯4)
[24.966s]  La♯4   Index       frame_749.jpg   (1077.24, 505.34)  amp=14.46px   OK
[25.533s]  Si4    Index       frame_766.jpg   (1114.50, 545.49)  amp=11.01px   OK
[26.433s]  Si4    Index       frame_793.jpg   (1102.71, 567.13)  amp=56.12px   INEXISTANTE
[28.466s]  Do5    Index       frame_854.jpg   (1133.34, 598.09)  amp=4.02px    OK
[29.200s]  Sol4   Index       frame_876.jpg   (1038.30, 563.18)  amp=33.42px   MAUVAISE TOUCHE (Sol♯4)
[29.500s]  Fa4    Index       frame_885.jpg   ( 991.92, 597.23)  amp=16.40px   OK
[29.800s]  Sol4   Index       frame_894.jpg   (1032.60, 601.85)  amp=6.21px    OK
[30.733s]  Do5    Index       frame_922.jpg   (1134.32, 603.58)  amp=6.54px    OK
[31.030s]  Do5    Index       frame_931.jpg   (1140.10, 603.60)  amp=?px       NON-DETECTEE (DELAI ENTRE TOUCHES)
[32.133s]  Fa4    Index       frame_964.jpg   ( 989.41, 615.42)  amp=15.07px   OK
[33.066s]  Sol4   Index       frame_992.jpg   (1025.64, 611.77)  amp=12.49px   OK
[33.666s]  Ré4    Index       frame_1010.jpg  ( 913.34, 597.28)  amp=6.39px    OK
[34.066s]  -----  Index       frame_1022.jpg  ( 899.42, 646.68)  amp=16.45px   INEXISTANTE (NON CONSIDEREE)
[34.433s]  Do4    Index       frame_1033.jpg  ( 879.23, 601.11)  amp=15.45px   OK
```

Modèle de prédiction (Estimation de la prochaine note jouée) :

Résultat de la prédiction des notes jouées sur `doigt 1.mp4` :

```
Il y a 39 notes dans le fichier MIDI.
Synchronisation terminée : 39 segments identifiés (Sol inclus).

Frame     | Timing    | Note Réelle  | Note Prédite | X Prédit   | Y Prédit
-------------------------------------------------------------------------------
76        | 2.53      | Sol4         | Sol4         | 1025.6     | 605.8
93        | 3.10      | Do5          | Do5          | 1131.1     | 602.5
108       | 3.60      | Ré♯5         | Ré♯5         | 1196.3     | 505.5
125       | 4.17      | Ré5          | Ré5          | 1170.4     | 596.5
168       | 5.60      | Ré♯5         | Ré♯5         | 1203.3     | 518.3
185       | 6.17      | Ré5          | Ré5          | 1171.5     | 600.3
194       | 6.47      | Sol4         | Sol4         | 1020.1     | 609.5
221       | 7.37      | Sol♯4        | Sol♯4        | 1052.0     | 537.6
240       | 8.00      | La♯4         | Si4          | 1096.7     | 536.6
257       | 8.57      | Do5          | Do5          | 1136.8     | 589.7
330       | 11.00     | Fa4          | Do5          | 1132.2     | 591.6
361       | 12.03     | Sol♯4        | Sol♯4        | 1047.6     | 534.3
377       | 12.57     | Do5          | Do5          | 1134.5     | 587.4
392       | 13.07     | Do5          | Do5          | 1134.2     | 590.0
432       | 14.40     | Si4          | Do5          | 1135.9     | 595.7
447       | 14.90     | Do5          | Do5          | 1130.6     | 605.6
456       | 15.20     | Ré5          | Ré5          | 1172.7     | 608.9
495       | 16.50     | Do5          | Ré5          | 1163.7     | 610.5
518       | 17.27     | Si4          | Si4          | 1098.3     | 601.3
585       | 19.50     | Sol4         | ???          | 1028.3     | 644.6
614       | 20.47     | Do5          | Do5          | 1131.9     | 588.7
629       | 20.97     | Ré♯5         | Ré♯5         | 1189.7     | 509.9
645       | 21.50     | Ré5          | Ré5          | 1172.8     | 599.2
685       | 22.83     | Ré♯5         | Mi5          | 1208.6     | 519.3
700       | 23.33     | Ré5          | Ré5          | 1173.5     | 593.6
709       | 23.63     | Sol4         | Sol4         | 1024.4     | 605.7
735       | 24.50     | Sol♯4        | Sol♯4        | 1057.4     | 527.4
754       | 25.13     | La♯4         | La♯4         | 1092.9     | 510.8
770       | 25.67     | Si4          | Si4          | 1104.0     | 576.8
840       | 28.00     | Do5          | Si4          | 1096.0     | 612.7
868       | 28.93     | Sol♯4        | Sol♯4        | 1054.9     | 527.2
884       | 29.47     | Fa4          | Fa4          | 992.3      | 598.6
900       | 30.00     | Sol4         | Sol4         | 1025.6     | 601.7
915       | 30.50     | Do5          | Do5          | 1137.0     | 603.3
931       | 31.03     | Do5          | Do5          | 1140.1     | 603.6
960       | 32.00     | Fa4          | Sol4         | 1021.3     | 621.2
990       | 33.00     | Sol4         | ???          | 1087.8     | 645.8
1009      | 33.63     | Ré4          | Ré4          | 909.5      | 595.1
1026      | 34.20     | Do4          | Do4          | 877.9      | 599.4
-------------------------------------------------------------------------------
Analyse terminée : 30/39 notes correctement prédites.
Taux de précision : 76.9%
```

Discussion et pistes d'amélioration :

Quelques améliorations que nous pourrions apporter seraient d'étendre notre algorithme pour fonctionner avec 2 mains, y compris lorsqu'elles se superposent.
Sinon, on a accompli tous nos objectifs.