SUPPLEMENTARY MATERIALS

The details such as the spinal level of lesions, classification, pathologic diagnosis, size of lesions, and sizes of designed and intraoperative incisions were gathered and shown for both mobile augmented reality navigation system (MARNS) and x-rays groups in Supplementary Tables 1 and 2, respectively.

Registration of Spinal Computed Tomography and Magnetic Resonance Imaging

Medical images are preprocessed before registration, including smoothing, grayscale normalization, edge detection, and other operations to reduce noise, enhance contrast, and improve image clarity. Key feature points or feature voxels with higher shape feature similarity are then extracted from the image to ensure accurate registration results. To evaluate the degree of similarity between 2 images, we use the mutual information function and normalized cross-correlation for similarity measurement. The mutual information function (1) and the normalized cross-correlation function (2) are as follows.

$$MI(I,J) = H(I) + H(J) - H(I,J) \tag{1}$$

$$NCC(I,J) = \frac{\sum_{i=1}^{N}(I_i - I)(J_i - J)}{\sqrt{\sum_{i=1}^{N}(I_i - I)^2 \sum_{i=1}^{N}(J_i - J)^2}} \tag{2}$$

We use rigid transformation to perform preliminary registration and obtain coarse registration transformation parameters. Finally, the similarity measure function reaches the optimal state with the help of the optimization algorithm, to achieve the best registration effect. The registration pipeline is provided in Fig. 3 (see the main text).

The Development Process of MARNS Application

(1) Model construction: Firstly, computed tomography (CT) and/or magnetic resonance imaging data of the patient's spine are obtained, and then 3-dimensional (3D) reconstruction is performed to obtain the three-dimensional virtual model stereolithography file of the patient's spine, which is imported into 3D Studio Max (Autodesk, San Francisco, CA, USA) for fine processing and converted into filmbox (FBX) file. Import the FBX files into Unity 3D (Unity Technologies, San Francisco, CA, USA), and the construction and coding of our augmented reality (AR) scenes are all done in Unity 3D.

(2) Marker identification: First, we obtain the image of the current frame of the camera and convert it into the format supported by Unity 3D, and then convert it into a grayscale image, and use the image processing algorithm to perform threshold segmentation and binarization. Then, we use the contour extraction and screening algorithm to screen out the area where the Marker is located in the image, and use the corner extraction algorithm to calculate the corner position. Then, the Marker image in the orthogonal view is obtained by perspective transformation and matched with the Marker data prepared in advance, so as to determine the position and posture of the Marker.

(3) Marker tracking: The subpixel corner points extracted in the image coordinate system are matched with the corresponding corner points in the world coordinate system, and then the spatial multi-point perspective problem is constructed to estimate the motion and solve the spatial position and attitude of the camera in real time. A mobile device (Vivo X60, Vivo, Dongguan, China) operating on the Android platform and an iron plate with a quick response (QR) code as fundamental hardware components were used in this study. The virtual 3D model and QR code image were imported in the Unity engine (ver. 2019.4.31, Unity Technologies, San Francisco, CA, USA). A specific metal point on the electrode patches within the 3D model was designated as the central reference point. Using this reference point, the QR code image was accurately positioned on the dorsal region of the virtual body surface. Subsequently, a spatial relationship was established between the 3D model and the QR code within the virtual reality environment of the Unity engine. Subsequently, an application was developed for future deployment on an Android smartphone device.

(4) AR implementation: Create a new scene in Unity 3D, then set up a camera dedicated to showing the AR, set the spine, skin, lesion site and incision area of the 3D virtual model with different colors and opacity, and correlate it with the coordinates of the Marker, and then package it as an Android Package Kit program running on the Android platform. The Marker is tracked by calling the camera every frame, and then the virtual model is rendered by the renderer of Unity 3D to display the spatial position and posture of the 3D virtual spine model in real time. The flowchart of MARNS application development were shown in Fig. 4 (see the main text). The use of MARNS were also showed in the Supplementary video clips 1-2.

Correction for Matching Errors Caused by Body Position Changes

In this study, postural differences were taken into account and an algorithm was used to try to adjust them. In clinical practice, we observed that there were deviations between the 3D model presented by MANRS and the real-life scene, and we have developed a correction formula to correct this bias. In intraspinal space-occupying surgery, our main goal is to locate the horizontal position of the spinal canal. Therefore, we pay special attention to the displacement of the space-occupying lesion in the long-axis direction of the spine. Based on the images displayed by the MARNS system, we simplified the relevant geometric models and conducted detailed mathematical derivation (Supplementary Fig. 1).

Specifically, we calculated the actual displacement distance of the space-occupying lesion in the long-axis direction of the spine by measuring the angle between the three-dimensional model and the actual situation and using a series of geometric relationships. The formula is as follows:

$$d = O'T' = d1 + d2 \approx OT'' \tag{3}$$

$$d2 = d\,(1 - \cos\beta) \tag{4}$$

Therefore, the actual displacement of the tumor can be expressed as d2 or -d2, indicating d2 above or below point T'.

The results indicate that due to alterations in body position, intraspinal tumors exhibited displacement along the long axis, with a mean shift of 0.16 cm (Supplementary Table 3). We believe that the spine is a relatively rigid structure, so that during spinal surgery, although the patient's position may be different from the position when the preoperative CT was taken, the displacement distance of the intraspinal tumor on the longitudinal axis is usually very small and can be ignored. Therefore, for body position changes that lead to obvious matching differences, based on the proposed double error theory, we only need to vertically project the T' point of the virtual tumor model displayed by MARNS onto the patient's back skin T point in the real world, instead of the virtual skin T" point of the model. The distance d3 between the T point and the T" point is approximately equal to the depth from the body surface to the tumor $h \times \tan\beta$. When $\beta = 14°$ and $h = 6$ cm, $d3 = 1.5$ cm, which means that for this patient the center of the incision T" point displayed by the MARNS frontal scan need to move it up about 1.5 cm to the T point. In summary, for the horizontal positioning of the spine, leveraging the visualization advantages of AR technology, we only need to vertically project the 3D tumor model, displayed by the MARNS scan in the lateral view, onto the actual back body surface to determine the center of the incision.

https://doi.org/10.14245/ns.2448188.094